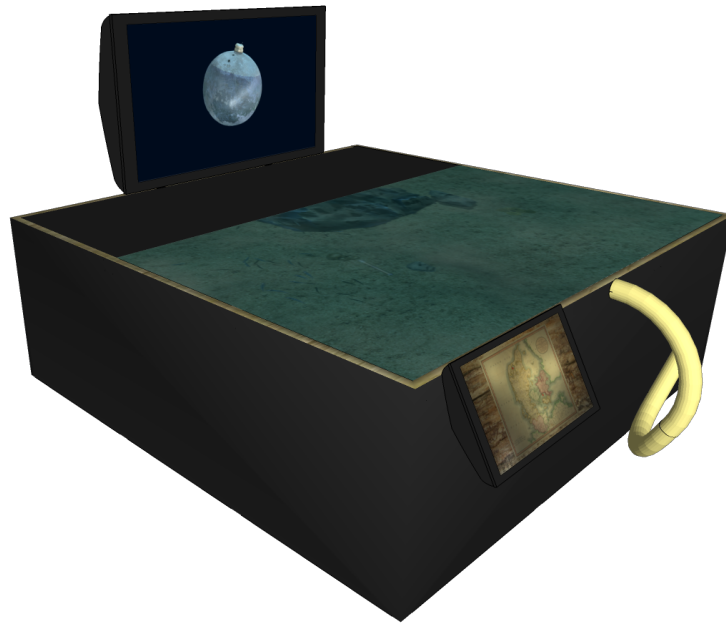


Embodied Immersiveness

- Udvidelse af en interaktiv installation



Gruppe 10 / Støvsugerbanden

Roskilde University

IMT

Titel:

Embodied Immersiveness

Projekt Periode:

Efterårs Semester 2016

Projekt Gruppe:

10

Medlemmer:

Rune B. Barrett

Studie Nummer: 58645

E-mail: runeb@ruc.dk

Nikolai P. Hjorth

Studie Nummer: 58477

E-mail: npeterh@ruc.dk

Sebastian A. V. Jakobsen

Studie Nummer: 57938

E-mail: savj@ruc.dk

Andreas O. Thomsen

Studie Nummer: 58650

E-mail: aobelt@ruc.dk

Jakob Thormod

Studie Nummer: 57312

E-mail: thormod@ruc.dk

Patrick F. Østergaard

Studie Nummer: 57963

E-mail: pafioe@ruc.dk

Vejledere:

Maja Fagerberg Ranten

Henning Christiansen

Sideantal: 219

Afleverings Dato:

22. december 2016

Abstract:

The objective of this project is to redesign, build and explore the significance an interactiv installationen has on its user. The installationen is placed at a museum and centers around an archaeological excavation under water. The focus is to examine the installations ability to form credibilty which in term forms immersiveness in order to mediate marinearchaeological in an inventive way. Through an exploration of archaeology, subjects & objects in museums and the implication of embodied perception in a virtual enviroment, this project justifies how we evaluate this interaction.

Notably this projects is formed on a previous project regarding the same installation. This project is an expansion of this. Furthermore the installation merits the ability to interact with a 3D generated seabed in which you can dig for historical artifacts.

As part of this project a comparison of rigid and agile models along with a revised design model is evaluated in order to optimize our design process. Additionally this project documents which technical means that amplifies immersiveness. These implementations consists, among other, of a Swarm Intelligence Algorithm in the form of animated fish, a description of the programming pattern ECS and tracking improvements.

In conclusion this project shows how, in order to form credibilty with the user it is not necessary to provide a real situation, but rather simulate it. Furthermore, in order to instantiate immersiveness with the user of the installation, and in order to mediate a marinearchaeological excavation, the visual output needs to be absolutely smooth. Any form of lag or misintrepeted interaction will break the given immersiveness.

Indhold

1	Introduktion	1
1.1	Indledning	1
1.1.1	Det tværfaglige	2
1.1.2	Fokus	3
1.1.3	Videnskabsteoretisk Retning	4
1.1.4	Dimensioner	4
2	Problemfelt	6
2.1	Problemformulering	7
2.1.1	Problemdeling	7
3	Arkæologi, subjekt og objekt	9
3.1	Arkæologi og Samfundet	9
3.2	Autenticitet, Branding og Udgravning	11
3.2.1	Fra Objekt til Subjekt	12
3.2.2	Den forskellige materialitet i det digitale	13
3.2.3	Fratræden fra de radikale implikationer i det digitale	14
3.2.4	Nye former for videnskonstruktion	16
3.2.5	Beslaglæggelse af det digitale	19
3.3	Interaktions-Design med den levende, tænkende krop	20
3.3.1	Embodied Interaction	20
3.3.2	Embodied Perception	23
3.3.3	Immersiveness	24
3.3.4	Interaktion som et værktøj	25
4	Metode	26
4.1	Deltagerobservation	26
4.2	Semistruktureret Interview	27

Indhold

4.3	Ekspertinterview	29
5	Design	30
5.1	Design, Model og Proces	30
5.2	Design Iterationer	36
5.2.1	Fase 1.	37
5.2.2	Fase 2.	37
5.2.3	Fase 3.	37
6	Teknisk Afsnit	39
6.1	Generel Software Beskrivelse	39
6.1.1	Unity, Spil/Interaktive Installationer & Entity Component Systemet	40
6.1.2	Multidisplay: Maindisplay, Site Finder og Infoskærm.	42
6.1.3	Distributed Behavioral Model of a School of Fish	43
6.1.4	Site Finder & Touch	47
6.1.5	Animation	48
6.2	IR Tracking	49
6.2.1	Den Forbedrede Tracking	50
7	Analyse & Diskussion	55
7.1	Analyse	55
7.2	Diskussion	58
8	Konklusion	63
9	Perspektivering	65
9.1	Alternative modifikationer	65
9.1.1	Loftspjektion	65
9.1.2	Lyd	65
9.2	Generelle forbedringer	66
9.2.1	Site Finder	66
9.2.2	IR tracking	66
9.3	Fremtidige planer for installationen	66
10	Acknowledgement	67
	Litteratur	68

11 Bilag	72
A Tværfagligt samarbejde	73
A.1 Gennemgang af Tværfagligt Samarbejde	73
A.2 STS	74
A.3 Informatik	74
A.3.1 Viden	75
A.3.2 Færdigheder	75
A.3.3 Kompetencer	76
A.4 Datalogi	77
A.4.1 Viden	77
A.4.2 Færdigheder	78
A.4.3 Kompetencer	78
B Interview Spørgsmål	80
B.1 Spørgsmål til semistrukturerede interviews	80
B.1.1 Formål	80
B.1.2 Form	80
B.1.3 Generelt:	80
B.1.4 Tekniske spørgsmål:	80
B.1.5 Oplevelsesspørgsmål:	81
B.1.6 Arkæologi spørgsmål:	81
B.1.7 Antropolog-spørgsmål:	82
B.1.8 STSamfund-Andreas-spørgsmål	82
C Interview	83
C.1 Interview med Marinarkæologer - Foretaget af Sebastian A. V. Jakobsen d. 14-11-2016	83
C.2 Interview med Marinarkæologer - Foretaget af Nikolai P. Hjorth d. 14-11-2016	89
C.3 Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016	95
C.4 Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016	108
C.5 Interview med Marinarkæolog Andreas K. Bloch - Foretaget af Sebastian A. V. Jakobsen d. 15-12-2016	123
D Diagrammer	135
D.1 Hierarki Diagram	135

E	Kilde kode	137
E.1	AddObjToWall.cs	137
E.2	AnimatedProjector.cs	139
E.3	AvoidObject.cs	140
E.4	BackButton.cs	141
E.5	Check1.cs	142
E.6	ClickToCarveTerrainVolume.cs	144
E.7	ControlDiggers.cs	147
E.8	CreateBody.cs	148
E.9	Current.cs	150
E.10	DisplayScript.cs	151
E.11	EnableCams.cs	152
E.12	Fish.cs	153
E.13	FloatingSandCurrent.cs	167
E.14	FlockAIUtilities.cs	168
E.15	GlobalFlock.cs	170
E.16	Goal.cs	174
E.17	LoadTimer.cs	175
E.18	MiddleMan.cs	176
E.19	MoveGoal.cs	177
E.20	MoveWater.cs	179
E.21	ObjectProperties.cs	181
E.22	PinSelected.cs	182
E.23	ProceduralTerrainVolume.cs	184
E.24	Restarter.cs	188
E.25	ScaryObject.cs	189
E.26	SelectSite.cs	190
E.27	SimplexNoise.cs	191
E.28	StartProcessingExe.cs	198
E.29	camZoom.cs	199
E.30	rotateObject.cs	201
E.31	trackingCast.cs	202
E.32	OSCListener.js	205
E.33	Blob.pde	207
E.34	IRTracking.pde	209
E.35	svsDebugger.pde	215
E.36	testCams.pde	218

Kapitel 1

Introduktion

1.1 Indledning

Denne opgave bygger på et tværfagligt samarbejde mellem gruppens medlemmer. Vi vil i dele af denne rapport tage udgangspunkt i en interaktiv installation om marinarkæologi på Vikingskibsmuseet i Roskilde. Denne interaktive installation var også omdrejningspunkt for udformningen af størstedelen af gruppens forrige semesterrapport. Denne rapport er derimod ikke en opfølgning, men snarere en udbygning eller videreførelse af den tidligere omtalte rapport (Barrett et al., 2016). Vi ønsker derfor i denne opgave at undersøge nye områder og implementere nye aspekter.

1.1. Indledning

1.1.1 Det tværfaglige

I afsnittet *Tværfagligt Samarbejde* (Bilag A) kan man læse yderligere om en mere specifik argumentation for, hvordan vi begrundet vores tværfaglige samarbejde. Vi vil derfor her kort ridse op, hvorledes det tværfaglige formynder sig i denne opgave, i grove træk.

Det tværfaglige i denne opgave forholder sig i sin essens mellem fagene STS (Subjekt, Teknologi, Samfund), Datalogi og Informatik. Årsagen bag skyldes at størstedelen af denne gruppe arbejdede på den omtalte interaktive installation i forårssemesteret 2016 og ønskede i denne foranledning at undersøge nye aspekter og implementere flere tekniske komponenter. Dette er den hovedsagelige årsag bag det tværfaglige samarbejde.

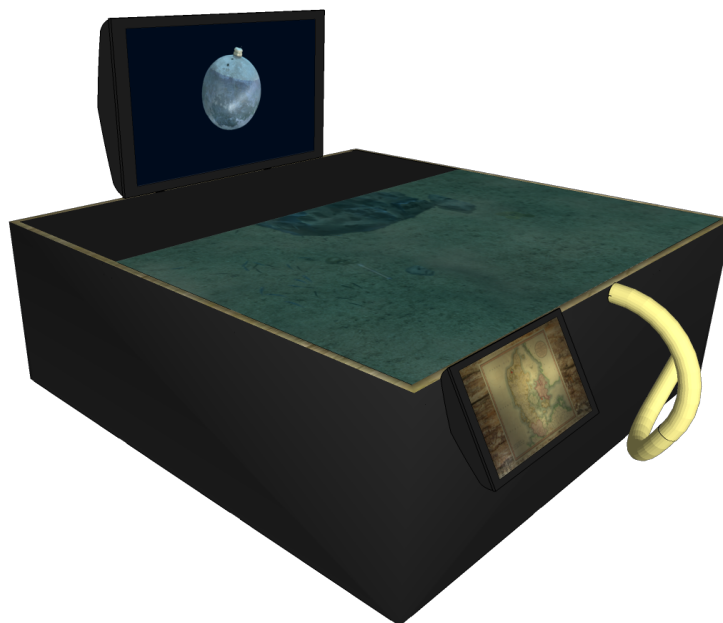
Nedenunder ses nogle grafiske illustrationer af hvordan den interaktive installation så ud før arbejdet i dette semester og dernæst ses en illustration af hvordan installationen ser ud anno december 2016.



Figur 1.1: Her ses en illustration af installationen anno maj 2016.

Kapitel 1. Introduktion

Installationen har til hensigt at simulere en marinarkæologisk udgravning, ved først at vælge et *site*, for derefter at vise en havbund, med skjulte genstande under sandet. Ved at tage det gule plastik rør (sug), som vist nedenunder, og føre det henover den viste havbund, forsvinder sandet og genstandene afdækkes. Når man har fundet en genstand, af arkæologisk betydning, vises den på den øverste skærm.



Figur 1.2: Her ses en illustration af installationen i nuværende stand anno december 2016.

1.1.2 Fokus

Det er vores mål med denne opgave at det tværfaglige fokus tydeliggøres af opgavens indhold, og forholdet mellem disse. I den tidligere rapport (Barrett et al., 2016) ønskede vi at undersøge hvorledes man bygger en interaktiv installation om marinarkæologi i en museums kontekst. Dette projekt handler derimod om en undersøgelse af installationens evne til at drage brugeren ind i dens univers, og forøge fordybelse og på denne måde give indsigt i hvordan arkæologer udfører udgravninger under vandet. Dette vil vi anskue gennem en undersøgelse af objekter på museer, arkæologers plads i samfundet, offentlighedens forhold til arkæologi, analyse af tests, gennemgang af tekniske implementeringer og meget mere. Det er derfor værd at bemærke, at denne opgave er en viderebygning af gruppens

1.1. Indledning

tidligere fokusområde - men ingen forlængelse.

1.1.3 Videnskabsteoretisk Retning

I dette projekt har vi valgt at anvende den postfænomenologiske videnskabsteoretiske retning. Denne tilslutter vi os for at kunne sige noget om hvordan vi undersøger. Denne retning beror, i vores projekt, på det fænomen der opstår i interaktionen mellem brugeren og installationen. Dette gøres for at kunne analysere på de interviews og tests vi har foretaget i forbindelse med projektet. Betydningen af denne videnskabsteoretiske retning vil foruden blive illustreret gennem teksterne af (Dourish, 2004) og (Svanæs, 2013). Det er vores forståelse at den postfænomenologiske retning anskuer subjekt og teknologi som indlejret i hinanden og derfor uadskillelige. I teksten *A field guide to Postphenomenology* af (Rosenberger and Verbeek, 2015) beskriver de hvorledes den amerikanske filosof Don Ihde anskuer dette forhold mellem menneske og teknologi i hans bog (Ihde, 1990). Ihde udtrykker som følger:

"When a technology is 'embodied,' a user's experience is reshaped through the device, with the device itself in some ways taken into the user's bodily awareness."

Rosenberger and Verbeek (2015, p. 14)

1.1.4 Dimensioner

- › Datalogi. Dimensionen indenfor Datalogi opfyldes gennem det tekniske afsnit. Dette afsnit består af en software gennemgang og et overblik over de tekniske dele vi har anvendt, samt kodeeksempler.
- › Informatik. Denne dimension opfyldes gennem design afsnittet og det tekniske afsnit. Med et indblik i forskellige typer modeller og diagrammer vil vi forsøge at skabe overblik og forståelse over vores design process og programmets dele.
- › STS. Dimensionen indenfor STS opfyldes gennem det teoretiske afsnit, metoden, analysen og diskussionen. Det er tanken at STS delen danner rammerne for opgavens teoretiske del og på denne måde pakker de øvrige dimensioner ind.
- › D&K. Design og Konstruktion opfyldes gennem design afsnittet, hvor vi berører agile og rigide modeller, samt en revidering af en tidligere anvendt design model.

Kapitel 1. Introduktion

- › TSA. Teknologiske Systemer og Artefakter indgår under det tekniske afsnit, hvor vores installation, i sig selv, forstås som et teknisk system, med et formål.

Kapitel 2

Problemfelt

I dette afsnit vil vi præsentere vores problem og skabe forståelse for årsagen bag de aspekter vi ønsker at undersøge. Det er hensigten at dette afsnit skal give læseren indblik i hvorledes vi forstår og ønsker at formidle vores problem før vi præsenterer det teoretiske afsnit.

Som nævnt i introduktionen er gruppen et tværfagligt samarbejde, der omhandler en interaktiv installation. Indledningsvis i projektet besluttede vi os for hvad vi gerne ville undersøge og implementere i dette semester, for dels at kunne forbedre den interaktive installation, men især også den interaktive oplevelse, uden at komme for tæt på det tidligere arbejde.

I gruppens tidligere rapport (Barrett et al., 2016) blev installationens evne til at skabe nysgerrighed i forhold til de arkæologiske genstande den indeholdt, undersøgt. Samt et indblik i den kontekst installationen var tænkt ind i, baseret på museers plads i samfundet. Vi ønskede derfor, som vi allerede har berørt, at undersøge nye, uberørte områder, og specifikt den samlede interaktive oplevelse, funderet ud fra det univers installationen præsenterer og dens evne til at kunne præsentere en del af marinarkæologiens faglighed, mere specifikt - en arkæologisk udgravning under vandet.

2.1 Problemformulering

- Hvordan designer vi en interaktiv installation som agter at formidle troværdigt om en arkæologisk udgravning under vandet og som gennem tekniske virkemidler formår at skabe fordybelse for brugeren?

2.1.1 Problemdeling

Vores intention er nu at skabe overblik over hvorledes vi ønsker at undersøge de forskellige dele af vores problemformulering. Indledningsvis kan den første del af problemformulering opdeles som følger:

- › *Hvordan designer vi en interaktiv installation?*
- › *Hvordan formår den interaktive installation at formidle troværdigt om en arkæologisk udgravning under vandet?*

Denne opdeling følger at det er opgavens formål at skabe tværfaglig forbindelse mellem fagene: Datalogi, Informatik, STS, D&K og TSA. Disse fag illustrerer derfor forskellige indgangsvinkler for at undersøge det stillede problem. Hvorledes vi designer, forudsætter en forståelse og indsigt i arkæologers hverv, en forståelse for tekniske virkemidler, for at kunne implementere iterationer og en udvikling af en designmodel. Hvordan vi designer dækker derfor over alle områderne vi berører i rapporten, det beror både på et teoretisk og et praktisk fundament.

Hvorledes den interaktive installation formår at formidle troværdigt om en arkæologisk udgravning under vandet, ønsker vi at undersøge gennem teksterne: (Dourish, 2004), (Svanæs, 2013), (Bayne et al., 2009). Derudover har vi foretaget brugerorienterede tests og interviews som vi, ved hjælp af teksten (Kvale and Brinkmann, 2009), vil analysere for at danne belæg for undersøgelsen af dette. Det skal også nævnes at installationens evne til at skabe troværdighed er relateret til, men ikke forbundet med, dens evne til at skabe fordybelse.

Den sidste del af problemformuleringen kan opdeles og forstås som følger:

- › *Hvordan designer vi en interaktiv installation som gennem tekniske virkemidler skaber fordybelse om en arkæologisk udgravning under vandet?*

2.1. Problemformulering

Denne del af problemformuleringen omhandler installationens evne til at skabe fordybelse. Med dette menes der, mere specifikt, betydningen af det output og feedback den virtuelle interaktivitet har. Via den teori, der skaber belæg for installationens evne til at fortælle om en arkæologisk udgravning under vandet, vil vi deraf se på betydningen for brugerens muligheder for at interagere med det terræn installationen visualiserer. Installationens potentiale for at skabe fordybelse beror altså også på empiri dannet under tests og interviews, men betydningen af denne er anderledes fra betydningen af troværdighed, da det er tanken at opnåelsen om fordybelse bygger mere på installationens interaktivitet. Det er med andre ord, gennem de udvidelser vi har implementeret, *tracking* og programtests at vi kan sige noget om installationens mulighed for at skabe fordybelse.

Kapitel 3

Arkæologi, subjekt og objekt

I dette afsnit vil vi gennem teksterne (Bayne et al., 2009), (Dourish, 2004), (Holtorf, 2007) og (Svanæs, 2013) gennemgå hvorledes vi ønsker at udfylde STS delens teoretiske element. Dette ønskes således også at være gennemskueligt igennem hele rapporten. Eftersom teksterne anvendt i dette kapitel skal agere som den overbliksgivende introduktion til digitalisering af museer, og hvordan dette potentielt set kan påvirke formidlingen af arkæologi. I dette tilfælde marinarkæologien og dets fund. Derudover skal teksterne også give et indblik i hvordan vores interaktive installation vil påvirke brugerens oplevelse, af den verden han eller hun befinder sig i, og dermed også mødet mellem menneske og teknologi.

3.1 Arkæologi og Samfundet

Indledningsvis til vores STS afsnit, vil vi gerne skitsere hvad Cornelius Holtorf et al. er kommet frem til, i forhold til hvordan arkæologi er forbundet med samfundet. I kapitel 4: *What people are thinking about archaeology* tager Holtorf (Holtorf, 2007) udgangspunkt i en række undersøgelser foretaget af andre, for at danne et generelt overblik over dette. Vi er klar over at de data vi benytter os af, ikke er så nutidige som vi kunne ønske os. Ikke desto mindre er de relevante og ydermere kommer Holtorf (Holtorf, 2007) selv ind på et problem der viste sig i forhold til vores projekt.

3.1. Arkæologi og Samfundet

"Nevertheless, I would have loved to obtain some new empirical data of how people actually perceive archaeology and archaeologist. Unfortunately I did not succeed in getting the necessary funding. (...) Unfortunately the amount of such existing research is very limited."

Holtorf (2007, p. 51)

Der er flere undersøgelser og interviews nævnt af Holtorf, (Holtorf, 2007) men vi har valgt at tage udgangspunkt i tre af dem. Eftersom de alle er nævnt for at belyse offentlighedens syn på arkæologi, så omhandler denne del mere specifikt arkæologi i forhold til samfundet. Som det fremgår af de nedenstående undersøgelser, så er det gennemgående tema, i forhold til hvordan samfundet forholder sig til arkæologi, udgravninger.

- › En undersøgelse foretaget af Anders Högberg (Högberg, 2004) viser at ud af 50 svenskere, som bliver spurgt hvorledes deres associationer er i forhold til arkæologi. Der er det hyppigste svar udgravninger og udgravningsredskaber (26%) (Holtorf, 2007, p. 56).
- › I et interview foretaget af Lisa Mackinney (Mackinney, 1994) hvor spørgsmålet: "What comes to mind when you hear the word 'archeology'?", bliver stillet, er det hyppigste svar, "digging (32%) (...)"(Holtorf, 2007, p. 56)
- › Den tredje undersøgelse er foretaget af Ramos og Duganne (Ramos and Duganne, 2000) her blev der spurgt: "What do you think of when you hear the word 'archaeology'?".(Holtorf, 2007, p. 54) Her er det igen, "(...)digging(...)" altså at grave der er det hyppigste svar. (Holtorf, 2007)

"Ironically, although archaeologists are usually obsessed with context in the past, the cultural and social context of archaeology itself is still little known."

Holtorf (2007, p. 61)

Holtorf(Holtorf, 2007) fremlægger en model kaldet *The Public Relation Model* i kapitlet *Strategies of engagement*. Denne model beskriver hvordan arkæologer har en 'afhængighed' af samfundet. Denne 'afhængighed' skal forstås således, at arkæologernes akademiske arbejde, paradoksalt nok, kun eksisterer i kraft af det samfund vi lever i.

Holtorf citerer John Cole(Cole, 1980, p. 23)

"On the selfish, professionalistic level, archaeologists have a stake in building and preserving a public constituency interested in their research if they are to keep their jobs, grants, book sales, and even their data base."

Holtorf (2007, p. 115).

Dette er uhyre interessant for os, da mødet i mellem offentligheden og arkæologien for vores projekt sker på vikingeskibsmuseet. Faktisk kan vi sige, at vores installation i dette partikulære tilfælde er mediet hvori arkæologien som praksis får mulighed for at vise sig for offentligheden. Og som der også beskrives:

"In other words, even a false image may need to be cultivated if that is what secures public support and interest in an entire discipline, ultimately perhaps even assuring its survival."

Holtorf (2007, p. 116)

Altså selvom vi kun viser en del af arkæologien, kan det være nødvendigt for at bibeholde, eller endda forstærke interessen for arkæologien og dets virke. Vi vil derfor i det kommende afsnit kigge på forholdet mellem individ, teknologi og formidling.

3.2 Autenticitet, Branding og Udgravning

Bayne et al. beskriver i deres artikel *Objects, Subjects, Bits and Bytes: Learning from the digital collections of the National Museum* (Bayne et al., 2009), hvordan der er stor forskel på hvordan man oplever museer og hvordan det fysiske element, der opstår i virkelige museer påvirker læringen i modsætning til virtuelle museer.

Deres artikel omhandler også hvordan læringen skal være museets eksistensgrundlag i stedet for de fysiske objekter på museet. De opstiller des videre en række vigtige spørgsmål i sammenhæng med 11 forskellige britiske museer som de har interviewet i forbindelse med det projekt artiklen omhandler.

I denne anledning bliver der fremlagt et interessant emne, i form af hvad et objekt egentlig er. De beskriver således at et objekt ikke længere handler om at have noget man kan tage fysisk fat i. Udtrykket 'objekt' har derimod fået en mere flyvsk betydning.

3.2. Autenticitet, Branding og Udgravning

“In the context of digitization and the placing of collections online, this turning of the gaze away from the object is further complicated by the fact that the nature of the object itself has radically changed.”

Bayne et al. (2009, p. 111)

Ud over dette, vil det også have en effekt på den måde man lærer af museer. Der er behov for at der er en vis ægthed, over for det objekt man beskuer online. Dette er noget der kan vise sig at give besvær, siden internettet er kendt som et sted hvor alt kan forfalskes.

3.2.1 Fra Objekt til Subjekt

Der bliver i artiklen *Objects, Subjects, Bits and Bytes: learning from the digital collections of the National Museums*(Bayne et al., 2009), beskrevet hvordan de virkelige museer vil blive nødt til at nytænke den måde de lever og skaber deres oplevelse på. Museer, der vil sætte fokus på en læringsbaseret oplevelse, bør mobilisere et paradigmeskift som blandt andet vil medføre at museer ikke længere er et sted hvor genstande skal konserveres(Bayne et al., 2009, p. 145) (Hetherington, 2000).

Hawkey (Hawkey, 2004) beskriver hvordan objektet, i dette nye paradigme der vil blive opstillet, skal være sekundært i forhold til den besked det skal videregive til brugeren (Bayne et al., 2009, p. 112) (Hawkey, 2004).

Objektet der befinder sig på museet, har stadig en essentiel rolle at spille i museumsoplevelsen. Dette danner en form for autoritet hos museet, da det er en del af museets sjæl. Det er dog vigtigt at objektets fysiske tilstedeværelse aldrig overskygger det som objektet har til formål at kommunikere.

“As the modern world and its conditions are now changing beyond recognition, both Thomas (Thomas, 2004) and Staaf (Staaf, 2000) wonder whether that means that scientific archaeology and heritage management, too, will need to change in order to remain relevant.”

Holtorf (2007, p. 11)

I det ovenstående citat skitserer Holtorf et behov fra arkæologiens side, til at forny sig. Det værende fra den allerede eksisterende akademiske tilgang, og over til en mere formidlende tilgang. Holtorf (Holtorf, 2007) og Bayne et al. (Bayne et al., 2009) beskriver begge nødvendigheden for et paradigmeskift i takt med at samfundet bevæger sig længere og længere ind i den digitale alder. Således må museer,

som alle andre institutioner, finde en katalysator for at udvikle sig, med formål at forny sin plads i samfundet. Yderligere skitseres et behov i kraft af at vores samfund tilbyder en sikkerhed, og den enkelte ganske simpelt har flere ressourcer til at søge 'nye' oplevelser og eventyr, som kan tilbyde et afbræk fra hverdagen.

"As people in affluent Western societies have become economically secure and possess all the tools they require, they are orientating their lives more and more towards experiences: to live and to experience have nearly come to mean the same thing."

Holtorf (2007, p. 4)

3.2.2 Den forskellige materialitet i det digitale

Fysiske museer har en fordel over digitale museer, nemlig kommunikation. Det er nemmere at holde en samtale, og skabe kommunikation mellem mennesker, hvis man står i rum sammen (Feenberg, 1989). Dette er blandt andet grundet det kropssprog, som man kan tage i brug, i modsætning til en online chat hvor det primært bare er tekst som bliver benyttet. I samme form har et fysisk objekt en vis håndgribelighed og materialitet over sig, hvor det digitale objekt kan ses mere som mobilt og flygtigt.

"'Space offers no resistance to bytes on the Internet', and the digital object can 'circle the globe in nanoseconds'"

Poster (2001, p. 92), Bayne et al. (2009).

Man kan muliggøre flere aspekter med digitale objekter. Du kan redigere med næsten endeløse grænser. Ross Parry (Parry, 2007) trækker opmærksomhed til variabiliteten, omskifteligheden, flygtigheden og muligheden for at redigere og omvende det digitale emne. Derimod har virkelige objekter, på trods af at de er indelukkede og sværere at interagere med, en vis form for autenticitet og autoritet til sig. Dette er grundet den håndgribelighed som nævnt tidligere.

"For variability interferes with the authership and authority of the curator, and yet allows new narratives to be told and new voices to be heard"

Parry (2007, p. 102), Bayne et al. (2009)

Artiklen af Bayne et al. (Bayne et al., 2009) *The National Museums Online Learning Experience(NMOLE)* beskriver at der er andre eksempler der tager præcedens, og har mulighed for at kunne give lige så stor pote. Der er for eksempel begyndt at bli-

3.2. Autenticitet, Branding og Udgravning

ve konstrueret folksonomier (indeksering af data, eller information, lavet af mennesker der benytter selvvalgte ord) hos digitale objekter, i form af *bruger-tagging* som kan gå hånd i hånd med mere konventionelle metoder, til at søge og klassificere kollektioner.

Derudover er der et projekt kaldet *Commons on Flickr* som lader brugerne lave digitale kollektioner hvor de kan lave deres egen kommentarer, og *tags* ved at gå på lokale netværk. *Commons on Flickr* tager plads på et langtids etableret socialt netværk, som kan virke mange steder i verden hvorimod at et museum, som for eksempel *Brooklyn Museum*, kun har *bruger-tagging* inden for museets parametre (Bayne et al., 2009, p. 113).

3.2.3 Fratræden fra de radikale implikationer i det digitale

Parry beskriver i sin analyse hvordan museer og teknologi begynder at finde ligevægt, selvom teknologi stadig er 'konstruktivt forstyrrende' (Parry, 2007, p. 140). Museer og computere er efter to generationer begyndt at finde sammen (Parry, 2007, p. 138). Førhen er der blevet diskuteret hvorvidt der findes autenticitet, i det digitale objekt. Med en sådan diskussion kan man indføre en mere nuanceret definition på værdien af et objekt.

Med denne re-definition på det digitale objekt bliver definitionen 'objekt' mere afhængig af de kulturelle forhold, hvor objekter bliver set som at være i konstant bevægelse og i den forstand bevæge sig igennem forskellige medier (Parry, 2007, p. 68).

" (...) Control of media production and with it, control of the learning process, is moving from the traditional producers to consumers, from transmitters to receivers, from teachers to learners. Yet the development of these technologies will, without doubt, also reduce control of knowledge by public institutions. (...) Once data has left the museum and become available digitally, it may be beyond copyright protection, especially in the huge deregulated zone of informal digital learning."

Anderson (1999, p. 21).

Dette citat af Anderson (Anderson, 1999) beskriver i høj grad hvordan den fysiske verden, begynder at blive mere digitaliseret. Dette medfører også, at kontrollen af mediet bliver yderligere ukontrolleret. Med dette menes der, at når forandringen fra det fysiske til det digitale udvikler sig, vil det betyde at kontrollen af hvordan

Kapitel 3. Arkæologi, subjekt og objekt

man lærer fra det digitale, vil udvikle sig. Derfor vil der forekomme et skift i og med at indlæringen fra det digitale, vil blive mere tilgængeligt for studerende. Når først det har forladt museet og er blevet uploadet til internettet, bliver det sværere at gøre krav på sin ophavsret.

Selvfølgelig er det ovenstående citat fra en tid, hvor internettet ikke på samme måde var udbredt og usikkerheden i krav fra ophavsret, er ikke på samme måde eksisterende i den nuværende digitale tidsalder. Det er dog stadig en valid pointe at fremhæve. Så snart noget er blevet uploadet til internettet, kan det i bred grad findes på diverse ulovlige sider helt gratis, selvom intentionen til at starte med har været at skabe en profit på det digitale produkt. Anderson (Anderson, 1999) beskriver videre, hvordan der forekommer en marginalisering af denne form for digital læring, samtidig med at det bliver mere anerkendt.

"Museums and galleries offer a unique kind of learning, based on first-hand experience of authentic objects, works of art and other resources in a public, social environment"

Anderson (1999, p. 31).

Der vil muligvis ikke være den samme overtagelse fra det digitale, som man kunne have frygtet. Der findes en form for antagelse af, at det digitale ikke er lige så godt som det fysiske, da det ikke giver den samme *first hand* oplevelse.

Den digitale læringsform bliver set mere som en forbedring af det fysiske, i stedet for at lave en fuldkommen re-definition af, hvordan læring kommer til i den digitale tidsalder (Bayne et al., 2009, p. 13).

"No matter how one animates the digital object or captures it in high resolution, the object received through a monitor seems remote. Its materiality, its being, its existence as proof, as evidence – its true value – remains illusive. The emotive experience of seeing the real requires the real and no surrogate will do. A virtual visitor may understand the thing better and be better prepared to interpret it when they see it but they receive those peculiar attributes of real things only through real world engagement."

Knell (2003, p. 140)

Det ovenstående citat fra Knell (Knell, 2003) indkapsler yderst godt, hvordan det digitale til en vis grad ikke vil have kapaciteten til at kunne overtage det fysiske objekt. Som Knell (Knell, 2003) beskriver har det digitale objekt en evne til at virke fjern. Samtidig kan dens bevis på faktisk at eksistere forblive uvirkelig. Som

3.2. Autenticitet, Branding og Udgravning

nævnt, vil mange af de ting man kan lære fra objektet være tilgængelige i den digitale verden, men nogle af de fysiske attributter der kommer fra at se objektet i virkeligheden, kan på flere måder overgå det digitale objekt. I vores installation har vi prøvet at imødegå dette problem mellem virtuelle og virkelige objekter. Når man som bruger afsøger havbunden i vores installation og hen af vejen finder de virtuelle genstande i programmet, så vil de selvsamme genstande, som udgraves, også være opstillet i et virkeligt format. På den måde oplever man både det virtuelle objekt og objektet i virkeligheden. Dette lader sig gøre, da vi har fået 3D objekter i vores installation af fire virkelige objekter.

3.2.4 Nye former for videnskonsstruktion

“The tendency of the digital to disrupt the distinction between producer and consumer has existed since the early days of the internet, though it has gained additional momentum and immediacy since the emergence of the ‘Web 2.0’ paradigm”

Bayne et al. (2009, p. 115).

I dette citat fra Bayne et al. (Bayne et al., 2009) bliver der beskrevet hvordan den førhen meget klare linje mellem producent og forbruger, begynder at blive mere og mere sløret med *‘web 2.0’ paradigmet*. Dette paradigme inderbærer en ny generation af internettet, hvor tjenester har mere interaktion med brugeren og giver derfor brugere og tjenester mere frihed (Bayne et al., 2009, p. 115). Den nye generation af internetbrugere har et meget mindre forbrug af manualer og har, til en vis grad, meget nemmere ved at kunne finde rundt i den brede verden af nettet. Ved at have en fast forbindelse til denne verden, har studerende ikke bare mulighed for at konsumere information, men de har også mulighed for at genskabe det og videregive det som deres eget (Lorenzo, 2006, p. 2).

“In such a media context, another element of power shifts to the user – alongside the ability to manipulate and share the digital object itself, there emerges the capacity to interpret and make meaning from digital object collections within a global public sphere.”

Bayne et al. (2009, p. 116).

Som nævnt får forbrugeren mere og mere kontrol over hvad der bliver lært på nettet. Samtidig begynder der også at opstå en frihed til at kunne lære de digitale objekter at kende, som man selv forstår dem og ikke som en lærer fortæller, at man

Kapitel 3. Arkæologi, subjekt og objekt

skal forstå dem. Udover dette vil brugerne af den digitale verden også vise sig at være et skridt foran dem, der ikke har fordybet sig på samme måde. Hvis man ikke har samme niveau af fordybelse, vil man meget naturligt begynde at falde bagud i forhold til online museer (Bayne et al., 2009, p. 116).

Som tidligere beskrevet i afsnittet *Fra Objekt til Subjekt* (3.2.1), så blæser fornyelsens vinde og de truer med at vælte de gamle paradigmer, hvis ikke de nytænkes - det samme gælder formidlingen af arkæologien.

Holtorf citerer:

"Nick Merriman (Merriman, 2004, p. 8)(2004: 8) is right in stating that archaeologists have largely been 'communicating blindly to an audience they do not understand, and [that] it is no wonder that so many attempts at communicating archaeology result in boredom or incomprehension'."

Holtorf (2007) s. 51

Dette citat placerer vores installation som et formidlingsredskab, specifikt vedrørende marinearkæologien, som værende et skridt i den rigtige retning. Det er med andre ord dette vi forsøgte i forrige rapport (Barrett et al., 2016), sammen med Vikingeskibsmuseet og Andreas Kallmeyer Bloch, at formidle de udstillede genstandes historie og kulturer. På samme måde forsøgte vi at formidle udgravningsmetoden og arkæologiens historie om selve fundet. Altså en interaktiv mediator for arkæologens arbejde. Dette understøttes af de interviews vi foretog d. 14-11-2016 (Se bilag C, s. 83 og Se bilag C.2, s. 89) af marinearkæologi studerende, som besøgte Vikingeskibsmuseet og prøvede vores installation.

SJ: "(...)Where you digged in the sand, do you think it comes close to telling the way, like your point of view. The way you work?"

Se bilag C.1, s. 85

3.2. Autenticitet, Branding og Udgravning

WW: " (...) When it comes to like when you're excavating, it's a pretty accurate, it's as close as you can get, to simulate it on land i would say. Uhm, and i understand why you guys choose excavating, because children is definitely going to relate more to excavation. The one difficulty, the one thing, and it's just me being. The one difficulty i have with excavation, showing excavation is, maritime archaeologist probably don't do excavation as often as people think we do, and presenting it, sometimes gives them false ideas of how often we do it "

Se bilag C.1, s. 85

Som ovennævnte svar beskriver, så giver vores installation et indtryk af hvad arkæologiens håndværk indeholder - på trods af at vi kun vælger at vise en mindre del. En god introduktion til de mere spændende aspekter af marinarkæologiens verden fra en udenforstående persons synspunkt. Fra en designers vinkel, vil det næppe være meningsfuldt for os, at opstille et skrivebord med en computer på, og bede de besøgende på Vikingskibsmuseet, om at udfylde lange rapporter om de fund de gør sig. Derimod, prøver vi at tilbyde de besøgende en oplevelse af at være i marinarkæologens sted. Godt nok ikke den 'fuldendte' oplevelse, men en simulering, som kommer tæt på. Sammen med Holtorf (Holtorf, 2007) tidligere nævnte citat fra afsnittet *Fra Objekt til Subjekt* (3.2.1), er det i dag også mere oplevelsen som folk i den vestlige verden søger, frem for den egentlige situation.

"As people in affluent Western societies have become economically secure and possess all the tools they require, they are orientating their lives more and more towards experiences: to live and to experience have nearly come to mean the same thing"

Holtorf (2007, p. 4).

Det er dermed et lille glimt af et eventyr, som vi forsøger at tilbyde den besøgende ved hjælp af *immersiveness* og *embodied interaction*. Denne oplevelse er dog tidsbegrænset i kraft af, at de besøgende er på et museum og vores installation ikke er den eneste oplevelse museet indeholder. Derfor har vi valgt et aspekt af marinarkæologens arbejde, som vi formoder er interessant for Vikingskibsmuseets gæster.

Holtorf citerer Moscovici, S.

"Social psychologists have long understood that every society and every age needs to provide specific "short-cuts" for making the unfamiliar familiar"

Holtorf (2007, p. 11).

Uden at gå yderligere psykologisk til værks viser dette citat, at det enkelte individ ikke kan pålægges, at have ekstensiv viden omkring alt denne verden indeholder. Derimod, for opfattelsens skyld, vælger vi som designere af denne arkæologiske mediator, at vise det vi synes skal vises ved brugen af *immersiveness* og principper fra *embodied interaction*.

3.2.5 Beslaglæggelse af det digitale

Et af de mere generelle elementer der opstår mellem forbrugere og producenter er, at forbrugerne skal have fuld tiltro til producenterne, som helst skal være yderst mistroiske overfor forbrugeren. Dette kan der ses eksempler på i museumsverdenen. Der har været eksempler på forbrugere af museer, der har haft deres udstillinger optaget eller fotograferet hvilket har resulteret i, at udstillingerne er endt på internettet. Dette giver meget mindre autoritet til museet og overgiver al kontrol over udstillingerne til forbrugerne.

"It may be impossible ever fully to resolve the dilemmas apparent in new digital ways of working. Flexibility, access, openness, control, authority and prestige are all highly valued within museum education contexts, and taking education online problematises the relationships between these qualities."

Bayne et al. (2009, p. 120).

Som Bayne et al. (Bayne et al., 2009) nævner i ovenstående citat, kan der findes en virkelighed hvor dette samspil mellem museum og den digitale verden ikke kan eksistere, og det kan meget muligt være denne. Der er som nævnt tidligere flere kvaliteter hvis samspil korrupperer, ved at tage læringen fra museer ud til den digitale verden.

Man kan argumentere for at flere af de sidstnævnte komplikationer der findes, ved at tage museet online, ikke på samme måde er eksisterende for installationen ude på Vikingskibsmuseet, eftersom der er mere end visuelle aspekter at hente fra udstillingen. Installationen giver også brugeren en fordybelse i selve området, hvorpå installationen bevæger sig ind i den marinarkæologiske verden.

Holtorf. citerer Merriman, N (Merriman, 2004).

3.3. Interaktions-Design med den levende, tænkende krop

"Archaeologist have until recently not treated their relationship with the public as something which merited their academic attention"

Holtorf (2007, p. 61).

Dette er et citat som burde udløse en vis skepsis, men essensen i citatet kan have sin sandhed, i den formidling af udstillingsgenstande som vi hidtil har set på Vikingskibsmuseet. Det er dog en for stor udfordring i dette projekt, at komme ind på hvorvidt den almene borger har været implicit overset af arkæologer hidtil. Men det kan fastslås at Vikingskibsmuseet har taget et valg, om at tanger mere i retning af en digital interaktiv formidling, og i kraft af dette, prøver at nytænke deres plads i samfundet og deres forhold til den almene borger. Med vores installation, tangerer vi imod nogle af de pointer som vi har draget citater af fra *Archaeology Is A Brand*. Det rejsende og udforskende aspekt, bliver på sin vis mødt i vores nylige implementering af en *Site Finder*. Denne *Site Finder* skal på sigt, kunne give den besøgende valgmuligheder om hvor, og hvilke "*sites*" han/hun føler vil være spændende, at besøge og udgrave. Det udforskende bliver derudover, i hvert fald til dels opfyldt, da den besøgende selv skal udgrave, og på den måde opdage de genstande som museet synes kunne være interessante at vise frem og fortælle omkring. De genstande der er i installationen nu, er dog en blanding af arkæologiske fund, og tilfældige genstande som vi har placeret for at simulere, at ikke alt man finder under jordens overflade, har kulturel og historisk betydning. Nytænkningen af arkæologien, i vores tilfælde nytænkning af marinarkæologiens fremvisning, sker da vores installation, i hvert fald på Vikingskibsmuseet, er en nytænkning der giver nye muligheder for formidling.

3.3 Interaktions-Design med den levende, tænkende krop

3.3.1 Embodied Interaction

Denne artikel af Dag Svanæs(Svanæs, 2013) berører hvilke implikationer der er opstået ved Merleau-Pontys fænomenologi. Pontys fænomenologi berører *Embodied Interaction*, som til en vis grad kan være meget abstrakt at bevæge sig rundt om. Dette er også et emne der er blevet taget fat i af adskillige andre forskere inden for emnet. Disse indebærer Dourish, Heidegger, Schütz og Husserl som alle også er forskere Merleau-Ponty(Merleau-Ponty, 1962) tager fat i, og han benytter sig af

Kapitel 3. Arkæologi, subjekt og objekt

deres viden. En af de mere omfattende områder i artiklen indebærer *Embodied Interaction* og *Embodied Perception* hvorpå man kan drage paralleller, mellem artiklen og installationen på Vikingskibsmuseet. Derudover, vil vi også belyse nogle af de interessante aspekter fra værket (Dourish, 2004) af Paul Dourish, *Where the Action Is, Chapter 4: "Being-in-the-World": Embodied Interaction(2004)*, som på fineste vis argumenterer for, hvordan *Tangible* og *Social Computing* er baseret på den samme ide: *Embodiment*. Til dette formål, trækker Dourish (Dourish, 2004) relevante tråde til fænomenologien og nogle af de store tænkere, som Dag Svanæs (Svanæs, 2013) også gør brug af.

Embodied Interaction kan kort forklares ved at se på hvordan man benytter og ser på objekter. Et objekt begynder at gå i forlængelse af personen der bruger objektet, og Merleau-Ponty benytter sig således af et eksempel med en skrædder:

"[A tailor], when put in front of his scissors, needle and familiar tasks, does not need to look for his hands or his fingers, because they are not objects to be discovered in objective space: bones, muscles and nerves, but potentialities already mobilised by the perception of scissors or needle, the central end of those 'intentional threads' which link him to the objects given. it is never or objective body that we move, but our phenomenal body(...)"

Merleau-Ponty (1962, p. 106).

Som Merleau-Ponty (Merleau-Ponty, 1962) udpeger i dette citat, så vil en skrædder ikke se til sine hænder, for at kunne finde en kortlagt rute til at kunne sy. Alle redskaberne er i forlængelse af skrædderen og mange af de elementer af syning, som vil være svært for en nybegynder at tage fat på, fungerer automatisk for den trænede skrædder og det vil først være ved mere avancerede teknikker, at det ikke er en automatisk proces.

Ganske som Svanæs, så er det interessant for Dourish at se på de ting som for os mennesker, er 'familiære' i verden. Til det, så kommer Dourish, P med sin version af *Embodiment*. Den kalder han *Embodiment 2* og i *Embodiment 2* argumenterer han særligt for at *Tangible* og *Social Computing* høre derunder.

"as physical beings we are unavoidably emmeshed in a world of physical facts"

Dourish (2004)

3.3. Interaktions-Design med den levende, tænkende krop

"we interact daily with other people, and we live in a world that is socially constructed"

Dourish (2004)

Disse citater viser, at vi som individer i verden lever i det fysiske. Det fysiske bliver for os forstået, i den sociale kontekst det udfolder sig i. Dette skal forstås således, at mennesker overordnet set er flokdyr og de enkelte fortællinger og kulturer er baseret på de individer, der lever dem. De bygger på en sammenhæng mellem oplevelse og mening. Man kan altså se *Tangible* som værende, den del der fokuserer på vores fysiske viden om verden. Ved hjælp af denne viden kan man forbedre en interaktiv installation - som så igen, gør det nemmere at interagere med vores installation.

I vores installation er den del der er *Tangible*, suget der fjernes sand med. Altså når en bruger af installationen benytter suget, så forventes der et output, der stemmer overens med brugerens opfattelse af verden. Som fysiske væsener, er det fysiske med til at give os en forståelse af verden. At sand forsvinder, og en genstand findes, forventes der at den besidder visse fysiske egenskaber, der stemmer overens med den enkeltes kognitive overensstemmelse, mellem et givet input og forventet output.

I *Social Computing* er *Social*, som nævnt tidligere, det der giver ting mening i en social kontekst. I vores sammenhæng, vil de besøgende for det meste have et indtryk af, at arkæologer finder genstande fra en tid, der for længst er ovre. Derfor må de nødvendigvis, allerede have været gravet frem. Det er også, i dette tilfælde, noget der gør sig gældende for de fleste genstande på Vikingskibsmuseet. Allerede her, vil det derfor give mening, når man benytter sig af vores installation, at der skal graves for at finde frem til de genstande, som er skjult i installationen - ganske som arkæologerne selv har gjort det. Et andet socialt aspekt er, at når mennesker går på museum, så er det i flok. På trods af at vores installation ikke understøtter *multiplayer*, har vi alligevel observeret, at når interaktionen opstår, er tilskuerne ikke tilbageholdende når det kommer til at påvirke den enkelte, der styrer suget. Tværtimod, opstår der et fænomen både mellem mennesker og maskine, men også mellem menneske og menneske.

Dette er den samme effekt, der er ønsket ved installationen på Vikingskibsmuseet. Den optimale effekt installationen skal kunne opnå, ville være at brugeren af installationen vil få det sug, der er i brug, til at fungere i en form for forlængelse

af dem selv. Det er derfor vigtigt, at når man støvsuger sandbunden, skal det være så uforstyrret en proces som overhovedet muligt.

“The body schema is our non conscious knowledge of our lived body and of our potential for bodily actions in the world (...) If we use a tool, the body schema changes to include the tool as part of our lived body.”

Svanæs (2013, p. 12)

Som Svanæs(Svanæs, 2013) beskriver i dette citat er vi, som mennesker, meget bekendte med hvordan vores egen krop fungerer, og hvordan den virker i kraft af sig selv, uden tilegnede genstande og de egenskaber disse videregiver. Svanæs beskriver derfor hvordan kroppen, nærmest per automatik, begynder at ændre dens viden om hvad dens rumfang er, samt dens kapaciteter når et objekt bliver samlet op. Eksempelvis vil, en hammer når den samles op af dens benytter, blive en forlængelse af benytteren selv og ikke længere blot et redskab. Dette er grundet, at den valgte aktion 'at løfte objektet' gør at objektet kommer i forlængelse af kroppen af sig selv. Dette kaldes for *Embodied Perception*(Svanæs, 2013).

3.3.2 Embodied Perception

Embodied Perception fungerer i høj grad i forlængelse af *Embodied Interaction*. Hvor *Embodied Interaction* fokuserer på hvordan man interagerer med objektet, vil *Embodied Perception* fokusere på hvordan man opfatter de objekter man tager i brug, ligesom med *Embodied Interaction*, er der ét eller flere eksempler at tage i brug, for at bevise denne pointe.

“If a person tests out a light switch to see whether it works, this interaction can be regarded as an embodied perceptual act involving both eyes and hand(...)”.

Svanæs (2013, p. 16)

Som Svanæs(Svanæs, 2013) mener med dette citat, vil bevægelse for at tjekke om stikkontakten virker, være for at danne en opfattelse af stikkontakten, for at se dens opførsel. Dette fungerer til en vis grad som en brugeroplevelse, hvor man som bruger, kan få en opfattelse af stikkontaktens interaktive opførsel.

Svanæs(Svanæs, 2013) nævner også:

3.3. Interaktions-Design med den levende, tænkende krop

"In more complex interactions, such as when an experienced computer user plays the game World-of-Warcraft, the perceiving body extends into the game (...) Playing World-of-Warcraft is similar to riding a bicycle or driving a car in the sense that the technology becomes an extension of the body, but different in that the world is computer generated."

Svanæs (2013, p. 16)

Med ovenstående citat bliver der nævnt, hvordan det at søge en opfattelse i et objekt, kan blive mere kompliceret ved at gøre linjen mellem *Embodied Interaction* og *Embodied Perception* mere sløret. Et spil som *World-of-Warcraft* vil tage fat i begge dele ved at gøre spillet til en forlængelse af en selv, ved muliggørelsen af at kunne leve i en anden verden og derved 'forlade sin krop'.

Begrebet *Embodied Perception*, bliver taget i brug for at skabe en betegnelse for interaktionen mellem menneske og artefakt. Opfattelsen skal ske aktivt, og formålet er at kunne skabe en forbindelse mellem en selv og et eksternt objekt.

For at drage *Embodied Perception* ind til installationen på Vikingskibsmuseet, skal den kunne skabe en form for forlængelse af en selv, som nævnt tidligere. Samtidig skal man aktivt søge en interaktion med maskineriet, og derved forsøge at skabe en opfattelse af installationen, og lade sig fordybe sig i oplevelsen.

3.3.3 Immersiveness

Et andet begreb Dourish (Dourish, 2004) bringer på banen er *Immersiveness*. *Immersiveness* drejer sig, i korte træk, omkring oplevelsen i interaktionen. Hvis nu, som tidligere nævnt i dette afsnit, at et givet input og et forventet output stemmer overens, forstærker det installationens *Immersiveness*. Forhåbentligt vil dette, give brugeren af installationen en pseudo *Indiana Jones* oplevelse. Dette ville være et ønskescenarie for os som designere, da det understøtter vores tidligere teori fra Cornelius Holtorf og formidlingen af arkæologien. På den anden side, hvis der opstår en fejl i vores *tracking* af skærmen og inputtet ikke matcher outputtet, hvordan skal den besøgende så kunne forventes, at føle sig 'familiær' med installationen. Ydermere kunne det formentlig også give problemer i form af kognitiv dissonans, og derfra fjerne noget af den 'immersive' oplevelse.

3.3.4 Interaktion som et værktøj

Tilbage i 1927 udgav Heidegger sit værk *Being and Time*, som koncentrerer sig stærkt omkring analysen af værktøjer. Heidegger definerer værktøjer som objekter.

Meget af analysen om værktøjer i *Being and Time* arbejder videre på filosoffen Husserls fænomenologi, som også kan blive anset som kilden til den moderne fænomenologi (Svanæs, 2013, p. 6). Husserls metode bestod af en proces der blev kaldt *Transcendental Reductions*. Som Svanæs (Svanæs, 2013) nævner, skal denne proces ikke ses som en proces der indebærer at 'trække fra'. Derimod indebærer den at man i stedet kommer ind til benet i emnet, for på den måde, at fremhæve baggrunden af et fænomen.

"as part of our everyday life, a light switch is an integral element of our background of 'readiness-to-hand', and the interaction with the switch is, metaphorically speaking, transparent to us. It is only when the switch stops working as expected, or when we consciously choose to reflect on its nature, that it emerges from the 'background' as an object."

Svanæs (2013, p. 6).

Som nævnt i ovenstående citat bliver noget så arbitrært som en stikkontakt aldrig rigtigt lagt mærke til, så længe den virker. Når den virker, er den en del af vores underbevidste handlinger og det er først når den ikke længere er i fungerende form, at man begynder at hæve den fra baggrunden, og begynder at undersøge hvordan det kan være, at den ikke virker. Det samme eksempel kan tages i brug, når man sidder og skriver i et skriveprogram på computeren. Så længe skriveprogrammet fungerer optimalt, vil man ikke skænke programmet en eneste tanke, og derimod udelukkende fokusere på den tekst der bliver skrevet. Til gengæld vil programmet blive belyst i baggrunden, når det ikke længere fungerer, og der vil man begynde at lægge al sin fokus på hvad der forårsager problemet. Dermed vil teksten der bliver skrevet, i stedet blive lagt i baggrunden.

Med det i tankerne, kan man derefter se på vores installation. Så længe installationen fungerer efter planen, og den opnår sin fulde funktionalitet, vil mange af aspekterne af installationen forblive i baggrunden og aldrig skænket en tanke. Det øjeblik der går noget galt, vil installationen træde frem som et objekt, stjæle fokus, og muligvis tvinge eksempelvis suget ud af fokus.

Kapitel 4

Metode

I de følgende sektioner, vil vi præsentere de metoder vi har anvendt i projektet. Vores metoder har til formål at fremskaffe data, som vi kan sammenligne med vores valgte teori. Vi har anvendt tre forskellige kvantitative metoder: deltagerobservation, semistruktureret interview og ekspertinterview.

I dette semester har vi valgt at fokusere mere målrettet på vores dataindsamling. I det forrige semester, forsvandt meget af vores tid, ned i design og konstrueringsfaserne. Dette medførte at vores dataindsamling, i form af semistrukturerede interviews, og observationer faldede og ikke mindst manglede struktur og understøttende teori og metode. I erkendelse af dette, har vi valgt at betrede en noget grundigere vej til vores dataindsamling, og tage udgangspunkt i bogen "Interview: Introduktion til et håndværk," Kvale and Brinkmann (2009) så vi bedre kan forsvare, forstå, strukturere og ikke mindst anvende den data vi generer løbende i semesteret.

4.1 Deltagerobservation

Deltagerobservation er en kvalitativ forskningsmetode, som anvendes til at observere mennesker i deres egne omgivelser. Dette giver observatøren mulighed for selv at vælge hvor meget man engagerer sig i de aktiviteter, som man gerne vil observere. Forskellen på deltagerobservation og den klassiske observationsmetode er, at det i nogle tilfælde er nødvendigt at være en del af den sociale kontekst, for at kunne forstå den. Hvis man som udenforstående observerer på en gruppe, vil ens observationer være prægede af ens egen forudindtagede viden.

Vi vil via vores deltagerobservation være involveret med brugerne når de anvender installationen. Vi prøver derved at belyse hvordan brugerne interagerer med installationen, og med hinanden. Via disse observationer håber vi at opnå resultater, som kan hjælpe os til at forbedre installationen, og i sidste ende fortælle os hvorvidt installationen opfylder de mål vi har sat for den.

Vores observationer er foretaget på den måde, at én person fra gruppen har fungeret som deltager, i den form at vedkommende har været placeret ved installationen, for at tale med brugerne. Dette både for at kunne give hjælp til brugen, svare på spørgsmål fra brugerne, og selv stille spørgsmål under forløbet. Derudover har en anden fra gruppen siddet på sidelinjen, og nedskrevet noter af de observationer som vedkommende har gjort sig. De passive observationer fokuserer både på den verbale og nonverbale interaktion, deltagerne imellem. Inden observationerne har vi gjort os nogle tanker om hvilke spørgsmål, som kunne være relevante at stille, men samtidig har vi under observationerne stillet andre spørgsmål, som vi følte var relevante i konteksten.

"God praksis består ikke blot i at udføre praktiske handlinger, den indebærer også en situeret bedømmelse af, hvilken viden og hvilke teknikker der skal tages i anvendelse, når man handler i en given kontekst, og når man står overfor forskellige mål og værdier, der kræver et velovervejet valg"

Kvale and Brinkmann (2009, p. 107)

4.2 Semistruktureret Interview

Alle vores interviews er foretaget som semistrukturerede interviews, idet formen tillader interviewereren, at følge op på svar hvis vedkommende føler at der er interessant information at hente. Det semistrukturerede interview har et forudbestemt formål, med hensyn til hvilken information man søger at opnå. De på forhånd nedskrevne spørgsmål fungerer mest af alt som guide, og der er ikke en fastlåst rækkefølge for spørgsmålene. Derfor vil interviewet komme til at forløbe mere som en samtale, hvilket har den fordel at det kan få den interviewede til at føle sig mere 'sikker' i situationen, og man opnår mere 'reelle' svar.

Vi startede med at finde formålet med interviewet. På den måde at kunne vi strukturere interviewet, så den viden vi fik ud af det, var det relevante for os. Efterfølgende forberedte vi vores interviews, ved at dele spørgsmålene op i 3 forskellige

4.2. Semistruktureret Interview

kategorier (Se bilag B, s. 80). På denne måde blev den data vi indsamlede nemmere at overskue. Derefter foretog vi vores interviews, som blev udført henholdsvis med interviewees imens de testede vores installation, og med interviewees efter de havde testet vores installation. Vores transskription er udført i talesprog, men de citater vi har valgt at uddrage, er omskrevet til skriftsprog uden at spolere essensen.

Vores interesse er des videre fænomenet der opstår imellem subjektet og objektet. Mere specifikt brugerne af Vikingskibsmuseets besøgende i brugen af vores installation, og hvordan deres opfattelse kan være med til at forbedre vores installations evne, til at skabe troværdighed og fordybelse, men også, hvordan denne 'nye' måde at gå på museum påvirker de besøgende. I vores projekt bevæger vi os væk fra det originale glasmontre besøg, til en mere teknologisk *embodient* af en udstilling hvor de besøgende kan opleve installationen gennem deres interaktion. Det er gennem data fra denne interaktion vi søger at kunne forbedre installationen.

“Der kan bygges bro over nogle af barriererne mellem børn og voksne, hvis man interviewer børn i deres naturlige omgivelser. I mange tilfælde er det at foretrække, at interview med børn finder sted i forbindelse med andre aktiviteter, for eksempel mens barnet tegner, læser en historie, ser tv eller video eller leger med dukker eller biler”

Kvale and Brinkmann (2009, p. 166)

Som beskrevet i *Interview: introduktion til et håndværk* (Kvale and Brinkmann, 2009). Så kan der opstå komplikationer i forbindelse med interview af børn. I vores dataindsamling foretog vi blandt andet brugerinddragelse med skoleklasser, da der blev åbnet for den del af museet hvor vi har vores installation stående. I den forbindelse var det oplagt at tage højde for de problemer der kan opstå ifølge interview bogen. Eksempelvis står det beskrevet, at interviewer skal være opmærksom på blandt andet magtforhold, og i denne forbindelse ikke sætte sig selv i en position, hvor den interviewede kan komme i tvivl om hvorvidt det er en lærer der henvender sig, da det kan forpurre de data vi prøver at indsamle.

Ledende spørgsmål er også en verbal landmine, der kan underminere og så tvivl omkring vores datas validitet. Vi var derfor opmærksomme på vores spørgsmål, samt hele rummet og situationen hvor vi iscenesatte interviewet. Dette gjorde vi for at sikre os det bedst mulige udbytte. Interviews og dataindsamling med skoleklasserne fandt således sted den 1. november 2016.

Ydermere foretog vi to interviews, i forbindelse med et planlagt besøg på Vikingskibsmuseets, med følgende personer.

- › Interview foretaget den 15. december 2016: Niclas Wathne. Uddannet antropolog og ansat som formidler på Vikingskibsmuseet. Niclas stod desuden for koordinering, undersøgelse og efterundersøgelse af skoleklassernes besøg på Vikingskibsmuseet den 1. november 2016.
- › Interview foretaget den 15. december 2016: Mette Jehrbo. Ansat som formidler og står også for dele af den undervisning, der foregår på Vikingskibsmuseet.

4.3 Ekspertinterview

Et ekspertinterview udføres med en person som er ekspert indenfor et givent område. Når man interviewer en ekspert, er det vigtigt for interviewerens at have styr på det emne, som interviewet skal handle om. På denne måde vil man blive respekteret af eksperten, og derfor have mulighed for at opnå bedre symmetri i interviewet. Det er også essentielt at interviewerens er bekendt med det sprog, og de termer som eksperten anvender. Derved undgår man at der opstår misforståelser mellem eksperten og interviewerens under forløbet.

Vores ekspertinterviews er mere konkret blevet udført med følgende personer:

- › Interview foretaget den 15. december 2016: Andreas Kallmeyer Bloch, som er uddannet marinarkæolog. Andreas har vi bl.a interviewet i forbindelse med niveauet af indlevelse i interaktionen med installation, og vores installation set fra en indiviets perspektiv. Samt arkæologiens rolle i samfundet.
- › Interview foretaget den 14. november 2016: En gruppe marinarkæologistuderende fra Syddansk Universitet, som blev interviewet i forbindelse med en fremvisning af installationen. C & C.2.

Kapitel 5

Design

5.1 Design, Model og Proces

I dette afsnit vil vi anskue vores designproces og modeller der understøtter dette. Vi vil derfor kigge på hvordan Craig Larman (Larman, 2004) beskriver agile modeller og designprocesser i hans bog: *Applying UML and Patterns* og desuden revidere vores egen designmodel, som blev anvendt i vores tidligere rapport (Barrett et al., 2016) fra sidste semester.

Kapitel 2. i Craig Larmans bog (Larman, 2004) hedder *Iterative, Evolutionary, and Agile* og fokuserer på forskellige modeller og processer indenfor design. Her især de iterative og evolutionære processer og agile processer samt modeller her indenfor. I kontrast til den iterative og evolutionære proces står *Waterfall* modellen - eller på dansk vandfaldsmodellen. Denne model er en klassisk model indenfor designproces. Det er en sekventiel model, som følger at visse forholdsmæssige skridt skal tages før man går videre til næste fase i processen. Vandfalds metaforen følger nemlig den sekventielle del i dette, da man går fra en del af fasen videre til det næste, i en lineær progressiv facon, som vandet der falder ned af.

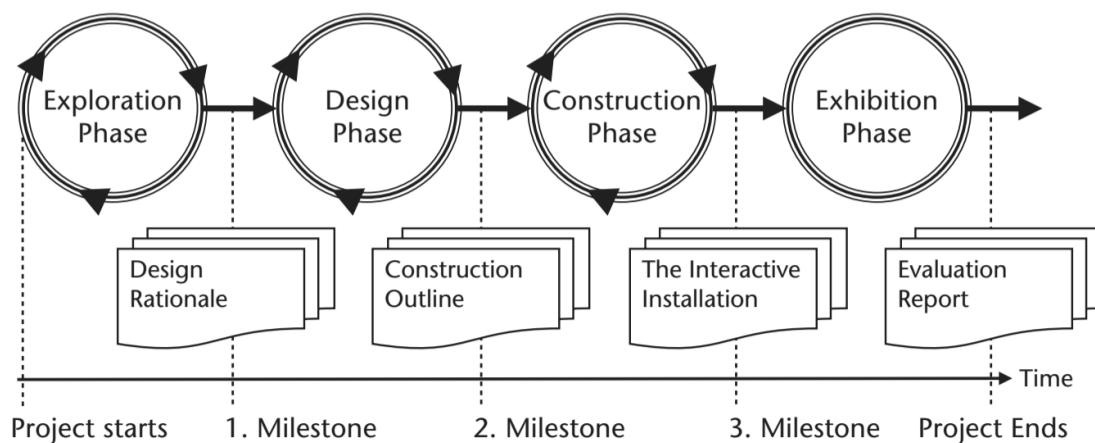
Den iterative og evolutionære udvikling som Larman beskriver det, står derfor i kontrast til vandfaldsmodellen idet at den indgår i et mere testende, sekventielt system med flere iterationer. Larman beskriver det som følger:

Kapitel 5. Design

“Iterative and evolutionary development contrasted with a sequential or ‘waterfall’ lifecycle involves early programming and testing of a partial system, in repeating cycles.”

Larman (2004, p. 55)

I vores tidligere rapport anvendte vi en designmodel fra bogen *Situated Design Methods* (Simonsen et al., 2014). Denne omhandlede en designmodel til at bygge interaktive installationer, og virkede derfor umiddelbart yderst relevant. Problemet ved denne model, i forhold til dette projekt, var at modellen var for rigid og mindede for meget om vandfaldsmodellen. Denne designmodel var nemlig bygget op om fire faser, som alle endte i en *Milestone*, som også kan beskrives som et *output* - enten et designrationale eller en endelig evaluering.



Figur 5.1: Four-phase Design Model (Simonsen et al., 2014, p. 276)

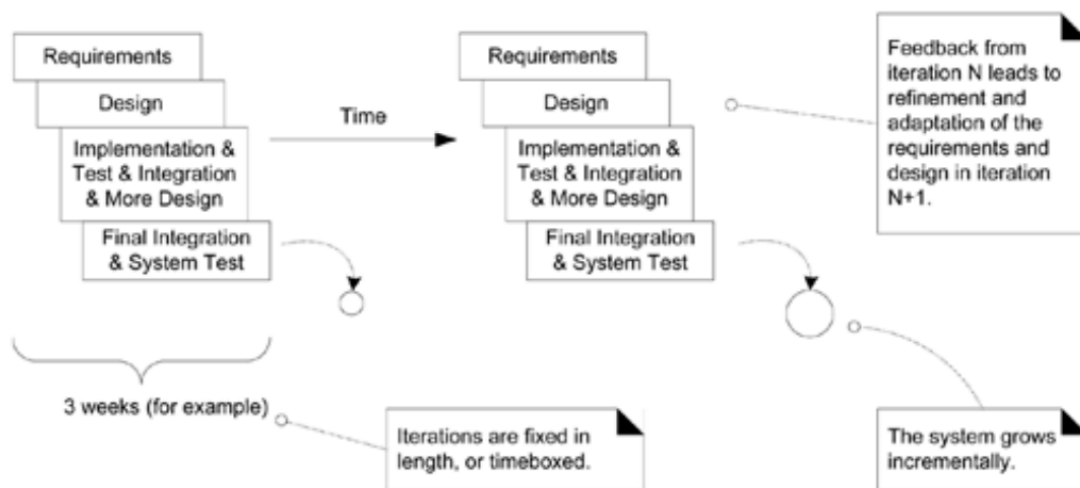
Ifølge Larman er problemet ved vandfaldsmodellen, at den antager at problemer og løsninger til disse, kan forudses allerede i starten af processen. Dette er en fejlagtig antagelse, der føjer til sandsynligheden for ikke at komme i mål med det ønskede resultat. Da krav og mål kan ændre sig ekstraordinært, er det derfor vigtigt at have en mere agil model, man kan følge da den giver muligheden for flere iterationer (Larman, 2004). Det er netop her at Larman introducerer fordelene ved de iterative, evolutionære og agile designmodeller. Det iterative, i sig selv, er baseret på at opnå en større succes, gennem gentagende raffineringer af mindre dele.

5.1. Design, Model og Proces

Da vi blev introduceret for vores designmodel, for første gang, skete det i forbindelse med et interaktions design kursus på RUC i efteråret 2015. Da vi således anvendte modellen i forbindelse med den tidligere rapport, var der dog også visse elementer vi overså. Ikke desto mindre, havde vi også et andet formål med at bruge designmodellen end ved interaktions kurset i efteråret 2015.

Formålet med dette kursus var dog også anderledes end det vi arbejder med i dette projekt og det Larman fokuserer på. Larmans fokus er eksempelvis software udvikling eller andre designprocesser, hvor man designer noget til nogen. Eksempelvis et stykke software til en kunde eller en investor. Interaktionskursets formål var derimod at eleverne dels skulle lære noget, men også at der skulle stå et produkt, som kunne anvendes i forbindelse med kursets afslutning. Vores formål med at anvende en designmodel er dog snarere tættere på Larmans. Dog ønsker vi selvfølgelig stadig at vi skal lære noget, men formålet er også at skabe en interaktiv installation som kan være troværdig i forhold til det univers, en given museums-gæst/bruger skal forsvinde ind i, og gøre dette ved at skabe fordybelse og indsigt i den marinarkæologiske faglighed.

Det er derfor vigtigt at vi, når vi designer, åbner mere op for muligheden for iterationer og en mere agil designmodel. Således at vi kan springe rundt imellem konstruktions og evalueringsfaser. Larman præsenterer eksempelvis også en model for den iterative udvikling. Denne model ser ud som følger:



Figur 5.2: Iterativ og evolutionær udvikling. (Larman, 2004, p. 58)

Kapitel 5. Design

I modellen ses to umiddelbart ensartede forløb, der starter i *Requirements* og afsluttes med *Final Integration & System Test*. Forskellen er dog at det første forløb fører til en iteration, som Larman kalder *N*. Med dette menes der, at der kan være utal af iterationer. *N* repræsenterer dermed en ukendt mængde af iterationer. Vigtigheden af dette er at antallet af iterationer, i teorien skaber et bedre design. Men hvornår er man så færdig med at designe? I teorien er man egentlig aldrig færdig med at designe, men et værktøj som Larman introducerer er *Timeboxing*.

Ved at sætte en tidsbegrænsning for iterationsforløbene, er det nemmere at gennemføre disse forløb, i en fornuftig tidshorisont. Dette illustreres også i figurens akkolade under det første iterations forløb. På denne måde kan man altså cirkulært gennemgå sit designforløb, ved at dele det op i små cykler, hvor man gennemgår hvilke krav der er gældende for ens design, og ender i test og integrering af hvad man har designet. Larman skriver som følger om den iterative cyklus:

“The iterative lifecycle is based on the successive enlargement and refinement of a system through multiple iterations, with cyclic feedback and adaption as core drivers to converge upon a suitable system. The system grows incrementally over time, iteration by iteration and thus this approach is also known as iterative and incremental development. Because feedback and adaptation evolve the specifications and design, it is also known as iterative and evolutionary development.”

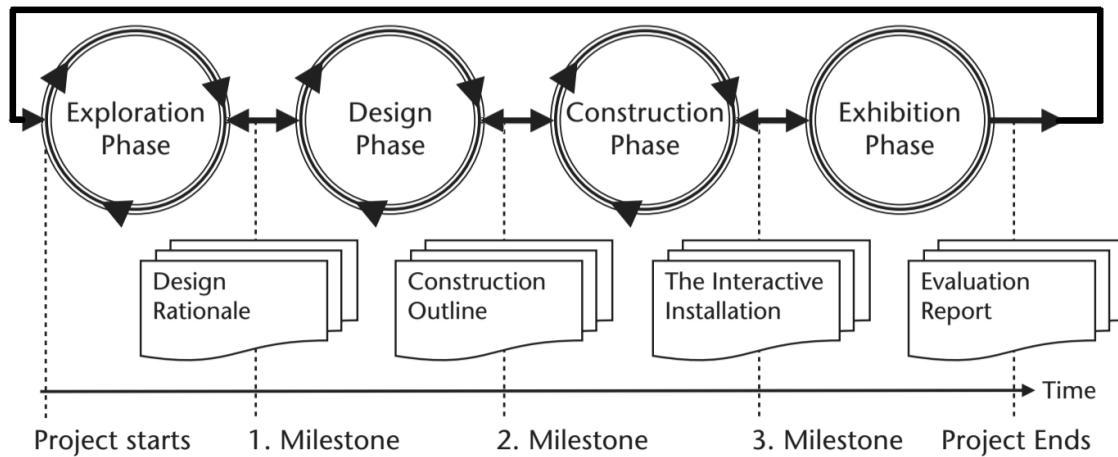
Larman (2004, p. 58)

Larmans model passer bedre til den designproces vi har været igennem, da vi har været nødt til at være fleksible og agile i forhold til krav og muligheder. Fra start af vidste vi at vi ønskede at skabe et troværdigt undervands univers i vores *Unity* program, som samtidigt skulle skabe fordybelse for brugeren gennem *Immersion*. Vi vidste dog ikke fra start hvilken måde vi skulle gøre det. Havde vi anvendt vandfaldsmodellen ville vi skulle sætte os ned og planlægge alle de implementeringer vi ønskede, skabe en fornuftig tidshorisont og dernæst gå til konstruktionsfasen.

Havde vi gjort dette var vi, efter al sandsynlighed, endt i nogle af de problematikker som Larman beskriver, omhandlende uforudsete vanskeligheder. Det skal dog også nævnes i sammenhæng med dette, at problemet som vandfaldsmodellen introducerer, også var det problem vi løb ind i forbindelse med den sidste rapport. Der opstod mange uforudsete vanskeligheder i forbindelse med udarbejdelsen af vores *tracking*, og vi endte derfor i tidsnød da den tidlige rapport skulle udfør-

5.1. Design, Model og Proces

mes. I dette forløb har vi derfor anvendt den iterative og agile model, og vi vil derfor argumentere for at man kan omskrive den model vi har anvendt tidligere, til at være mere agil. Dette har vi derfor gjort i figur. 5.3



Figur 5.3: *Four Phased Design Method*

Vi har i denne reviderede udgave, af den designmodel vi anvendte i den forrige rapport (Barrett et al., 2016), givet mulighed for at man både kan gå frem og tilbage mellem faserne, og endeligt at forløbet kan starte forfra. På denne måde forholder modellen sig dels til at omhandle designet af en interaktiv installation, og samtidig integrerer den Larman's idéer om effektiviteten ved en mere agil model. Vores model kan derfor ses som et kortere forløb, hvor den endelige evalueringsrapport ender ud i en iteration, og dermed giver mulighed for at man køre forløbet igennem igen. Des videre skabes muligheden for at gå et skridt tilbage i faserne. Således at, hvis man støder ind i uforudsete problemer i konstruktionsfasen, kan gå tilbage til designfasen og dermed revidere ens design. På denne måde er det nemmere at konstruere, og styre uden om forhindringer, der gør designmæssige beslutninger alt for besværlige at konstruere. Denne nye reviderede model mangler dog nogle forbehold som Larman præsenterer omhandlende agile modeller. Larman skriver selv:

“Agile development methods usually apply timeboxed iterative and evolutionary development, employ adaptive planning, promote incremental delivery, and include other values and practices that encourage agility rapid and flexible responds to change”

Larman (2004, p. 69)

Vores reviderede model indeholder ikke *Timeboxing* altså konkrete tidsforløb, men derimod *Milestones*. Det vil sige, at modellen bruger en form for pejlemærker i form af *Milestones*, som signal for at en fase er gennemgået.

Hvad angår det egentlige formål ved at skabe en agil model, som bedre beskriver vores designproces, følger det samme princip som Larman præsenterer under afsnittet om principper inden for agile modeller (Larman, 2004, p. 70). Her indleder Larman med det som han kalder for *The Agile Manifesto and Principles*. Dette består af fire delpunkter der skal fremhæves i en mere agil designproces:

The Agile Manifesto and Principles

- › *Individuals and interactions - over processes and tools*

- › *Working Software - over comprehensive documentation*

- › *Customer collaboration - over contract negotiation*

- › *Responding to change - over following a plan*

Larman (2004, p. 70)

Alle disse fire dele af det agile manifest er også dele der passer ned over vores designproces. Dels har vi haft individet og interaktionen i fokus da vi, først og fremmest, designer en interaktiv installation til en bruger. Derudover har vi fordret et software der virker, således at vi har haft muligheden for teste og samle empiri. Hvad angår det tredje punkt har der ikke været nogen kontrakt eller konkret kunde til stede, men vores hypotetiske kunde er, i en vis forstand, museumsinspektøren *Andreas Kallmeyer Bloch*, der står for udvikling af det eksperimentarium,

5.2. Design Iterationer

som det er meningen at vores installationen skal indgå i. Han er selv marinarkæolog og vi har derfor anvendt ham løbende i designprocessen til at hjælpe med at skabe et troværdigt undervands univers.

Afslutningsvis har vi også forholdt os mere åbne overfor ændringer frem for at følge en plan, da vores formål med at kunne skrive denne opgave, følger at vi har noget vi kan teste på, for at kunne evaluere. Der har dermed været fokus på at kunne være agil, skabe *mock-ups* eller blot kunne illustrere implementationer, frem for at følge deadlines.

Ydermere følger gennemgangen af dette manifest, i kontrast til vores designproces, det formål at skabe en revideret model af vores designproces, på baggrund af Larmans udtalelse om formålet med modellering. Larman skriver selv:

“The purpose of modelling is primarily to understand not to document.”

Larman (2004, p. 72)

Det er netop også dette der er formålet med at skabe en model for vores designproces. I det kommende tekniske afsnit, hvor vi vil beskrive vores kode struktur eller hierarki i vores software, følger vi samme udtalelse fra Larman. Formålet med vores modeller er at skabe forståelse.

5.2 Design Iterationer

I denne del vil vi kort beskrive hvilke iterationer der er foretaget i forbindelse med designet af dette projekt. Iterationerne tager udgangspunkt i den periode der er forløbet siden vi afsluttede sidste projekt (Barrett et al., 2016), altså fra starten af dette projekt og til udarbejdelsen af denne rapport. Formålet med at vise de følgende iterationer er at skabe forståelse og overblik over de konkrete udarbejdelser, der er foretaget i forbindelse med projektet.

Iterationerne der er blevet udført i forbindelse med designet kategoriserer vi i tre faser. En fase indeholder derved op til flere iterationer, der er foretaget med henblik på den efterfølgende fases indhold af tests.

5.2.1 Fase 1.

Den første fase omhandler perioden fra den 1. september til den 1. november. Den 1. september er tidspunktet hvor projektet startede og den 1. november er tidspunktet hvor Vikingskibsmuseet fik besøg af skoleklasser, som et led af museets egen forundersøgelse.

- › Installationen er blevet beklædt med sort stof.
- › Antallet af kameraer til anvending af *tracking* er blevet fordoblet, for at forbedre *trackingen*.
- › Kameraerne er blevet skilt ad og monteret i laserskåret kabinetter, dels for at højne synsfeltet for kameraerne og dels for at implementere et *IR-Filter*.
- › IR-LED'en, som sender infrarødt lys til webkameraerne, blev installeret i et plastikrør, som er et faktisk rør der anvendes under marinarkæologiske udgravninger.
- › En af de samlede tre 3D modeller blev tilføjet til programmet.
- › Antallet af *vertices* på 3D modellen blev formindsket fra ca. 1.5 millioner til 100.000.

5.2.2 Fase 2.

Fase 2. omhandler perioden fra den 2. til den 14. november. Her fik Vikingskibsmuseet besøg af et hold Marinarkæologi studerende fra Syddansk Universitet. Der blev derfor i denne forbindelse udført følgende iterationer.

- › Implementering af to yderligere 3D modellerede objekter og formindskning af *vertices* på disse.
- › Forbedring af *CreateBody* scriptet Se bilag E.8, s. 148 , der hjælper til objekternes opførsel i programmet, når objektet er frigjort af sand og dermed opdaget.

5.2.3 Fase 3.

Fase 3 forholder sig til den periode der er forløbet fra den 15. november til den 15. december. Her foretog vi en brugertest med antropolog og formidler på Vikingskibsmuseet, Niclas Wathne, samt en test af en uindviet bruger. I forbindelse med denne periode, vores tests og evalueringer blev følgende iterationer foretaget.

5.2. Design Iterationer

- › Plastikrøret som agerer sug er flyttet fra bunden til toppen af rammen af installationen, for at forbedre evnen til at flytte røret over skærmen.
- › Implementeringen og udviklingen af en touchskærm, med *Site Finder*.
- › Implementeringen af FishFlock algoritmen.
- › implementering af infomonitor
- › optimering af *tracking* kode for mere responsiv *tracking*
- › start animation af site for mere immersiveness

Kapitel 6

Teknisk Afsnit

Dette kapitel tager udgangspunkt i Datalogi og Informatik delen i dette projekt. Afsnittet vil beskrive vores software og hardware individuelt, samt gå i dybden med de udvidelser vi har foretaget, i løbet af semesteret. Afsnittet indeholder en generel software beskrivelse, et indblik i vores anvendte udviklingsmønstre, modeller, touchskærm, animationer og tracking.

6.1 Generel Software Beskrivelse

Den udviklede software har til formål at give brugeren oplevelsen af at være i den situation en marinarkæolog er i, under en marinarkæologisk udgravning.

For at opnå dette var vi nødsaget til at finde en måde at skabe en dynamisk *sandbunke*, hvori man kan grave. Da sand i den virkelige verden består af en stor mængde små partikler (i form af sandkorn) er det ikke muligt, med vores udstyr, at processere en direkte simulering af dette. Som vi beskrev mere dybdegående i den tidligere rapport (Barrett et al., 2016, p. 44).

Som alternativ til dette vendte vi opmærksomheden mod en 3D teknologi kaldet voxels. Voxels er en måde at repræsentere pixels, i et 3D miljø. Et forholdsvis velkendt program der er bygget på voxels er spillet minecraft. Gameplayet i dette spil består, som sådan, af at bygge og grave, med nogle forholdsvis store kvadrater der kan tilføjes eller fjernes.

Med den voxel teknologi spillet Minecraft benytter, er udseendet meget kantet og på ingen måde realistisk nok til at repræsentere sand på en troværdig måde.

6.1. Generel Software Beskrivelse

Der findes dog en teknologi ved navn triangulering (Flick), der kan benyttes til at skære hjørnerne af de kvadratiske voxels, således at de yderste voxels vil opnå en 'blød' overflade, der ved hjælp af *textures* kan komme til at ligne rigtigt sand forholdsvis godt. Da generering af voxels er en forholdsvis kompleks teknologi, og ville kræve en større arbejdsbyrde end vi har til rådighed, specielt hvis man har behov for triangulering, har vi derfor valgt at benytte et opensource bibliotek ved navn Qubiquity (Williams) til dette.

Som nævnt tidligere i rapporten, har denne mere indlevelsesrige oplevelse været et mål i løbet af dette semester. For at give brugeren denne mere fordybende og troværdige oplevelse, har vi tilføjet en række software elementer der ikke direkte udvider funktionaliteten, men i stedet udvider muligheden for indlevelse og fordybelse for brugeren.

Vi har benyttet forskellige teknikker til at opnå dette. Nogle af de større udvidelser er blandt andet en *Swarm Intelligence* algoritme, der styrer et antal fisk, som bliver påvirket af deres omgivelser og hinanden. Derudover har vi også tilføjet et antal animationer, så som overgang fra over vandet til under vandet, semirealistisk vand med bølger, skiftende strøm under vandet visualiseret med ophvirvlet sand og fysisk påvirkning af objekter. Disse udvidelser vil blive beskrevet senere i dette afsnit, men indledningsvis vil vi beskue det udviklingsmønster vi har anvendt i forbindelse med udarbejdelsen af vores software.

6.1.1 Unity, Spil/Interaktive Installationer & Entity Component Systemet

Den software vi har udviklet til vores installation forsøger, som sagt, at gengive en virkelig situation, så troværdigt som muligt. For at gøre dette kræves der forskellige elementer som et landskab, objekter, fysik simulering og lignende. På den måde er der en forholdsvis stor lighed mellem denne type software, og 3D computerspil. Den mest væsentlige forskel mellem 3D spil og vores software er, at vi ikke har implementeret spilleregler og et generelt mål for hvornår en spiller har vundet, eller på en anden måde er færdig med spillet.

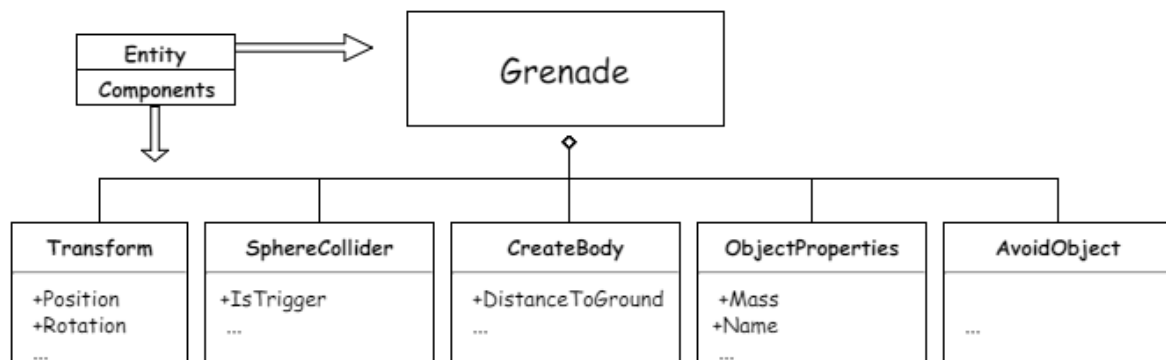
Med hensyn til den struktur den endelige software kan bygges op efter, hvad enten det er en simulering af noget virkeligt til et spil, eller til en interaktiv installation, findes der dog mange ligheder mellem de to former for software.

På den baggrund har vi valgt at benytte et udviklingsmønster ved navn *Entity*

Component System (ECS), til at udvikle vores software.

ECS blev benyttet første gang i 1998 til at udviklet et *FPS (First Person Shooter)* spil ved navn *Thief: The Dark Project* (Wikipedia, a), og blev dokumenteret og beskrevet som et egentligt udviklingsmønster i 2002 (Wikipedia, a) da det blev benyttet til at udvikle spillet *Dungeon Siege*. Senere hen er mønstret blevet anvendt, i en lettere modificeret udgave, som den anbefalede udviklingsprocess i spiludviklings miljøet *Unity* (Stagner, a), som er det udviklingsmiljø vi benytter.

Generelt er *ECS* strukturen en objekt orienteret arkitekturisk teknik. På samme måde som *Objekt Orienteret Programmering (OOP)*, har termen *ECS* ikke en veldefineret konsensus for hvad den egentligt dækker over (Stagner, b). Det er udviklet som et alternativ til en nedarvningsbaseret arkitektur. Som hovedregel kan man sige at en nedarvningsbaseret arkitektur beskriver hvad et objekt *er*, hvor et *ECS* beskriver hvad et objekt *kan/gør*. Diagrammet i figur 6.1 viser et eksempel på hvordan *ECS* hænger sammen, ud fra et faktisk eksempel i vores program.



Figur 6.1: *Entity Component Systemet*

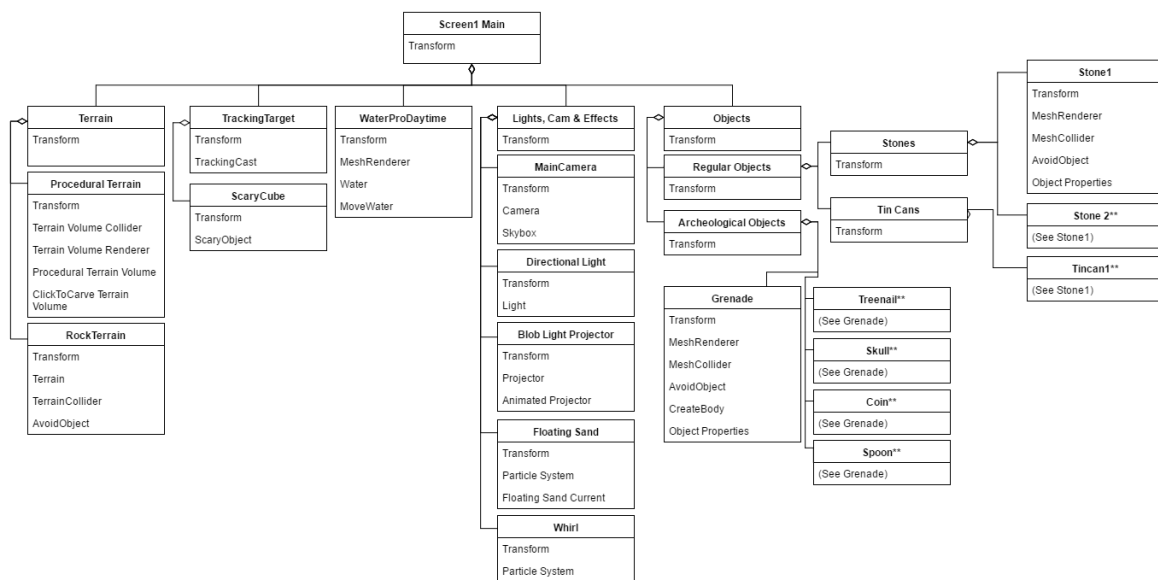
I *Unitys* version af et *Entity Component System* har de dog tilføjet et ekstra niveau i form af et hierarki. Det fungerer ved, at samtlige dele der er synlige, eller har en anden effekt i en såkaldt *Scene*, er entiteter eller, som *Unity* kalder det, *Gameobjects*. Det smarte ved et hierarki som dette er, at man har muligheden for at opbygge en 'trælignende' struktur. Et *GameObject* kan placeres som et *child object*, af et andet *GameObject*, hvilket åbner op for, eksempelvis at et *parent object* har et *Component Script* der påvirker samtlige *child objects*.

I sammenhæng giver *ECS* og *Unitys* hierarki mulighed for at skrive et program,

6.1. Generel Software Beskrivelse

der er modulært og deraf let at udvide. Ny funktionalitet kræver ikke omskrivning af eksisterende *scripts*, eller omstrukturering af hierakiets opsætning.

På baggrund af dette system har vi derfor lavet en model, der beskriver hvorledes vores software er inddelt, baseret på vores brug af *Unitys* 'Hierarki-ECS mønster'. Vi har valgt at lave et trædiagram (Se figur D.1) med et relativt højt abstraktionsniveau, for at bevare overskueligheden, da systemet efterhånden består af et forholdvis dybt hierarki. Vi har derfor udeladt at medtage metoder og variabler for de enkelte klasser/komponenter, og viser da kun deres navn. Diagrammet viser den del af hierakiet, der indeholder alt hvad der benyttes på den såkaldte hovedskærm - der hvor brugeren graver og finder ting.



Figur 6.2: Overskrifter svarer til entiteter eller *Gameobjects*, Indhold svarer til komponenter eller klasser. For et større diagram Se bilag D.1, s. 136

6.1.2 Multidisplay: Maindisplay, Site Finder og Infoskærm.

Som en del af den tekniske implementering vi har foretaget, har vi indsat to yderligere skærme til installationen (se figur 1.2 på s. 3). Installationen består derfor ikke længere kun af en hovedskærm (*Maindisplay*) hvor man kan grave, men vi har tilføjet to ekstra skærme med ny funktionalitet, der har til hensigt at øge oplevelsen og muligheden for fordybelse. Disse tilføjede skærme er henholdsvis vores *Site Finder* og en info skærm.

Maindisplayet har stadig sin hovedfunktionalitet. Det er på denne skærm man kan grave og interagere med havbunden. Vi har dog udvidet denne skærm med en række nye funktioner som eksempelvis fisk og animationer, som vil blive beskrevet senere i dette kapitel.

Site Finder Displayet er en ny tilføjelse, der giver mulighed for at vælge mellem sites, ud fra et kort over Danmark. Vi har dog ikke på nuværende tidspunkt implementeret mere end en et site, da dette også ville kræve flere reelle arkæologiske fund, scannet som 3D modeller og med information om hvor disse er fundet. Der er dog stadig mulighed for at brugeren får en følelse af rent faktisk at 'opleve' det site de har valgt. Se afsnit 6.1.4 for en mere uddybende beskrivelse af *Site Finderen*.

Info Skærmen er endnu en ny skærm med ny funktionalitet. Denne skærm har til formål at give information om de objekter man finder. Vi er ikke, i vores samarbejde med vikingeskibsmuseet, endnu blevet helt enige om hvor meget information brugeren egentlig skal have herfra. Dette skyldes at for meget information vil reducere den realistiske følelse, en marinarkæolog har, ved at finde noget ukendt. På nuværende tidspunkt har vi sørget for at de objekter man finder, der kan have arkæologisk interesse, bliver vist på skærmen og roterer så man kan se det fra alle vinkler.

6.1.3 Distributed Behavioral Model of a School of Fish

Dette delafsnit omhandler en såkaldt *Swarm Intelligence* algoritme, bygget som en distribueret adfærdsmodel, baseret på Craig W. Reynolds tre trins algoritme, beskrevet i hans artikel *Flocks, Herds and Schools: A Distributed Behavioral Model* (Reynolds, 1987)

Som en del af vores mål om at skabe et troværdigt undervands miljø, har vi anvendt den omtalte *Swarm Intelligence* algoritme, for at gøre den virtuelle sandbund mere levende. Craig W. Reynolds udtaler således også selv om hans algoritme:

6.1. Generel Software Beskrivelse

"The aggregate motion of a flock of birds, a herd of land animals, or a school of fish is a beautiful and familiar part of the natural world. But this type of complex motion is rarely seen in computer animation. This paper explores an approach based on simulation as an alternative to scripting the paths of each bird individually."

Reynolds (1987)

En distribueret adfærds model kan resultere i en opførsel, der ligger forholdsvis tæt op ad den, som findes i naturlige flokke. Hvert medlem af flokken (agent) vælger sin egen vej(distribueret). Hver agent tager beslutninger på baggrund af deres perception og omgivelser individuelt, og uden at kende noget til andre agents valg. Den enkelte agents perception, de adfærdsregler den agerer under, og de simulerede fysiske love - ender ud i en kompleks opførsel, baseret på de relativt simple regler, når de er fulgt af mange individuelle agenter.

Craig W. Reynolds skrev et program kaldet Boids (Ordspil på Bird-thing) (Reynolds, 1987) som implementerede algoritmen bestående af de tre følgende regler.

- › **Alignment:** Alle agenter skal bevæge sig i en fælles retning.
- › **Cohesion:** Alle agenter skal bevæge sig mod gruppens gennemsnitlige centrum.
- › **Seperation:** Alle agenter skal bevæge sig væk fra hinanden hvis de kommer for tæt, for at undgå overlap.

Vores implementering er som sagt baseret på Reynolds algoritme, men er derudover udvidet med nogle ekstra regler, for at opnå en mere divers og realistisk opførsel.

En ekstra regel vi har sat op, er at fiskene har et slags *energi niveau* (i form af et *exhaustion level*), der daler i takt med at fiskene svømmer rundt. Hvis fiskene ikke har meget energi tilbage, vil de søge nedad og svømme langsomt langs en overflade - medmindre de er alene. I så fald vil de søge efter en gruppe. Fiskene kan, på denne måde, kun 'slappe af' i grupper.

Derudover var det meget oplagt, i vores installation, at fiskene også kunne blive

bange og stikke af i høj fart. Dette fungerer ved, at et hvilket som helst objekt i programmet, kan få tilføjet et script ved navn *ScaryObject*. Fiskene vil da genkende det og stikke af fra det. I vores tilfælde er det skræmmende objekt selve suget, der bruges til at suge sand op med.

Da en algoritme som denne hurtigt kan blive uoverskuelig, grundet de mange beslutninger, hver fisk skal tage hver frame, har vi implementeret en såkaldt *Finite State Machine (FSM)* Wikipedia (b). En *FSM* er en teknik, eller en software baseret maskine, der kan køre i et endeligt, (*finite*), antal tilstande (*states*), og kun en af gangen. Generelt når man udvikler software, der skal kunne køre i en ikke forud-defineret periode og have skiftende output, vil der være en form for loop, der køres hver frame og ændrer forskellige tilstande i programmet. Det er gennem dette loop de enkelte fisk får alt deres opførsel. Da fisk kan være i mere end én tilstand, dog kun en tilstand af gangen, og opfører sig forskelligt på baggrund af dette, er det en stor fordel at implementere en *FSM*, for at opdele koden og dermed gøre den langt mere læsbar. Koden herunder viser hvordan vores *FSM* er implementeret. I det her tilfælde er variabelen *States* af typen *Enum* (*Enum* er en samling af en type navngivne værdier), og indeholder samtlige tilstande.

```
switch (state)
{
    //State 1: Flocking
    case States.flocking:
        FlockingUpdate();
        break;
    //State 2: Resting
    case States.resting:
        RestingUpdate();
        break;
    //State 3: Playful
    case States.playing:
        PlayingUpdate();
        break;
}
```

Herunder ses et overblik over de regler, der styrer fiskene. Denne overblikliste blev skrevet som en guide, da vi designede vores implementering, og udvidelse af algoritmen. Guiden er inkluderet her, da den beskriver funktionaliteten forholdsvis simpelt og præcist.

› All States:

6.1. General Software Beskrivelse

I. Avoidance Rules.

- a. A fish flee's in the opposite direction when a scary object comes within range from any side.
- b. A fish turns away from an avoid object when it gets close (Could be oil, floating sand or other non-scary unpleasant objects)

› **State 1: Flocking** (The base state for a fish)

I. Classic Flocking Rules.

- a. Alignment: All fish must move towards the common direction of the group.
- b. Cohesion: All fish must move towards the common centre of the group.
- c. Separation: All fish must steer away from other fish within a given range.

II. Additional Flocking Rules.

- a. There should be a repositioning goal position for each group.
- b. A fish should always stay within the given area.
- c. When in flocking state, but not currently in a group (moving alone), move faster to increase the chance of meeting a group.
- d. When flocking the group should not swim too much along the edges of the area, since this seems unnatural.
- e. When approaching a surface, sometimes alter direction to move along the surface.
- f. Sometimes when looking straight at another fish, switch to its group

› **State 2: Resting** (Lying still, or moving slowly) (Entered from: Flocking)

I. *Entered: After a set amount of exhaustion is reached*

- a. a. Search for a place with good resting conditions.
- b. a. Move slowly with intervals of lying still.

De følgende tilstande (*states*) er ikke implementeret i vores software på nuværende tidspunkt, men vil på sigt være relativt nemme at implementere via den benyttede *State Machine* og struktur.

› **State 3: Playful** (Interacting in a playful way with one or more other playful fish)

I. Entered: After a set amount of playfulness is reached

- a. Move quickly towards other fish.
- b. Chase other fish.
- c. Run away from other fish

› **State 4: Curious** (Stay close to and circle around an interesting object)

› **State 5: Hungry** (Look for food/hunt)

6.1.4 Site Finder & Touch

Se figur 6.3 En af de udvidelser vi har implementeret på dette semester, som både har krævet ny hardware og software er en, som vi selv kalder den, *Site Finder*. Det fungerer ved at når brugeren nærmer sig installationen, mødes vedkommende af en mindre touchskærm, der viser et gammelt kort. Se figur 6.3 tv. På denne skærm kan brugeren, ved at bruge en, to eller tre fingre - navigere, zoome ind eller zoome ud. Dette virker umiddelbart relativt godt, og er utroligt nemt at gøre - når man ved hvordan. Det har dog vist sig ved tests, at folk pr. automatik forsøger sig med det sædvanlige *pinch* zoom, som er kendt fra tablets og smartphones. Det vil gøre den en smule mere besværlig at bruge, men da de fleste kun vil prøve en installation som denne en enkelt gang, er *pinch* zoom muligvis en mere intuitiv funktionalitet.

Der vises på kortet nogle små *knappenåle*, der svarer til individuelle sites. Brugeren kan trykke på disse knappenåle og vil derefter blive præsenteret med startanimationen til selve grave delen af programmet. Se afsnit 6.1.5



Figur 6.3: Site Finderen set uden zoom tv. og med zoom th.

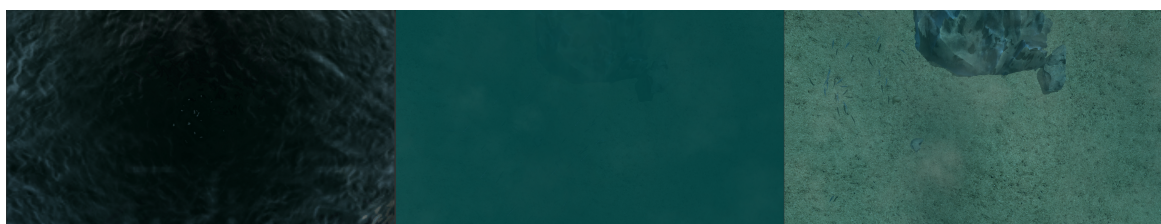
6.1. Generel Software Beskrivelse

Denne skærm fungerer, som sagt, på nuværende tidspunkt, ved at brugeren navigerer og zoomer ved hjælp af multitouch. Det viste sig dog at være en større udfordring at få dette til at fungere i vores miljø, *Unity*. Touch virker på forskellige måder, hvad enten man benytter et touch device som en smartphone/tablet eller en computer. Som standard vil en computer ikke se det som touch, men som en musemarkør, og dette er selvfølgelig ikke kompatibelt med multitouch, da en computer kun har en musemarkør af gangen. Normalt kan man, på en windows computer, løse dette ved at bruge microsofts eget bibliotek *windows.h* til at forstå touch input som touch. Men da *Unity* er bygget til at være multiplatform, findes sådanne Windows specifikke elementer ikke, og *Unity* har ikke på nuværende tidspunkt implementeret et alternativ.

6.1.5 Animation

I dette delafsnit vil vi beskrive nogle af de animationer vi har anvendt for at give en mere glidende overgang, mellem vores *scenes* og en mere livagtig aktivitet, mens programmet kører, har vi tilføjet en række animationer.

Den mest omfattende animation er den der kører i starten af selve 'grave' delen af vores applikation. Den fungerer ved, at når en bruger har valgt et site at undersøge, vises en vandoverflade på hovedskærmen. Efter et par sekunder begynder det virtuelle kamera at bevæge sig ned ad mod skærmen, for derefter at bryde med vandoverfladen og der vises i stedet noget uigennemsigtigt vand. Kameraet fortsætter lidt tid med at bevæge sig mod bunden, som langsomt kommer til syne, i takt med at vandet bliver mere gennemsigtigt. Det er meningen at dette skal føje til følelsen af at man er under vandet. Se figur 6.4.



Figur 6.4: Screenshots af start animationen, fra start til slut.

En anden animation vi har arbejdet på er visualisering af strøm under vandet. I den første udgave af denne installation skrev vi et *Current Script*, der påvirker

samtlige objekter ved kontinuerligt at tilføje kræfter, i svingende retninger. Vi har nu udvidet dette til også at styre en partikelgenerator, der simulerer en slags op-hvirvlet sand. Det ophvirvlede sand bevæger sig nu i samme retning som *Current Scriptet*, der skubber objekterne.

6.2 IR Tracking

IR trackingen er en forbedret udgave af det, der blev benyttet i sidste projekt (Barrett et al., 2016) og derfor vil følgende være en delvis gengivelse af dette, samt de forbedringer, der er foretaget i det nuværende projekt.

IR trackingens positioneringens basale virkemåde beskrev vi også i vores tidligere rapport, og IR trackingen følger også samme funktionalitet. Kort beskrevet, udføres trackingen ved at sende et IR lys fra slangen ned mod skærmen, med den viste havbund. Under skærmen befinder sig fire webkameraer, med påsatte IR filtre. Yderligere specifikationer omhandlende dette vil blive gennemgået senere i afsnittet. De fire kameraer opfanger IR lyset, og på denne måde kan vi tracke hvor slangen befinder sig henne, over skærmen. Den konkrete positionering af slangen dannes ved at omdanne det modtagne IR lys til en *blob*. Altså en plet. Til dette anvender vi et bibliotek ved navn *OpenCV* til at udvikle vores *blob detection* (Bradski). I vores tidligere rapport beskrev vi således også om *blob detection*.

Blob detection

Da der benyttes samme *blob detection* kode som i foregående projekt, er her en kort gennemgang:

“Programmet omdanner derefter denne mængde pixels til en “blob”, altså en plet, og tegner en firkant herom. Denne firkants x og y koordinat samt størrelse mappes fra processings koordinater til nogle der giver mening med Unitys units, og sendes til vores Unity software pakket som OSC, via en UDP protokol.”

Barrett et al. (2016, p. 42)

OpenCV biblioteket *Blob Detection* kan ses i bilaget Se bilag E.33, s. 207 og vores anvendelse af dette kan ses i bilaget Se bilag E.34, s. 209 .

6.2. IR Tracking

OSCListener

Den førnævnte kommunikationen med *Unity* foregår, ligesom i sidste projekt, via *OSC (Open Sound Controller)*. En kort gennemgang med udgangspunkt i foregående projekt blev beskrevet som følger:

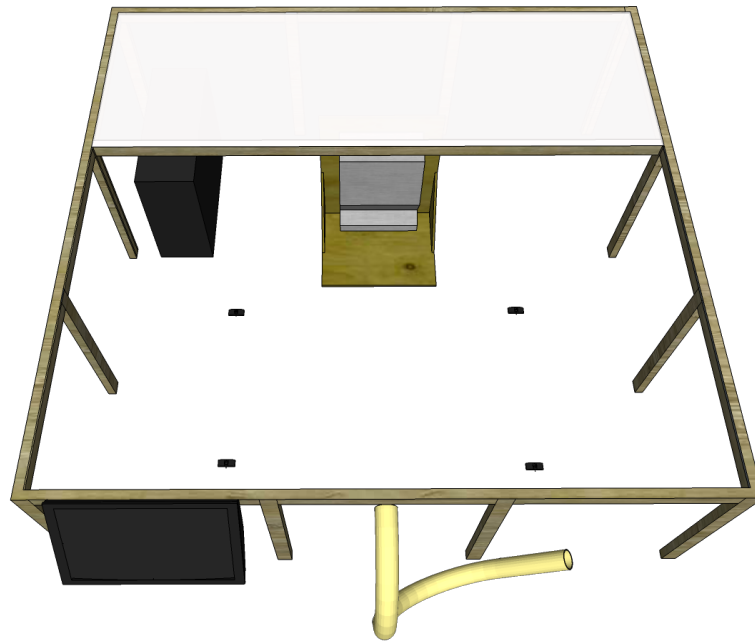
“Kommunikationen mellem Unity og Processing fungerer ved at blive pakket som OSC(Open Sound Control) pakker, og sendt via localhost som UDP (User Datagram Packets[28]) til Unity.”

Barrett et al. (2016, p. 45)

6.2.1 Den Forbedrede Tracking

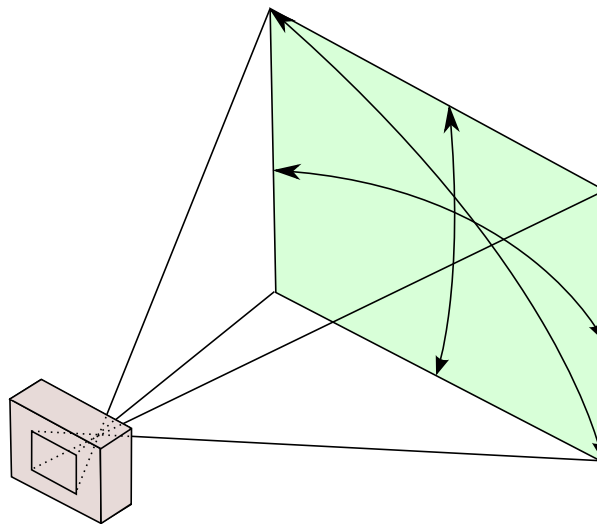
Årsagen til at *trackingen* forbedres, tager udgangspunkt i, at den havde 'døde' områder i ekstremerne og midten, hvor feedet bliver samlet. Dette skyldtes at måden *trackingen* var løst på, var ved hjælp af nogen alt for kraftige vidvinkelslinser, der forvrængede billedet i ekstremerne og gjorde det svært at aflæse det infrarøde signal.

For at modvirke dette er der blevet monteret to yderligere kameraer, figur 6.5 på s. 51 . De fire sorte firkanter, der ses indefor rammen er kamera opstillingen. Disse gør tilsammen at hele fladen dækkes, uden brug af vidvinkelslinser og hermed undgås de førnævnte problemer. To af kameraerne er genbrug fra sidste projekt og er sammen med de to tilkomne kameraer, blevet skilt ad og har fået fjernet deres *IR-cutoff* filtre, så denne del af spektrummet kan registreres. For at filtrere synligt lys fra, placeres IR filtre foran kameraerne i hver kamerakasse, hvor disse passer ned i og reflektioner undgås.



Figur 6.5: Installation med fire kameraer installeret.

Det specifikke kamera der gøres brug af er - Logitech C525 - de har et *Field Of View* (FOV) på 69° .



Figur 6.6: Illustration af Field Of View. (Wikipedia, the free encyclopedia)

6.2. IR Tracking

Et *FOV* er defineret som den vinkel et givent objektiv, eller linse, producerer. Dette giver over en given afstand, en billedbredde. Man har en konsensus om at det som udgangspunkt altid er billedbredden, der refereres til. Se figur 6.6 på s. 51 .

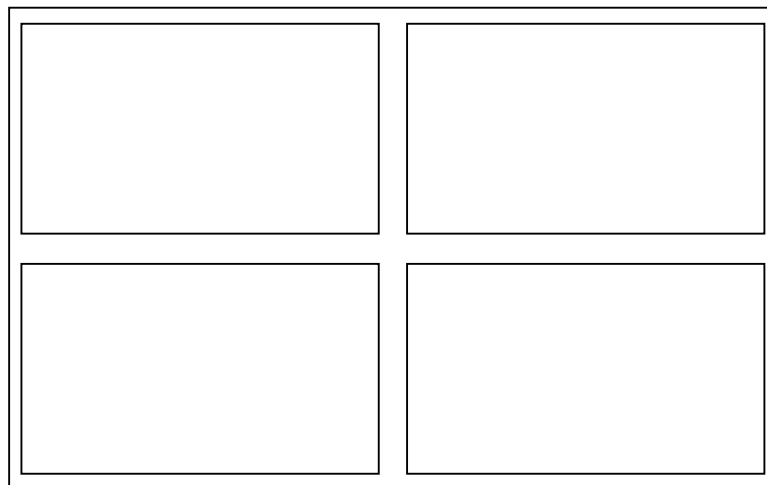
Følgende er beregninger, for hvilken afstand fra pladen til kameraet og *FOV* der skal benyttes, for at opnå fuld dækning for *trackingen*. Den nuværende opsætning med en afstand på 69 cm, giver følgende billedbredde for hvert kamera:

$$Bredde = 2 \cdot (Afstand \cdot \tan\left(\frac{FOV}{2}\right)) = 2 \cdot (69cm \cdot \tan\left(\frac{69^\circ}{2}\right)) = 95cm \quad (6.1)$$

Ud fra dette kan billedhøjden beregnes, da billedformattet er 16/9:

$$Højde = Bredden \cdot \frac{9}{16} = 95cm \cdot \frac{9}{16} = 56cm \quad (6.2)$$

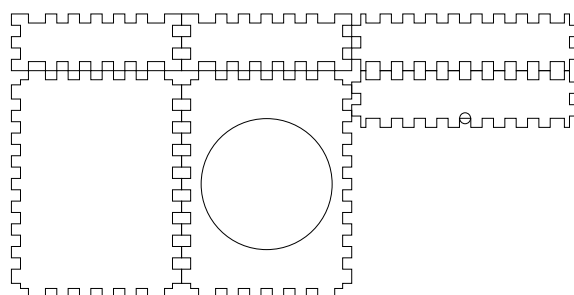
Fladen, de fire kameraer skal dække, er 2,03 m bredt og 1,27 m højt, hvilket giver følgende dilemma:



Figur 6.7: Kameraernes synsfelt for pladen.(Wikipedia, the free encyclopedia)

Dette resulterer i, som illustreret i (figur 6.7 på s. 52), at vi har et område (krydset i midten og rundt om) som ikke kan se IR LED'en. Effekten er ikke så tydelig som dette måske kunne lede til at tro, da pletten har en vis størrelse, der gør at den bliver set, omend ikke lige så godt i det pågældende område. Dette undgås, hvis kameraets synsfelts bredde ændres til 102 cm.

For at kameraerne har den største afstand til pladen som muligt, er de blevet, som tidligere nævnt, skilt ad og monteret i et lasercuttet kabinet for at få kameraet så langt ned mod gulvet som muligt - se (figur 6.8 på s. 53) for planerne af de nye kabinetter. Kabinetterne måler $b \times h \times d$, 75 mm x 25 mm x 100 mm og er konstrueret i 4 mm HDF, dvs. at kameraet nu i stedet sidder ca. 8 mm over gulvet.



Figur 6.8: Lasercutter planer for nyt kabinet til kamera.

Der er to måder hvorpå vi kan eliminere disse *døde* områder. Navnligt ved at forøge afstanden yderligere eller benytte et kamera med et bredere *FOV*. Med det nuværende *FOV* på 69° skal der, for at opnå en billedbredde på 102 cm, skal afstanden være:

$$Afstand = \frac{Bredden/2}{\tan\left(\frac{69^\circ}{2}\right)} = 74cm \quad (6.3)$$

Altså skal bordet hæves minimum 5 cm.

Alternativt skal et andet kamera med et højere *FOV* benyttes, i så fald skal denne være større end 73° :

$$\tan^{-1}\left(\frac{FOV^\circ}{2}\right) = \frac{Bredden}{2 \cdot Afstand} = \frac{102cm}{2 \cdot 69cm} = 0,74, \tan^{-1}(0,74) = 36,5^\circ \cdot 2 = 73^\circ \quad (6.4)$$

At dømme ud fra Niclas Wathnes udtalelse vedrørende installationshøjde ville et kamera med et større *FOV* være at foretrække:

“Så er det altså en rigtig god højde. Specielt for ungerne (skolebørn som testede), når de skal stå og støvsuge. Og det var endda udskoling ungerne, altså 7., 8., og 9. klasse. Hvis man skal have nogle mindre børn herind, så kan man selvfølgelig stille dem op på et eller andet, men højden er altså ret god.”

Se bilag C.3, s. 98

6.2. IR Tracking

Billedbehandling

Undervejs i projektet blev der brugt en del *remapping* og funktioner, der hjalp visuelt med *troubleshooting*, da *trackingen* var under udarbejdelse. Dette resulterede i en framerate på ca. 9 og medvirkede til en ikke særlig responsiv *tracking*, der var med til skabe distance mellem brugeren og systemet. Efter at *IRtracking.pde* (Se bilag E.34, s. 209), blev optimeret, gennem oprydning af alle unødvendige funktioner og beregninger, blev framerateen løftet til ca. 50. På denne måde blev den mere end hurtig nok, da kameraerne ikke kan levere billederne hurtigere end 30 frames per sekund.

Af optimering kan nævnes:

- › Unødig mapping fjernet
- › Den eneste billedbehandling, der foretages er at sætte Threshold
- › Kamera-feedet bliver ikke resizet til at akkommodere mainskærmens opløsning

Kapitel 7

Analyse & Diskussion

7.1 Analyse

I analysen forholder vi os til den empiri vi har indsamlet gennem vores semi-strukturerede og ekspertinterviews. På baggrund af den viden vi har udarbejdet gennem det teoretiske afsnit og måden vi har indsamlet empiri på, vil vi her beskrive hvad vi kan sige om betydningen af de tekniske virkemidler, i forhold til installationens evne, til at skabe troværdighed og fordybelse, i dets forsøg på at formidle om marinarkæologi.

I det teoretiske afsnit så vi på, hvad folk associerer mest med arkæologi. Her svarede størstedelen at den del af det opdagende og eventyrlige element ved arkæologien, som de finder mest interessant er at grave. Dette forholdte museumsinspektør Andreas Kallmeyer Bloch sig også til, i forbindelse med vores ekspertinterview.

"Jeg tror at det der med, at folk som udgangspunkt har svært ved at forestille sig hvordan vi graver under vand. På den måde lærer folk noget om vores metode. Så forhåbentlig når man så tager sin genstand videre, så lærer man også noget om selve genstandene, men lige præcis her er det jo vores metode, at vi graver med et sug."

Se bilag C.5, s. 127

Ydermere kiggede vi på hvordan arkæologer er afhængige af samfundet. Dette indebærer at offentligheden forbliver interesseret i arkæologien samt de genstande de finder, for at faget kan overleve. Denne afhængighed af samfundet, kommenterede Andreas også på:

7.1. Analyse

"(...) det heldige ved vores det er at det ligger i museums hjørnet og det er en statslig pligt, at hvis man finder noget der er et fortidsminde, at man så skal undersøge det. (...) det er jo den her økonomi der er det springende punkt om vi får lov til, at blive ved med at gøre det vi gør. Hvis ikke vi viser det i udstillinger, så er det helt sikkert vi mister den opbakning der kan være både fra befolkningen og staten."

Se bilag C.5, s. 133

Ifølge den empiri vi indsamlede i forbindelse med vores tests, forholder vores installation sig til det opdagende element i arkæologien - at grave. Altså at gå på opdagelse.

Vores installations evne til at skabe fordybelse følger derfor også, at når brugere anvender vores installation, så skal det opdagende element svare til arkæologiens, for at skabe troværdighed og formidle noget om marinarkæologi. I forbindelse med de tests foretaget under besøget af skoleklasserne udtalte Niclas Wathne følgende:

"Med hensyn til selve undervisningsforløbet, og det flow der skal være, så var noget af det gode herovre (Peger på vores installation), det var at de faktisk ikke helt vidste hvad de fandt. Altså for eksempel det her skjold, som var et låg. De vidste ikke hvad det var da de gik herfra, og det er ligesom det der pirrede dem til at gå derover (peger på en af de andre installationer), og prøve at finde ud af hvad det egentlig var. Ligesom en arkæolog gør."

Se bilag C.3, s. 96

Fordi vi har skabt et troværdigt miljø, hvor brugeren, i dette tilfælde skoleklasserne, ikke altid kunne gennemskue hvilken genstand de havde fundet, ligesom en arkæolog heller ikke altid kan, pirrede det dem til at vide mere. Lignende beskrev vi også i teorien, hvorledes der er et behov for en overensstemmelse, i forhold til ægthed til de objekter/genstande man beskuer på et museum. Det er altså dels vigtigt at de genstande man finder i installationen indeholder en vis ægthed, for at kunne formidle om det fag de forholder sig til. Dette er tilfældet i vores installation da de genstande, der befinder sig i vores installation er 3D scannede objekter af ægte genstande. Betydningen af dette beskrev på baggrund af et behov, ifølge (Bayne et al., 2009) og (Holtorf, 2007) for museer og arkæologi om at forny sig selv.

Des videre så vi hvordan der er sket en ændring i forhold til definition af digitale

objekter, og hvorledes den fysiske verden er blevet mere digitaliseret. Museer tilbyder en unik læringsoplevelse, hvor førstehåndsoplevelser med autentiske objekter, giver belæg for læringsoplevelsen. Der er derfor stadig behov for det egentlige objekt, men interagering med et objekt og muligheden for dette, højner læringsoplevelsen og på denne måde også muligheden for fordybelse. I forbindelse med interviewet med Niclas Wathne, hvor vi siden undersøgelseerne med skoleklasserne, havde indført en ekstra skærm, som viste den arkæologiske genstand man havde fundet, opdagede vi betydningen af dette. Niclas udtalte som følger om betydningen af denne ekstra skærm:

“Men det er rigtigt fedt at den kommer op når man finder noget. For noget af det som ungerne (skoleklasserne som testede) også nævnte her, var at de ikke helt vidste, da de fandt dåsen, om det var noget de skulle bruge til noget. Der er det der (ekstra skærm), jo en ret god ide. En indikator for “Hvad er det egentlig der er vigtigt?”, BANG, så kommer det op på skærmen. Så det er en rigtigt god ide.”

Se bilag C.3, s. 98

Tidligere beskrev vi hvordan vi som mennesker, forstår os gennem den fysiske verden vi befinder os i, og et værktøj kan ses som en forlængelse af den handling den anvendes til. Med dette i mente kan vi, på baggrund af vores interviews og tests, fastslå at en bruger af installationen automatisk vil opnå en eller anden form for fordybelse, i den blotte interaktion med denne. I henhold til dette spurgte vi også, i forbindelse med vores interview med Niclas, betydningen af interaktion med installationen. Til dette svarede han, på baggrund af den efterundersøgelse han foretog med skoleklasserne, hvor dele af *trackingen* ikke fungerede optimalt, og dets visuelle output ikke stemte perfekt overens med brugerens input (førslen af slangen hen over bordet) følgende:

(...)der forstyrrede det, fordi man blev i tvivl om den virkede, eller om man gjorde det rigtigt. Det var også det som ungerne fandt ud af meget hurtigt, specielt dem som var meget interesserede i teknik, computere og telefoner og sådan noget. De (Her mimer han at han slår slangen ned i bordet), og så virkede den ikke. Så gik de hurtigt rundt, og så skete der ikke noget kunne de se, så smed de suget, og sagde “Så virker det ikke”. For det kunne i hvert fald ikke passe at de ikke kunne finde ud af det.

Se bilag C.3, s. 99

7.2. Diskussion

Yderligere svarede Mette Jehrbo, i forbindelse med den del af hendes interview, der omhandlede installationens evne til at skabe en indlevelses evne hos hende:

(...) det der med, at jeg ikke automatisk ved hvor langt den skal holdes fra, der støvsugede jeg meget længe, og tænkte hvorfor sker der ikke en fis. Der havde jeg nok mistet tålmodigheden og gået min vej. Og tænkte, jeg gider ikke.

Se bilag C.4, s. 121

Det visuelle output betyder dermed utroligt meget for brugeren. Selvom man har skabt et troværdigt univers, med mange brugbare virkemidler, skal der altså være en optimal balance mellem det visuelle output og det tekniske input, for at holde brugeren interesseret. Hvis vores installation opnår en ønsket funktionalitet vil det højne fordybelsen, da det først er ved uoverensstemmelse mellem input og output, at der opstår problemer med fordybelsen.

7.2 Diskussion

I diskussionen forholder vi os til de emner vi har afklaret i analysen, ved at beskue dem i forhold til vores problemdeling som følger herefter.

- › *Hvordan designer vi en interaktiv installation?*
- › *Hvordan formår den interaktive installation at formidle troværdigt om en arkæologisk udgravning under vandet?*

Indledningsvis vil vi forholde os til det første delspørgsmål, nemlig hvordan vi designer. I designafsnittet kiggede vi på betydningen af at være agil i sin designproces, for at opnå et bedre resultat. I forbindelse med de omtalte undersøgelser, spurgte vi således også ind til implementeringen af en *Site Finder*, for at give brugere valgmulighed mellem forskellige udgravningssteder. Det var tanken at det kunne hjælpe til at gøre opdagelses-oplevelsen mere personlig, og samtidigt formidle om marinarkæologi. Til dette udtalte Niclas Wathne således også:

"Det at man kan vælge er jo vildt fedt. Det er ikke bare et eller andet fiktivt sted under vand. Det her er faktisk noget man prøver at genskabe, og der er endda forskellige vrage."

Se bilag C.3, s. 101

Vi har således også, på baggrund af nogle de elementer vi diskuterede i designafsnittet, haft i mente hvem det er vi designer til og hvad vores design handler om,

for på denne måde at kunne formidle om marinarkæologi. Derfor omhandlede mange af de undersøgende delspørgsmål, vi stillede under de forskellige tests og interviews, installationens evne til at skabe et troværdigt univers. For at kunne formidle om en marinarkæologisk udgravning, skal vi, på en troværdig måde, kunne føre folk med under vand ved hjælp af teknologi. Oftest vil den almene bruger af installationen ikke vide noget på forhånd, om marinarkæologi. Man skal derfor, når man designer, både forhøre sig med eksperter, der ved hvorledes det er og føles, men samtidigt huske på at den almene bruger ikke ved det. Denne diskurs har vi derfor forsøgt at indføre ved implementeringen af de forskellige animationer, blandt andet vores fisk, vand overflade, fysiske love samt 3D scannede objekter af virkelige genstande. Under besøget af en gruppe marinarkæologi studerende fra Syddansk Universitet blev der således udtalt følgende:

"It does spark the sort of intrigue and excitement. You get like, oh my god, i'm finding something. But also in the water, you have constant noise of breathing. It's totally different than 'just' this. It does, you do get a bit of excitement like oh my god. There is something, i can see something. That, that's fun, and it does make it interesting. That's what you get out of it."

Se bilag C.2, s. 92

Alice, som her svarer på et spørgsmål om hvor relevant vores installation er, i forhold til en virkelig marinarkæologisk udgravning, forholder sig netop også til dette. Havde vi forsøgt at gøre installationen fuldstændig overensstemmende med en virkelig marinarkæologisk udgravning, havde vi efter al sandsynlighed mistet noget af den oplevelse vi forsøger at give folk. Som Alice kommer ind på her, er den virkelige oplevelse ikke ens med den verden vi forsøger at skabe, da man, under vand, stort set intet kan se og meget lidt kan høre. Vi skal derfor med hensyn til troværdigheden i vores installation, forsøge at være mediator. Vi skal skabe et univers hvor vi kan give oplevelsen og fornemmelsen, af at agere under vand, som en arkæolog, uden at man er det. Warren, som også er studerende marinarkæolog, udtalte også i forhold til installationens evne til at skabe troværdighed, om en virkelig marinarkæologisk udgravning:

"I think, definitely yeah. When it comes to like when you're excavating, it's pretty accurate. It's as close as you can get, to simulate it on land i would say."

Se bilag C.1, s. 85

Ydermere skal det også siges at vi, i besøget om at skabe troværdighed om en marinarkæologisk udgravning, kun fortæller en lille del af historien. Vi fokuse-

7.2. Diskussion

rer, som designere, kun på det vi opfatter som det spændende, opdagende og oplevelsesrige element. Andreas Kallmeyer Bloch beskriver således også, om troværdigheden ved vores installation, i forhold til den historie, som er det samlede marinarkæologiske arbejde ved en udgravning:

(...)Det kræver jo at vi kigger site-scan data igennem, og vi kigger på topografien og kan så se ud fra topografien hvor der er størst sandsynlighed for at vi har en boplads eller et vrug eller hvad det er vi har. Derefter så har vi nogle yderligere rekognosceringer og så er det først det sidste trin der er at vi suger, så noget af det sidste vi gør. Men det ville fylde for meget som det er nu at have det med, det ville være for stor en mundfuld for at nå frem til det. Og så ville det ende med en installation, man ville blive ved i 20 minutter for at finde sit (site). Jeg er ikke i tvivl om at det kan laves, men man ville simpelthen bruge for lang tid på det.

Se bilag C.5, s. 133

- › *Hvordan designer vi en interaktiv installation som gennem tekniske virkemidler skaber fordybelse om en arkæologisk udgravning under vandet?*

Hvad angår designet, har vi på baggrund af de ting vi undersøgte om folks indlevelsesevne, i mødet med teknologi, forsøgt at skabe fordybelse gennem det fænomen, der opstår, når en bruger interagerer med installationen. Vores installation har, til en vis grad, levet op til vores mål om at skabe fordybelse. Fordybelsen, er uhyre vigtig for at kunne fange brugerens opmærksomhed i det korte tidsrum brugeren skal interagere med installationen. Niclas Wathne forklarer således:

"Det er rigtigt stort. Man gør noget aktivt. Man er marinarkæolog. Man har den der støvsuger, og man står ikke og læser noget. Man står ikke og får passivt information i hovedet. Man er aktivt søgende efter noget og får noget op. Man føler det er lidt unikt at man står der og finder et eller andet, som man kan se hernede på bunden eller på skærmen. Det er ret stort, og høj indlevelsesevne synes jeg. I forbindelse med den her (vores installation), og i flowet af de her installationer, så er det jo, som vi også stod og snakkede om før, perfekt for at skabe basen for hvad man laver som marinarkæolog, og hvad man skal herinde. Man skal finde det der objekt, det ukendte nede i vandet, og arbejde med det. Det er jo rigtigt godt sat op her."

Se bilag C.3, s. 101

Vores installation, er ifølge dette citat ganske velfungerende. Men som det ser ud nu, skal der dog en introduktion til, for at man kan se hvorledes hele installationen fungerer med *Site Finder* skærmen, hoved skærmen og info skærmen. Dette fandt vi ud af, efter en test hvor Mette Jehrbo udtalte følgende:

Jeg er nok sådan en der har rigtig meget brug for at blive guidet lidt fordi, altså. Jeg finder jo de der genstande, men hvad det er, og hvad man skal med det. Det har jeg nok lidt brug for noget hjælp til tror jeg. Men noget auditivt, det ville være super tror jeg.

Se bilag C.4, s. 121

Som dette citat beskriver, så siger Mette Jehrbo, at hendes oplevelse med installationen kunne have været væsentligt forbedret, hvis en introduktion havde været givet inden interaktionen. I kraft af at installationen ikke er helt færdig, da der mangler vejledning, kan noget af forklaringen være at, der mangler information om nogle af de skridt brugeren skal gennemgå, for at opnå en succesfuld oplevelse. På trods af en mangel på information, var det derfor også vigtigt for os at undersøge om interaktionen var fordybende nok, da installationen, som den er nu, står for sig selv, uden en egentlig introduktion til det samlede rum. For at vi kan måle oplevelsessevnen ved vores installation, som taget ud af kontekst, er det derfor meget vigtigt at det visuelle output, når man bevæger suget hen over skærmen, er forstærkende i forhold til dens mål om fordybelse. På baggrund af en udtalelse fra Niclas Wathne, opdagede vi dog at der opstod et problem. I vores designprocess vægtede vi også *trackingen*, og dermed den visuelle effekt ved suget, meget højt, og selvom vi kom så langt, så vi mere eller mindre havde en flydende overgang, viste det sig ikke at være nok. Niclas udtalte som følger, på baggrund af hans egen og skoleklassernes oplevelse med interaktionen ved vores installation:

"Nu har jeg jo prøvet det før, og der forstyrrede det, fordi man blev i tvivl om den virkede, eller om man gjorde det rigtigt. Det var også det som ungerne fandt ud af meget hurtigt, specielt dem som var meget interesserede i teknik, computere og telefoner og sådan noget. De (Her mimer han at han slår slangen ned i bordet), og så virkede den ikke. Så gik de hurtigt rundt, og så skete der ikke noget kunne de se, så smed de suget, og sagde "Så virker det ikke."

Se bilag C.3, s. 99

Det er dermed vigtigt, hvis man skal skabe en så optimal mulighed for fordybelse som muligt, at det visuelle output er perfekt i forhold til det faktiske input. *Tracking* og *mapping* skal altså være meget præcise og lagrer den visuelle del bare en

7.2. Diskussion

smule, bliver man hurtigt forstyrret fra det virtuelle univers, man befinder sig i. Her forstået som en marinarkæologisk udgravning. På denne måde mister man noget af muligheden ved at formidle om faget, hvis det tekniske halter. På samme måde udtalte Andreas Kallmeyer Bloch, sig således om installationens evne til at skabe fordybelse hos ham:

"(...) Men jeg synes den er fordybende og jeg synes at det giver en fornemmelse af det vi gør. Med det der som vi snakkede om før, måske den der mere sugende fornemmelse ville være god at finde frem til. Jeg kunne godt tænke mig at der var mere sådan et konstant tempo på det man graver efter. Fordi det kan virke forvirrende at den hopper en lille smule, og at man har en fornemmelse af at man graver i et konstant tempo."

Se bilag C.5, s. 124

Afslutningsvis, skal det også bemærkes at ved en optimering af de funktioner vi anvender, og en eventuelt stærkere CPU, ville det visuelle output have muligheden for at være mere flydende, og derved ville vi have en større mulighed for at øge fordybelsen hos brugeren.

Kapitel 8

Konklusion

I konklusionen vil vi kort opsummere hvad vi har afdækket i vores analyse og diskussionsafsnit om det stillede problem.

I forbindelse med analysen og diskussionen beskrev vi hvad vi kunne udlede, på baggrund af vores empiri, om vores stillede problem. Vores problemformulering lød således:

- Hvordan designer vi en interaktiv installation som agter at formidle troværdigt om en arkæologisk udgravning under vandet og som gennem tekniske virkemidler formår at skabe fordybelse for brugeren?

I forbindelse med analysen og diskussionen fik vi afklaret at, det er vigtigt at anse troværdighed og fordybelse som to forskellige aspekter, for at man kan designe en interaktiv installation, hvis mål er at formidle marinarkæologi, og som samtidigt formår at skabe fordybelse hos brugeren. Disse to aspekter hjælper til formidlingen af marinarkæologi på hver deres måde.

Troværdigheden skal ses som det aspekt, der formår at fortælle en del af den historie der er en marinarkæologs arbejde, ved en marinarkæologisk udgravning. Denne troværdighed fordrede vi ved at indføre 3D scannede objekter, af virkelige genstande, samt andre visuelle animationer. På baggrund af vores tests kan vi konkludere at vi til dels formåede at skabe et troværdigt undervandsmiljø, der formidler om marinarkæologi. Samtidigt fik vi også afklaret at når man forsøger at formidle om marinarkæologi, ved at skabe et troværdigt miljø, er det ikke nødvendigt at skabe et virtuelt miljø, der stemmer fuldstændigt overens med den virkelige handling. Ved derimod skabe at et virtuelt univers, som er fordybende og indle-

velsesrigt, formår installationen at formidle om marinarkæologi på en troværdig måde.

Hvad angår det fordybende element i vores problemstilling, kan vi på baggrund af vores indsamlede empiri, samt tests, interviews og undersøgelser, konkludere at det visuelle output, i form af installationens evne til at kunne 'suge' flydende, er yderst essentielt for brugerens mulighed, for at fordybe sig i det præsenterede virtuelle miljø. Hvis man altså vil skabe fordybelse, er det ikke blot vigtigt at skabe en underholdende installation, som formår at præsentere muligheden for at gå på opdagelse. Det er også vigtigt at alle de tekniske dele fungerer optimalt, i forhold til målet om at skabe formidling om marinarkæologi gennem fordybelse.

Kapitel 9

Perspektivering

9.1 Alternative modifikationer

Da dette andet projekt i samarbejde med Vikingeskibsmuseet gik i gang, overvejede vi flere nye modifikationer. Vores opgave var at vælge hvilke modifikationer der ville være ønsket, først og fremmest.

Der er blevet implementeret en *Site Finder* og der er blevet monteret en tredje skærm, hvis formål er at visualisere de objekter, der bliver gravet frem. Dette er bare nogle af de mere synlige modifikationer der er blevet foretaget. Andre modifikationer vi overvejede, indebar blandt andet en loftsprøjektion og at indføre et auditivt element.

9.1.1 Loftsprøjektion

Loftsprøjektionen vi overvejede, ville være blevet installeret, for at forøge fordybelsen i installationen, og på denne måde skabe en fornemmelse af, at man befandt sig under vandet. Når man er i gang med at suge, giver den mulighed for at kigge op og se en video af vandoverfladen nedefra. Inden valget af et *site* var tanken, at vise en prøjektion af en blå himmel, i stedet.

9.1.2 Lyd

Vi overvejede derudover at indføre lyd til installationen, for igen at skabe en større fordybelse, ved at give brugeren en større fornemmelse af at være under vandet.

9.2. Generelle forbedringer

Dette ville dog igen være en effekt, der muligvis ville fjerne noget af realismen fra installationen, for dem med erfaring om emnet. Dette ville derimod give en større oplevelses faktor ved at give følelsen af at man opholder sig under vandet.

9.2 Generelle forbedringer

9.2.1 Site Finder

Site Finderen som langt hen af vejen er bygget færdig, har dog stadig mulighed for at kunne forbedres yderligere. Der kan være muligheden for at lave en mere flydende overgang fra valget af *site*, til hovedskærmen, med hensyn til eksempelvis at benytte kortet som er taget i brug i *Site Finderen*. For derefter at trække det hen på hovedskærmen, og zoome ind på *sitet* der fra.

9.2.2 IR tracking

Et problem vi stødte på under videreudviklingen af installationen var, at der godt kunne være mere præcis *tracking*, ved hjælp af kameraerne under bordet, og det er et problem der ikke blev løst helt optimalt. Den ene løsning indebar at få fat i nogle andre linser til kameraerne, eller nye kameraer, siden der var brug for mere vidvinkel på kameraerne på 73°, som beskrevet i det tekniske afsnit. Den anden løsning indebar at hæve bordet præcis 5 centimeter, for manuelt at skabe denne vidvinkel. Som beskrevet i det tekniske afsnit hvor beregningerne er foretaget, ville dette være en mulighed. Dog vil dette medføre mindre tilgængelighed for børn, da den ville blive højere. Som beskrevet af antropolog Niclas Wathne i det tekniske afsnit, skal bordet helst ikke hæves.

9.3 Fremtidige planer for installationen

På nuværende tidspunkt findes der en deadline sat fra Vikingeskibsmuseets side, som lyder på at de gerne vil have installationen færdig den 10. februar 2017, i hvert fald til et vis niveau. Dette betyder at de igangværende processer skal være færdige og klar til den dato. Dette indebærer ikke andre fremtidige planer der blevet lagt for installationen, hvilket betyder at der vil være rig mulighed for at kunne bygge videre på installationen efter dette projekt.

Kapitel 10

Acknowledgement

Gruppens medlemmer vil gerne takke alle medarbejdere på Vikingskibsmuseet, alle dem som har prøvet installationen og andre implicerede.

Derudover vil vi også gerne udbringe en speciel tak til:

Andreas Kallmeyer Bloch.

Niclas Wathne.

Steffen Engler Thorlund.

Litteratur

Rune Bastian Barrett, Nikolai Peter Hjorth, Sebastian Valentin Jakobsen, Johannes Riis, Jakob Thormod, and Patrick Østergaard. Marinarkæologisk interaktiv installation / marine archaeological interactive installation, 2016. URL http://rudar.ruc.dk/handle/1800/dspace_oai:oai:pure.atira.dk:studentproject/34b833d7-9ea9-47bc-b2ab-237954d320da.

Paul Dourish. *Where the Action Is, Chapter 4: "Being-in-the-World": Embodied Interaction*. First MIT Press, pb edition, 2004.

Dag Svanæs. Interaction design for and with the lived body: Some implications of merleau-ponty's phenomenology. *A C M Transactions on Computer - Human Interaction*, 20(1), 3 2013. ISSN 1073-0516. doi: <http://dx.doi.org/10.1145/2442106.2442114>. The paper was also presented as at CHI 2013 in Paris, France. Slides: <http://www.slideshare.net/DagSvans/chi2013-presentation>.

Robert Rosenberger and Peter-Paul Verbeek. A field guide to postphenomenology. In Robert Rosenberger and Peter-Paul Verbeek, editors, *Postphenomenological Investigations: Essays on Human-Technology Relations*, Postphenomenology and the Philosophy of Technology, pages 9 – 41. Lexington Books, 2015.

Don Ihde. *Technology and the lifeworld: From garden to earth*. Number 560. Indiana University Press, 1990.

Sian Bayne, Jen Ross, and Zoe Williamson. Objects, subjects, bits and bytes: learning from the digital collections of the national museums. *Museum and Society*, 7(2):110–124, 7 2009. ISSN 1479-8360.

Steinar Kvale and Svend Brinkmann. *Interview: Introduktion til et håndværk*. Hans Reitzel, Danmark, 2 edition, 2009. ISBN 9788741251981.

Litteratur

- Cornelius Holtorf. *Archaeology is a Brand: The Meaning of Archaeology in Contemporary Popular Culture*. Archaeopress, Gordon House, 276 Banbury Road, Oxford OX2 7ED England, 7. ed. edition, 2007.
- Anders Högberg. Bilder av skånsk uppdragsarkeologi hos allmänheten och i dagspress. *Meta: Medeltidsarkeologisk tidskrift*, 4, 2004.
- Lisa Mackinney. Something old in the earth: Front-end interviews about archaeology with visitors to the California Academy of Sciences. *Ms. on file, California Academy of Sciences, San Francisco, California*, 1994.
- Maria Ramos and David Duganne. *Exploring public perceptions and attitudes about archaeology*. Harris Interactive Inc, 2000.
- John R Cole. Cult archaeology and unscientific method and theory. *Advances in Archaeological Method and Theory*, 3:1–33, 1980.
- Kevin Hetherington. Museums and the visually impaired: the spatial politics of access. *The Sociological Review*, 48(3):444–463, 2000. ISSN 0038-0261. doi: 10.1111/1467-954X.00225.
- Roy Hawkey. Learning with digital technologies in museums, science centres and galleries. 2004.
- Julian Thomas. *Archaeology and modernity*. Psychology Press, 2004.
- Björn Magnusson Staaf. The rise and decline (?) of the modern in Sweden: reflected through cultural resource management archaeology. *Current Swedish Archaeology*, 8:179–194, 2000.
- Andrew Feenberg. The written world: On the theory and practice of computer conferencing. *Mindweave: Communication, computers, and distance education*, pages 22–39, 1989.
- Mark Poster. *What's the Matter with the Internet?*, volume 3. U of Minnesota Press, 2001.
- Ross Parry. *Recoding the museum: Digital heritage and the technologies of change*. Routledge, 2007.
- D. Anderson. *A Common Wealth: Museums in the Learning Age*. Stationery Office, 1999. ISBN 9780117023376.

Litteratur

Simon J Knell. The shape of things to come: Museums in the technological landscape. 2003.

Oblinger D Dziuban C Lorenzo, G. How choice, co-creation, and culture are changing what it means to be net savvy. 2006.

Nick Merriman. *Public archaeology*. Routledge, 2004.

Maurice Merleau-Ponty. Phenomenology of perception routledge & kegan paul. London, UK, 182, 1962.

Craig Larman. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Prentice Hall PTR Upper Saddle River, NJ, USA, 3. ed. edition, 2004.

Jesper Simonsen, Connie Svabo, Sara Malou Strandvad, Kristine Samson, Morten Hertzum, and Ole Erik Hansen. *Situated Methods in Design*. MIT Press, 7 2014. ISBN 978-0-262-02763-2.

Jasper Flick. Trianguliserig. <http://catlikecoding.com/unity/tutorials/marching-squares/>.

Matt Williams. Cubiquity. <http://www.volumesoffun.com/tag/cubiquity/>.

Wikipedia. Entity component systems. <https://en.wikipedia.org/wiki/Entity%E2%80%93component%E2%80%93system>, a.

Alan Stagner. Good coding practices in unity. <https://unity3d.com/learn/tutorials/modules/intermediate/scripting/coding-practices>, a.

Alan Stagner. Debate regarding the entity component system. <http://softwareengineering.stackexchange.com/questions/211708/is-the-entity-component-system-architecture-object-oriented-by-definition/244571>, b.

Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '87*, pages 25–34, New York, NY, USA, 1987. ACM. ISBN 0-89791-227-6. doi: 10.1145/37401.37406.

Wikipedia. Cubiquity. https://en.wikipedia.org/wiki/Finite-state_machine, b.

Litteratur

G. Bradski. *Dr. Dobb's Journal of Software Tools*.

Wikipedia, the free encyclopedia. Field of view. https://upload.wikimedia.org/wikipedia/commons/7/72/Angle_of_view.svg.

Humtek. Studieordning - den humanistisk-teknologiske bacheloruddannelse af 20. februar 2013. https://ruc.dk/sites/default/files/2016-07/Studieordning_for_den_Humanistisk_Teknologiske_Bacheloruddannelse_1_februar_2016.pdf, 2016.

Informatik. Studieordning - fagmodul i informatik af 1. september 2013 med ændringer af 1. februar 2014 og 1. september 2015. https://ruc.dk/sites/default/files/2016-07/Informatik_fagmodulbeskrivelse_pr_1_september_2015.pdf, 2015.

Datalogi. Studieordning - fagmodul i datalogi af 1. september 2013 med ændringer af 1. februar 2014. https://ruc.dk/sites/default/files/2016-07/Computer_Science_fagmodulbeskrivelse_pr_1_februar_2014.pdf, 2014.

Kapitel 11

Bilag

Bilag A

Tværfagligt samarbejde

A.1 Gennemgang af Tværfagligt Samarbejde

Her forklares hvorledes sammenhængen mellem det tværfaglige fagmoduler og opfyldning af disse muliggøres gennem studieordningen og hvorledes disse fag og fagmoduler forholder sig til gruppens medlemmer.

Det tværfaglige består i at kombinere fagmodulprojekter i hhv. STS, Informatik og Datalogi. Hver især skal gruppens medlemmer stå forfølgende hovedområde:

- › Rune B. Barrett - Datalogi / Informatik
- › Sebastian A. V. Jakobsen - STS / TSA / D&K
- › Nikolai P. Hjorth - Datalogi / Informatik
- › Andreas O. Thomsen - STS / TSA / D&K
- › Jakob Thormod - Datalogi / Informatik
- › Patrick Østergaard - STS / TSA / D&K

I vores 2. Semester projekt havde vi alle tre af HumTeks dimensioner indeholdt, da 2 af gruppens medlemmer (Jakob Thormod og Rune Barrett) fik mulighed for at skrive deres 3. Semesterprojekt i samarbejde med det samlede 2. Semesterprojekt. Vi har derfor ligesom, i forbindelse med vores 2. Semeseterprojekt (Barrett et al., 2016), taget forbehold for STS dimensionen og samtidigt overholdt de relevante dimensioner (herunder Design & Konstruktion og Teknologiske Systemer & Artefakter). (Humtek, 2016)

A.2. STS

Vi argumenterer derfor for, gennem en gennemgang af kravsspecifikationerne for de tre forskellige dimensioner, STS, Informatik og Datalogi, hvorledes vi tænker at de sammenhørigt indgår i et fagmoduls projekt med gruppens medlemmer.

A.2 STS

I vores 2. Semester projekt (Barrett et al., 2016) fokuserede vi i vores STS del hovedsageligt på videnskabsteori og anvendte således også observationsteknikker for at anskue hvorledes brugerne interagerede med vores installation. Til netop denne disciplin, som vi videreudvikler på i dette projekt, ser vi også en kobling til informatik. For som der står i §2 af studieordningen for informatik ønses der en udformning af viden, færdigheder og kompetencer inden for:

*"Viden om systemudvikling med hovedvægt på forundersøgelse og evaluering.
Viden om samspillet mellem it-systemer og forandringsprocesser i samfundet."*

Den videnskabsteoretiske del, samt koblingen til informatik i forbindelse med forundersøgelse og evaluering, tager således udgangspunkt i hvordan vi som mennesker erkender og anvender viden som aktivitet, samt en observation og evaluering af brugernes tilgang til vores installation.

Da vi har bygget en installation der er anvendelig og test klar ser vi dette som et opnået og slutteligt skal det også nævnes at Vikingskibsmuseet gav mulighed for tests og eksperimenter i november og december, som del af en forundersøgelse til den færdige udstilling.

A.3 Informatik

I studieordningen for informatik (Informatik, 2015) §2 præsenteres der visse punkter der ønskes at de studerende får færdigheder, kompetencer og viden indenfor. Vi vil derfor herunder, med udgangspunkter i disse tre forskellige punkter som nævnt i §2 i studieordningen for informatik, fremhæve hvorledes vi udfylder disse kravspecifikationer:

A.3.1 Viden

- › *Viden om modellering, herunder videns- og informationsmodellering. Viden om projektledelse og it i en organisatorisk kontekst.*
- › *Viden om realisering af it ud fra eksisterende rammeteknologier.*

Vi mener at vi foruden at have opfyldt disse krav i vores fagmodulprojekt og at vi også kan argumentere for muliggørelse af en tværfaglighed mellem STS og Data-logi.

Vi mener yderligere at vi bidrager til tværfagligheden i dette projekt gennem punktet omhandlende IT i en organisatorisk kontekst og realisering af IT ud fra eksisterende rammeteknologier. Dette muliggøres gennem STS del som blandt andet omhandler hvorledes teknologi og formidling af arkæologi samt fund, skal tænkes ind i den sociokulturelle kontekst, den selv og dens besøgende befinder sig i. Derudover realiserer vi målet om tværfaglighed gennem punktet om realisering af IT ud fra eksisterende rammeteknologier, ved både at analysere og viderebygge på vores anvendte design model (Simonsen et al., 2014, p. 276) og ydermere analyserer vi på vores STS baserede observationer, hvorledes vi realiserer vores ønsker om at skabe fordybelse for brugeren, der interagerer med installationen.

A.3.2 Færdigheder

Under punkterne for færdigheder, der ønskes, indgår der fire punkter. Disse fire punkter lyder som følger:

- › *Færdighed i at analysere, designe og vurdere it-anvendelser.*
- › *Færdighed i at realisere et mindre program ved hjælp af rammeteknologier.*
- › *Færdighed i at modellere domæneviden, og vælge og argumentere for valg af metoder hertil.*
- › *Færdighed i at vurdere styrker og svagheder ved de mest udbredte menneske-maskine grænseflader.*

Af disse fire punkter indenfor færdigheder vil vi argumentere for at vi har berørt alle fire. Hvad angår punktet omhandlende at opnå færdigheder indenfor analyse, design og vurdering af IT-anvendelser er disse forbundet tværfagligt, i den forstand at vi gennem vores brugerundersøgelser og tests opnår indsigt i hvilke dele

A.3. Informatik

der hjælper eller ikke hjælper til at skabe fordybelse for brugeren i anvendelse af vores IT-system. Vi kan på denne måde nemmere vurdere hvilke tilføjelser eller eksisterende dele som fungerer eller hvilke dele der skal føjes. På denne måde opnår vi færdigheder i at designe ved hjælp af analyse og vurdering af IT-anvendelser. Des videre opnår vi færdigheder i at realisere et mindre program via rammeteknologier ved at tilføje nye dele som hjælper til en optimering af vores installation. Et eksempel på dette er implementeringen af de fisk der svømmer rundt i over den virtuelle sandbund. Disse fisks opførsel er skrevet ud fra en *Swarm Intelligence* algoritme originalt udviklet af Craig Reynolds (Reynolds, 1987) altså en rammeteknologi og vil blive beskrevet i afsnittet "Nye Features". Vi ønsker ydermere at opnå færdigheder indenfor modellering af domæneviden gennem vores tværfaglighed.

Da formidling af arkæologi er det domæne vi arbejder i og hvis kontekst skal itænkes i vores it system vil vi argumentere at vi kan opnå færdigheder til at vælge og argumentere for valg af metoder ud fra denne tankegang. Vores STS begrundede brugerundersøgelser vil på denne måde danne empiri der fordrer til til-og-fravalg af metoder der kan modellere vores domæneviden. Afslutningsvis vil vi argumentere for at vi opnår færdigheder i at vurdere styrker og svagheder ved de mest udbredte menneske-maskine grænseflader gennem vores analyse af vores problemformulering. Altså hvorledes mødet mellem mennesker og maskiner eller mennesker og teknologi udformer sig i vores installation.

A.3.3 Kompetencer

Det samme gælder således også for de to nedenstående punkter inden for den ønskede opnåelse af kompetencer inden for faget informatik.

- › *Kompetence til at planlægge, styre og gennemføre et mindre systemudviklingsforløb, med særlig vægt på forløbets tidlige aktiviteter.*
- › *Kompetence til at dokumentere ovenstående.*

Hvad angår opnåelsen af de ovenstående kompetencer vil vi dette projekt argumentere for at vi opnår disse kompetencer i vores tværfaglige projekt gennem de tilføjelser vi tilføjer til installationen i systemudviklingens forløb. Projektets konklusion berører på denne måde, i hvor høj grad vi er kommet i mål med den ønskede kompetence om at planlægge, styre og gennemføre dette systemudviklingsforløb. Yderligere er den ønskede kompetence om at kunne dokumentere evnen til

at kunne planlægge, styre og gennemføre et mindre systemudviklingsforløb være opgaven i sig selv.

A.4 Datalogi

I studieordningen for datalogi (Datalogi, 2014) indgår der, ligesom i studieordningen for informatik de tre felter, omhandlende viden, færdigheder og kompetencer der ønskes at de studerende får. Vi vil herunder argumentere for hvorledes vi tænker at overholde og sammenbinde disse punkter med de to andre studieordninger.

A.4.1 Viden

I punktet under viden indgår følgende to punkter:

- › *Viden om softwareudvikling, herunder programmering, algoritmer og datastrukturer, datalogiske modeller, beregnelighed og kompleksitet.*
- › *Viden om forskellige programmeringsparadigmer, herunder det objektorienterede. Viden om designprocesser og IT systemers arkitektur.*

Hvad angår Datalogis studieordning om den ønskede opnåelse af viden vil vi her dels argumentere for muliggørelsen af dette, men her især muliggørelsen af dette gennem et tværfagligt perspektiv. I dette projekt tilgår vores viden om softwareudvikling, programmering, algoritmer og datastrukturer gennem vores tekniske afsnit, som omhandler opfyldelsen af datalogi og informatik baserede del. Denne del vil vi i det tekniske afsnit beskrive hvorledes vores software fungerer, dets struktur, anvendte programmering og tilføjelsen af en algoritme. Dette lader sig dels gøre gennem vores opnåede viden til at udføre tilføjelser til installationen som muliggør dette, men vil også forholde sig til den del af tværfagligheden som omhandler mennesket kontra teknologi og virkeligheden kontra virtualitet. Som også beskrevet i afsnittet omhandlende nye features inkluderer vi en *Swarm Intelligence* algoritme til de fisk, der svømmer rundt i vores installationen. Programmeringen og udformningen af denne *Swarm Intelligence* algoritme tjener til opnåelsen af viden inden for datalogi, men føjer også til den samlede oplevelse som brugeren får, ved anvendelsen af vores installation. Yderligere vil denne rapport også illustrere hvorledes vi har opnået viden om forskellige design processer og it-systemers arkitektur gennem dets tekniske afsnit. I denne rapport illustrerer vi derfor hvorledes vi har gennemgået og analyseret vores design process for at

A.4. Datalogi

opnå det ønskede mål og vi beskriver desuden også vores it-systems arkitektur i vores tekniske afsnit.

A.4.2 Færdigheder

I punktet under færdigheder for studieordningen inden for datalogi, §2. Står følgende tre punkter nævnt.

- › *Færdighed i at programmere, afprøve og dokumentere et mindre program i et højere, generelt programmeringssprog.*
- › *Færdighed i at vælge og argumentere for valg af design, datastrukturer og algoritmer til konkrete opgaver.*
- › *Færdighed i at estimere effektivitet og skalerbarhed af generelle algoritmetyper. Færdighed i at specificere og modellere krav til it-systemers funktionalitet.*

De tre ovenstående ønskede færdigheder vil vi således også henvise til at vi har formået at opnå. Dels ved at opnå færdigheder inden programmering, afprøvning og dokumentation. Dette gjort i et højere generelt programmeringssprog ved udførelsen af de tilføjelser vi har tilføjet til installationen. Det eksisterende software er skrevet i C# og vi har gennem vores afprøvningen i forbindelse med brugerundersøgelserne, dels opnået færdigheder inden for dette, ved at udforme en testbar installation. Samtidig har vi også udformet dokumentation, som repræsenteres i det tekniske afsnit og den vedlagte kode. Des videre vil vi argumentere for at vi opnået færdigheder i at argumentere for valg af design gennem de tilføjelser vi tilføjer på baggrund af de første brugerorienterede observationer. Disse observationer analyseres i forbindelse med vores metodiske afsnit, som berører STS delen og samtidigt føjer til yderligere perspektiveringer inden for design. Afslutningsvis har vi opnået færdigheder i at kunne specificere og modellere krav til IT-systemers funktionalitet på samme måde som ovenstående gennemgang af opnåelse af færdigheder.

A.4.3 Kompetencer

I punktet under kompetencer for studieordningen inden for datalogi, §2. Står følgende to punkter nævnt.

- › *Kompetence til at planlægge, kravspecificere, styre og gennemføre et mindre softwareudviklingsforløb.*

› *Kompetence til at dokumentere ovenstående.*

I forbindelse med de ønskede kompetencer mener vi, som vi også beskriver i forbindelse med de ønskede kompetencer i forbindelse med informatik, at rapporten illustrerer hvorledes vi løser dette. Dels illustrerer den, hvorledes vi har opnået kompetencen til at planlægge, kravsspecificere, styre og gennemføre et mindre softwareudviklingsforløb, gennem analysen og konklusionen på projektet. Hvad angår det sidste punkt advokerer vi for at den ønskede opnåede kompetence for dokumentation af ovenstående punkt er illustreret gennem rapportens evne til dette.

Bilag B

Interview Spørgsmål

B.1 Spørgsmål til semistrukturerede interviews

B.1.1 Formål

Formålet er at få "svar" på problemformuleringen:

"Hvordan skaber vi en troværdig/virkelighedsnær interaktiv installation, som gennem dens tekniske virkemidler formår at skabe fordybelse for brugeren og samtidig formidler om marinarkæologi?"

B.1.2 Form

Semistruktureret interview - Giver mulighed for at stille uddybende spørgsmål, og vi behøver ikke holde os til rækkefølgen af spørgsmål. Interviewet må gerne forløbe mere som en samtale.

B.1.3 Generelt:

Anspor til samtale eller opfordre til at de kommer med kommentarer mens de bruger den. De må meget gerne sige højt hvad de tænker. Gør det klart at det er okay at komme med kritik. (Vi bliver ikke kede af det hvis de kommer med kritik. Tvært imod ;=))

B.1.4 Tekniske spørgsmål:

Spørgsmål som omhandler selve installationens udseende og konstruktion.

Bilag B. Interview Spørgsmål

- › Vi vil gerne have svar på hvor "behagelig" installationen er at bruge. Bliver folk for eksempel frustreret over trackingen, er det intuitivt at bruge den etc.
- › Hvor troværdig føles undervandsmiljøet? Kan du sætte nogen ord på? Bliver man draget?
- › Hvor fordybende virker installationen i forhold til hvordan man bevæger suget?
- › Tror du at folk ville kunne få fornemmelse af at være arkæolog og gå på opdagelse?
- › Tror du at, hvis måden man bevægede suget på virkede bedre, om det så ville højne oplevelsen? Eller oplevelsesevnen?
- › Tror du for meget eller for lidt info omkring objekterne, eller brugervejledning vil forstyrre oplevelsen?

B.1.5 Oplevelsesspørgsmål:

Spørgsmål som omhandler selve oplevelsens.

Vi vil gerne have svar på om vores installation fanger/tiltaler brugerne. Lærer de noget om marinearkæologi?

Bed brugerne om at beskrive hvad det er de laver når de står ved installationen.

- › Hvilken følelse fik du da du prøvede installationen? (Bliver det for føle føle?)
- › Synes du at du har lært noget om (marin)arkæologi? (Hvis ja, hvad? uddyb.)

B.1.6 Arkæologi spørgsmål:

Spørgsmål henvendt til arkæologer, eller andre med viden inden for, for eksempel formidling.

- › Hvad synes du om måden at formidle (marin)arkæologi på?
- › Hvad synes du er det vigtigste at få formidlet omkring (marin)arkæologi?
- › Synes du der mangler noget. Hvis ja, hvad så?

B.1. Spørgsmål til semistrukturerede interviews

B.1.7 Antropolog-spørgsmål:

Spørgsmål henvendt til antropologen(Niclas?) fra VSM, omhandlende hans interview med skoleklasser.

- › Hvis han har analyseret på den data han har indsamlet, hvad er så hans umiddelbare konklusion? Hvad er han kommet frem til?

B.1.8 STSamfund-Andreas-spørgsmål

Spørgsmål til Andreas angående samfund vs. museum. Kan Andreas/museet "mærke".

- › Hvad er arkæologi for dig?
- › Opfatter du "offentligheden" som havende en indflydelse på arkæologien?
- › Har du/i kunne mærke ændringer, i for eksempel antal besøgende. Ud fra bøger, film, serier og så videre?
- › Hvor meget indflydelse har regler og love for arkæologien?
- › Hvordan kan vores installation bidrage til arkæologien, og er der overhovedet behov for det?

Bilag C

Interview

C.1 Interview med Marinarkæologer - Foretaget af Sebastian A. V. Jakobsen d. 14-11-2016

Interviewer: Sebastian Jakobsen (SJ)

Interviewede: Warren Wrigler (WW)

SJ:

Would you mind telling me your name, and what you study?

WW:

Ah, my name is Warren Wrigler, and i'm studying maritime archaeology master study at STU.

SJ:

Okay, thank you hmm. The installation you just tried, would you think. What would you say to its immersiveness like does it simulate the way you work out in the field? At least to some degree.

WW:

Yeah, i think definetly to some degree it does simulate it, and i think that's the key. To try to simulate underwater work is complex. So i think with museums you gotta do what you can, and i like. I liked the lot of it hmm. But i think the most, at least for my perspective the most important thing, is immersing people underwater as best they can. And that's why the room where you lay down where you like feel you are underwater. I like that the most. Because like i think, whenever I talk to people about doing maritime archaeology. The most interesting thing there is i how do you do work underwater.

SJ:

Yeah it's a different environment.

WW:

Yeah it's a different environment, like how do you do this. Well it's not easy. Explaining and getting people to understand that, is different than to make them feel like they are under water, trying to like do something which is difficult but yeah that was my favourite room definitely.

SJ:

Okay jep, what should ehm. If there should be the thing that could make it more realistic? If it's even possibly now that we are above water level so it's pretty hard to simulate that you cant breath and such.

WW:

Yeah arhm. The coolest, and i was mentioning it to Andreas, arh the coolest thing that i personally find as a maritime archaeologist, and the coolest thing i think anybody will love is literally somehow creating like, they created those 3D videos. So like a 3D video where you put on the glasses, and it's a go pro on an archaeologist, on a dive, just jumping into the waters. So now you are the archaeologist, and two minutes descending, and as you're descending you see nothing, nothing, nothing and then the wreck is beneath you. So that, that experience, that first experience of going under water and seeing a wreck under water i think that is the coolest thing, and i think that is the thing that people will find the coolest. Cause one of my favourite, debatable my favourite dive, on my first under water excavation or survey was the first one i did where i just descended on, and i just saw the wreck and was able to swim over it, and that was the coolest thing ever too me.

Bilag C. Interview

SJ:

That sounds pretty cool.

WW:

I think, somehow giving people that experience i really, i think the idea that youre going under water, and you are going deeper and deeper and deeper. And suddenly there's a wreck showing up, and it could be a two minutes long video. It doesn't require a lot of time. But that i think.

SJ:

would be good?

WW:

That i think, people would really like. But that's of course me saying that he he.

SJ:

Yeah, yeah but it's ehh but it's the. It's the subjective view we are after. Okay let me try to see this. The installation hmm. Where you digged in the sand, do you think it comes close to telling the way, like your point of view. The way you work?

WW:

hmm yeah. I think, definitely yeah. When it comes to like when you're excavating, it's a pretty accurate, it's as close as you can get, to simulate it on land i would say. Ahm, and i understand why you guys choose excavating, because children is definitely going to relate more to excavation. The one difficulty, the one thing, and it's just me being. The one difficulty i have with excavation, showing excavation is, maritime archaeologist proberly dont do excavation as often as people think we do, and presenting it, sometimes gives them false idea of how often we do it.

SJ:

So like a different reality?

WW:

Yeah.

SJ:

An opposition to reality.

WW:

Yeah yeah, a slight yeah, maybe. I don't know. But i get why it works with kids. Like you cant tell kids, ohh we don't actually dig stuff up that often, you cant do that cause kids aren't going to be excited anymore. So i think it's a complex question, and a complex problem, and i really don't know how to solve it. But i do think it should be kept in mind, that maritime archaeology don't excavate as much as people think we do.

SJ:

Hmm. Sorry about that, yes. I was just writing down what you said. Did you see, i don't know if you even noticed it, it wasn't so clear, but there was actually some real objects behind.

WW:

Yes, yeah yeah.

SJ:

Did you see like ehm, what's the word for it. Like a connection between the items that you found, which where archaeological, and the other items that where there.

WW:

Yeah yeah i mean. You've, i mean there there. I mean having the items like digging them up, and having the actual physical item to touch i think, or like a similar one is kinda nice, i think definitely people like that. And it's, it's cool, cause it's like. Ohh i just did this thing virtually, now i have an actual thing to touch with my hands. As opposed to. Ohh i saw a virtually cool.. And then you walk away. Ha ha. It's a little more, i think, i think. Having the object is really important, and i think that's good, because it maintains interest

SJ:

Yeah we. That's also what we are trying to make, a relation between each.

WW:

Each station?

SJ:

Yeah was that clear?

Bilag C. Interview

WW:

Yeah yeah, it makes, it makes a relation between each station i think.

SJ:

That's what it's supposed to end out with i think. Now it can be a little bit unclear.

WW:

Yeah, yeah i get it.

SJ:

but if you hang around for like an hour or two he he. Ehm, yes, yes, yes. If you should. Ohh yeah. We have an idea of making like a sitefinder. To like. 2.0 edition or maybe 7.0 edition we made so many iterations. We thought of making a sitefinder if you saw the extra screen in there, Where you like can point that this is the danish ocean. And then you canpoint to the mediteranean or somewhere, and when you click it, you dive into that place, and then you can find stuff from that place.

WW:

Ohh that would be really cool.

SJ:

Compared to like.

WW:

Yeah, yeah. I think it would be really cool to show people different areas, than just danish waters. If you could? If it was possible yeah, if you could like use mediterranean site or maybe a caribbean site or something. It would be cool to show people different, cause it would be different environments, it would be completely different waters, it would be different understanding that archaeologists go to different. Like there is being underwater, and there is being underwater in different areas. Which is like, everybody assume oh you are under water, it's the same environment. When in actuality it's not he he.

SS:

Yeah exactly.

C.1. Interview med Marinarkæologer - Foretaget af Sebastian A. V. Jakobsen d.
14-11-2016

WW:

and i thinks that's cool and that would be a cool idea to show them different environments under water, that would definitely work. My personal opinion.

SJ:

which is what's interesting again. I think that was it, thank you very much.

WW:

No problem.

C.2 Interview med Marinarkæologer - Foretaget af Nikolai P. Hjorth d. 14-11-2016

Interviewer: Nikolai Peter Hjorth (NPH)

Interviewede: Joseph Smith(JS) Alice Nate(AN)

NPH:

So if you can just start stating your names.

JS:

Joseph Smith.

AN:

Alice Nate.

NPH:

and what do you do?

JS:

I study maritime archaeology at, syddansk universitet.

AN:

The same.

NPH:

alright. So the first question i have about the installation is how immersive it seems? Does it seem true to the actual work in maritime archaeology?

AN:

well.

JS:

it has the potential to get there.

AN:

Yeah, it's not particular true no, but it comes quite close. I was more thinking of usability.

JS:

when i think of that exhibit, the one with the dredger. When you're doing that, there's so much shit coming up. There is so much like, dirt and sand coming up that makes it hard to see, and maybe you can add that in somehow.

NPH:

yeah

JS:

where you pulling stuff out.

AN:

just a bit more, we know of course, that you cant really do it as dirty as it gets.

JS:

just like a little bit.

AN:

just a bit.

NPH:

It seems a lot, so it's too clean.

JS:

it's too clean.

NPH:

So that was basically my next question. How well it simulates that actual work? I am thinking about the tube.

AN:

The tube ehh. Actually we haven't worked with one yet

JS:

we haven't done one yet, but we know how it works.

AN:

We know how it works yeah.

NPH:

So what do you usually work with, tool wise?

Bilag C. Interview

JS:

measuring sticks.

AN:

yeah, measuring sticks, drawing boards.

JS:

but we do use the actual hose.

NPH:

you use that for actual excavations?

JS:

yep, that's the tools we use.

NPH:

alright.

AN:

maybe the hands as well.

JS:

Yeah but, it's not like massive motions. Just a gentle sweep.

AN:

yeah.

NPH:

SO how relevant do you think the installation is to the actual feeling of excavating something?

AN:

I think.

JS:

It's simulating some stuff.

C.2. Interview med Marinarkæologer - Foretaget af Nikolai P. Hjorth d. 14-11-2016

AN:

It is, it is. It does spark the sort of intrigue and excitement. You get like, oh my god, i'm finding something. But also in the water, you have constant noise of breathing. It's totally different than 'just' this. It does, you do get a bit of excitement like oh my god. There is something, i can see something. That, that's fun, and it does make it interesting. That's what you get out of it.

NPH:

How do you see the connection between the actual object, and the virtual object? So there is the actual object you can see, and then there is the virtual object in the installation.

AN:

Can you recognize them you mean?

NPH:

It's just important the definition between them sort of. When you can see the actual object, then you excavate that virtual object?

AN:

No because, you don't really see them.

JS:

Actually, did i miss like what are they for?

NPH:

The actual objects? They are for bringing around to the other installations. So there is a carbon dating machine.

JS:

i actually know what they are for.

AN:

Yeah, so maybe, i don't know. Put some words on the wall saying, take the object and go to station B.

NPH:

So it needs to be more obvious? The connecting between the..

Bilag C. Interview

NPH:

Does it work with showing the actual object, before you found the virtual object? Or would it be better to..

JS:

You cant really hide it can you.

NPH:

we would hide that actual object, and then reveal it once the virtual object was found?

AN:

How would you do that?

NPH:

we have a bark and shelf, and then a light turns on.

JS:

arh okay, that a good idea.

AN:

that could work.

NPH:

could you see that working better?would it be better that having it sitting there?

JS:

Yeah, at least it brings a little bit of mystery. If it just sits outside, and you already see it. If it's like not so obvious, in like a shelf as you said.

AN:

and the moment you find it then the lights comes on, it adds too it a lot.

NPH:

How important do you see the mystery part as being?

AN:

well it a part of archaeology, that you don't know it's there. It is the one thing of finding treasures. Especially if it's for kids, keep it simple, keep it exciting.

NPH:

alright, so the next implementation we are thinking of a sitefinder, so you can choose between different sites, that actually what the second screen is supposed to be. It's supposed to be a touchscreen, and then you can choose. And then I don't know, whatever. And then the object will change depending on the site Can you think of any obvious problems or limitations you can see with this?

JS:

No.

AN:

No.

JS:

No, i don't think so... Cause you could have some danish wreck it could be really modern, with like a cannonball or something.

AN:

Yeah and maybe with different time periods.

NPH

So time periods would be good as well?

AN:

Yeah.

JS

Yeah.

C.3 Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

Interviewer: Nikolaj P. Hjorth **NPH**, Rune B. Barrett **RB**, Sebastian A. V. Jakobsen **SJ**, Andreas K. Bloch **AB** & Steffen E. Thorlund **ST**

Interviewede: Niclas Wathne(**NW**)

RB:

Og så er der.. Der er omkring 10-10-10-15 genstande, hvis man tæller mønter med i hvert fald.

NW:

Ja, og så er deet, men det er kun de der objekter, arkæologiske objekter, som så kommer op på skærmen, eller?

RB:

Jae. Det er det der er meningen. Der er jo også alle de her sten i hvert fald. Det viiiil.

NW:

Jaer... jaer

RB:

det var det der var tanken. Det er faktisk også meningen at den der dåse skulle poppe op

NW:

Jaer

RB:

Det er ... altså alt hvad der ikke bare. alt hvad der ikke bare er almindelige.. øhh. ting, som sten og planter og

NW:

Jaer

C.3. Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

RB:

Og sådan noget der. var tanken i hvert fald. øh. skulle poppe op, og så er det jo så også meningen. Det er vi faktisk også gået i gang med at skrive, nu har vi bare ikke fået det med over på den her, men at der så kommer noget tekst op

NW:

Jaer

RB:

På den skærm også omkring

NW:

Jaer

RB:

Det objekt man har fundet

NW:

Jaer

RB:

Og måske en. Sådan en baggrundsvideo, med med lidt vandmænd og

NW:

Jaer. Med hensyn til øh selve undervisningsforløbet i hvert fald, og det flow der skal være, så var noget af det goe herovre (vores installation). Det var at de faktisk ikke vidste helt hvad det var (de fandt). Altså for eksempel, øh det her skjold derovre, som var et låg

RB:

Jaer

NW:

Fandt de faktisk, de vidste ikke hvad det var da de gik herfra. Og det er ligesom det der også, som pirrede dem til at gå derover (en af de andre installationer), og prøve at finde ud af hvad det egentlig var, ligesom en arkæolog gør.

AB

Vi ved altid (inaudible)

Bilag C. Interview

NW:

I ved altid hvad i (inaudible) hahahahah

RB:

Jaja historier og årstal og sådan noget med

NW:

De arkæologstuderende, der ikke er helt arkæologer endnu, altså finder noget som som de skal finde ud af hvad er, altså så der må helst ikke, tænker jeg i hvert fald, stå for meget

AB:

Nej der må ikke stå noget. Der må ikke stå noget om genstandene. Der må stå hvor man skal tage den hen bagefter

RB:

Ja. Ja det er måske en meget god ide. for at skabe lidt, skabe lidt nysgerrighed

SJ:

ja, også i forhold til at det skulle være en base, det der, ikk? altså

N:

Præcis. og øh og øh, og netop hele den der ide med at øh, finde det ukendte herovre (vores installation), eller forholdsvis ukendte, og så finde mere ud af det i en proces, som ikke er under vandet, men som er i den der arkæologiske, hvad hedder sådan noget

AB:

Kontor

NW:

Ja

RB:

ja, og hvor man kan stå og holde det i, i hånden . . .

C.3. Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

NW:

Ja, og måle det og gætte og alt muligt andet øh. Noget man ikke kan gøre dernede, når man står med en støvsugerslange og alt det der. Såeh, men det er rigtigt fedt at den kommer op når man, når man finder noget. For noget af det som ungerne også nævnte her, det var at de ikke, vidste ikke helt om da de fandt dåsen om det var faktisk noget de skulle bruge til noget. øh, og der er det der (ekstra skærm), jo en ret god iden.. sådan indikator for hvad er det egentlig, er det noget der er vigtigt BANG, så kommer det op på skærmen. Øh, så det er en rigtig god ide. Øh, og så så snakker vi om de der.. Altså højden på bordet os at kunne måske skulle være lidt højere eller sådan noget. Så er det altså en rigtig god højde. Specielt for ungerne. Når de skal stå og støvsuge os. Og det var endda udskoling ungerne, altså 7., 8., 9. klasse. Så hvis man også skal have nogle mindre børn herind, så kan man selvfølgelig stille dem op på et eller andet, men men øh, så er højden altså ret god.

RB:

Så er den god nu tænker du, eller?

NW:

Ja, det synes jeg.

RB:

Især hvis det også skal kunne være for de lidt mindre øh

NW

Lige præcis

RB:

... børn

NW:

Vi har desværre ikke, som også jeg snakkede med dig (Hvem?) om, øh, fået testet på familier og børn under de der 13 år. øh, så vi ved ikke helt, jo vi har gjort det på nogen, men men øh men men øh, og det har også været fint nok, men vi har ikke fået sådan en hel familie ind og se hvordan det egentlig fungerer med når sådan en familie kommer ind og skal skal til, og være arkæologer, men øh det kan være vi får prøvet det her en(?)

NPH:

Hvad med noget af det visuelle?

NW:

Ja altså jeg synes det er . . . jeg synes det er rigtigt godt. Altså det er, det er realistisk på den måde at man kan se hvad det er. at i . . . altså man kan jo ikke se eksakt at det der det er sådan en, en ting man skal banke ind i et skib, så det er en slags søm eller øh et eller andet, men man kan se at det er noget træ, man kan se det er . . . Noget som er interessant. Noget man skal bruge. Det er ikke en sten. Det er ikke noget andet. Man er ikke i tvivl om hvad der er sten og hvad der er et skelet, altså. Og det er øh .. Det det er så godt at man synes det er . . . altså at det er . . . det er flot! det er flot sat op på den måde, det visuelle.

NPH:

Altså nu nævnte du noget om sådan, sugsevnen da du var i gang. Hvor meget synes du det forstyrrer? I forhold til øh, den samlede oplevelse?

NW:

*Jamen øh, nu har jeg jo prøvet det før, og og og der forstyrrede det. Fordi man man blev i tvivl om . . . om den virkede eller om den øh om man gjorde det rigtigt. og det var også det som ungerne kom op af, altså fandt ud af meget hurtigt at øh. specielt faktisk dem som var rigtigt gode, eller meget interesserede i sådan noget med teknik og computere og telefoner og sådan noget. De *dong dong* (mimer at han slår slangen ned i bordet?) og så virkede den ikke, og så gik de hurtigt rundt og så skete der ikke noget kunne de se, og så smed de det (suget), og så sagde de "så virker det ikke". Øh fordi det kunne i hvert fald ikke passe at de ikke kunne finde ud af det. Indtil man fortalte dem at det skulle være vinkelret på, og 10 centimeter og sådan nogen ting. Og så fandt de ud af det stille og roligt. Men men øh, men det at man kan se en effekt hurtigere og meget lettere, det gør rigtigt meget. Altså specielt også hvis det skal, man skal kunne gennemføre sådan en undersøgelse her og finde noget.*

NPH:

Tror du også sådan noget, altså information om hvordan man bruger suget, eksempelvis hvis den er 10 centimeter over, tror du det ville forstyrre oplevelsen også?

C.3. Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

NW:

Øh. Det er meget skægt du siger det fordi øh, der var her i tirsdags var der nogen fra Aalborg Universitet i Sydhavnen, som lavede det her hologram herude, og øh hvor man også kunne bygge et skib. Man skulle hugge noget træ, og så ta' træet, og så bygge skibet. Og der stod meget meget lidt. Der stod "Hug" herovre på venstreside, og "Saml med hånden" på højreside. Ingen læste det. og man prøver sig frem. Jeg læste det heller ikke. Og vi, vi lavede sådan et interview bagefter hvor de gerne ville høre hvordan hvad vi synes om det. Og de sagde der var ingen der læste det. Og det var store bogstaver, og det var ikke engang særligt meget. Man læser det ikke. Der var i hvert fald ikke nogen der læste det. Og de her unger som har været herinde, selvom øh støvsugerlangen har ligget sådan her (rullet sammen på gulvet?), så har de gået direkte hen, selvom de ikke har været her før. altså så så ved de at det er der de skal gøre et eller andet. Og det er endda selvom det bare ligner en støvsuger faktisk, som ligger dernede (på gulvet), som ikke er blevet ryddet op. Og så ved de at de skal prøve at gøre et eller andet. Så det der med forklaringen, altså det var der med mange af de andre installationer, men denne her installation er der ingen der er i tvivl om, i hvert fald af ungerne, hvad de skal. Og det synes jeg jo var øh ret vildt, og ret fedt. Øh, så der har i virkelig ramt rigtigt med øh også med ... ja ... Med funktionaliteten af den.

NPH:

Hvad med sitefindereren specifikt? Fordi den var her jo ikke da i testede.

NW

Nej det er første gang jeg prøver den, og det er øh altså øh det er jo virkelig flot sat op. og øh ... og smadder smart i den forbindelse at man kan vælge øhm. Måske bliver det lidt for detaljeret tænker jeg på. Hvis, hvis det man skal er at vælge. Øhm, jeg havde lidt svært ved at finde ud af hvordan man egentlig valgte. Man skal bare holde den lidt.

RB:

Og det er måske ikke helt kommunikeret specielt godt. For det ser jo faktisk ud som om at den går i gang når man trykker...

Bilag C. Interview

NW:

Ja, og der er grønt lys og sådan noget, så men men men øh, men det at man kan vælge det er jo vildt fedt. Altså at det ikke bare er et eller andet sted, fiktivt under vand. Det her det er faktisk noget som man prøver at genskabe. Og der er endda forskellige dele. Forskellige vrage. Altså det. Det er jo noget som også bare er ... altså er fedt også for detaljen i selve oplevelsen. øh. synes jeg. Og som også gør at man kan bruge det til.. til forskellige klasser og forskellige mennesker.. og alt efter hvad man har lyst til også. Så øh. i hvert fald også hvis man snakker undervisningsforløb. Øhm.

SJ:

Taget i betragtning af at det sådan er en installation sammen med andre installationer, hvordan synes du så at øh niveauet af indlevelse er der, når man kun skal være der et kort stykke tid?

NW:

Jamen det synes jeg er rigtigt rigtigt stort. øhm. Man man gør noget aktivt. Man man er marinarkæolog. man har den der støvsuger og man man.. man står ikke og læser noget. man står ikke og får noget passivt information i hovedet. man man øh. man er aktivt søgende efter noget og får noget op. Man føler det er lidt unikt at man står der og man finder et eller andet øhm, som man kan se hernede på bunden eller på skærmen. Og det er øh, det er ret stort øhm, og høj indlevelseshøjde synes jeg. Og det. I forbindelse med den her (vores installation), både udstillingen, og i flowet af de her installationer, så er det jo, som også vi stod og snakkede om før, perfekt for ligesom at skabe basen for hvad man laver som marinarkæolog og hvad man skal herinde. Man skal finde det der objekt, det ukendte nede i vandet, og arbejde med det. Og der er det jo rigtigt godt sat op her. Synes jeg. Med hensyn til den der øh, hvad var de i kalder den her (sitefinder)?

NPH:

Sitefinder

NW:

Okay, sitefinder. Så så er det øhm.

RB:

Det er måske sådan lidt internt. At vi kalder den det.

C.3. Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

NW:

Ja øhh. Men men det jeg må. Nu nu så jeg faktisk bagefter at man kunne zoome ud og ind og sådan noget. Det ved jeg ikke, det tror jeg måske er... Det er rigtigt fedt at man kan, men det er måske også sådan lidt for detaljeret. måske. I forhold til at, at det som man egentlig skal her det er at vælge. Og alle ved at det her det er Danmark, og der er nogen forskellige punkter, og øhm. Det tror jeg egentlig er nok. Altså jeg kunne mærke i hvert fald, jeg havde ... jeg kunne se at de automatisk, tror jeg bare, begyndte og lave den der (Pinch med fingrene), og så zoomede den, og så. Men jeg kunne ikke, jeg kunne ikke. Der er ikke så meget formål i det andet end at det er fedt og det ser godt ud. Men men det jeg egentlig bare har brug for det er at vælge. Jeg kan allerede, jeg kan overskue stedet, og jeg kan se at det der er danmark og det dernede det er Nordjylland, og sådan. Og det vælger jeg og så er det det. Det er jo nok information der, og det er det den skal. Synes jeg i hvert fald. Så det der med at sugningen (tror han mener zoomningen), det er fedt den kan, men men øh det er ikke noget som jeg vil have et stort behov for.

RB:

Nej, det er ikke noget der sådan bidrager til til?

NW:

Nej det synes jeg ikke, altså øhm. Nej.

SJ:

Man vil måske hellere bare i gang, eller?

Bilag C. Interview

NW:

Jaer, eller det som man, det som den skal når jeg ser den. Det tænker jeg, det er et sted hvor jeg kan vælge nogen ting. Og og de detaljer jeg skal vide, for at jeg synes jeg kan vælge nogen af de ting, dem har jeg her allerede. Der er et kort over Danmark. Der er et fund heroppe i Nordjylland, og et nede i Sydsjælland, og det er det.. Jeg ved ikke. jeg har ikke helt præcist behov for at vide hvor på sydsjælland eller hvor det er, til at zoome ind på det. Der vil jeg bare kunne vælge. Der behøver jeg ikke at kunne, men det kan der jo være at der er nogen der har. Men det tænker jeg i hvert fald, at det er. Det er måske heller ikke et sted hvor man skal stå og bruge for meget tid. Når man er her. Øhm. Man kan stå og kigge, og man kan stå og se "ja, der er Danmark", og så kan man se der hvor de er, og så kan man vælge. Også fordi hovedattraktionen er her (selve installationen). Øh, og det er også det der skete når man så trykker og man kommer i gang, så kommer det her fantastiske med at man kommer ned under vandet, og som virker helt vildt fedt. Altså. Øhm. Måske med lidt delay. Fordi øhm når man har trykket der, så står man lige og kigger og sådan lidt. og så..

NPH:

Ja, "Hvad sker der nu?"

NW:

Så den skal lige øhh, det der sekund eller et eller andet. Så man når at følge med, og man får det her med. Fordi det er jo netop også noget med det der med fornemmelsen af at komme med ned under vandet.

RB:

Der skal måske være en eller anden, måske man skulle vise fra den her skærm når man har valgt noget, at nu er det den store skærm. . .

NW:

Ja men jeg ved ikke hvor... bliver den lige så mørk?

RB:

Nej den bliver ikke lige så mørk.

NW:

Fordi ellers så kan man jo se meget ... tydeligt at der kommer lys, og at der sker noget herovre. Men hvor er det den skal stå henne? Eller være henne?

C.3. Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

RB:

Vi har faktisk ikke fuldstændigt besluttet det. Altså det kunne

13:53 - 14:13 Vi diskuterer lidt placering af Sitefinder, indtil Steffen (Det teknologiske fyrtårn) afbryder.

ST:

Én ide man kunne gøre, man fik folk til at flytte fokus, det var at lade kortet, alt efter hvor skærmen står henne, nu står den der, animere ind her, og så zoome ind helt fra kortet af, hurtigt. Så fik man flyttet folks øjne fordi nu bevæger det sig ind på den store skærm.

RB:

Ja det kunne selvfølgelig også være en ret sjov måde at komme ned på.

ST:

Eller et eller andet andet der minder om det.

AB:

Det der er lidt specielt når den står der (På installationen), det er ligesom arbejdsområdet

14:47 - 15:00 Alle diskuterer i munden på hinanden hvor den fucking sitefinder skal stå.

NW:

Hvor er det genstandende skal være henne? Er det på en hylde heroppe?

15:05 - 16:51 Placering af de fysiske genstande bliver diskuteret i plenum, og igen placering af sitefinder

Bilag C. Interview

SJ:

Vi kunne vel også, på en eller anden måde, med den der sitefinder, prøve at lave det sådan lidt rejsende, måske lidt eventyr agtigt, det der med "Nu kan du helt selv bestemme hvor du vil hen." og du har det der store billede, og så der hvor du trykker der går din rejse hen, og sådan. Jeg ved ikke om det var den følelse man fik?

NW:

Jo, det er jo netop det med at man man, man fik en, synes jeg i hvert fald, fornemmelse af at det, det var ikke bare et eller andet virtuelt sted i havde bygget op uden tanke om hvad man egentlig kunne finde og sådan noget. Det var et eller andet historisk, altså, sket. Noget der var reelt. øhm. et sted som var fundet i virkeligheden. Og det er. Det synes jeg gør det mere interessant, fordi der så øges detaljen og indlevelsen.

RB:

Den følelse den fik du lidt nu, eller hvad?

NW:

Ja det synes jeg. JEg synes det var rigtigt godt ved den. Også netop at det er ikke et moderne google map, eller noget med E4'eren (The fuck?) og alt det der, men det faktisk er sådan et historisk kort, som i har smidt op. Det synes jeg også er ret fedt. Det er også .. den der .. Stemningen bliver skabt omkring at det her det er noget historisk og det er noget .. ja det er i hvert fald ikke noget fra nutiden, vi går og søger efter her.

SJ:

Så det bidrager også til indragelsen kan man sige?

NW:

Ja, helt sikkert.

18:26 - 19:54 RB roser det kort han selv har fundet? SJ cracker en joke. Der bliver leet. Der bliver snakket i plenum om de 2 "steder" vi har på sitefinderen, om det er rigtige steder og så videre. AB peget steder ud på kortet hvor de har udgravet.

AB udpeger nogle steder som ligger forholdsvis tæt på hinanden, og RB gør sin entre:

C.3. Interview med Antropolog Niclas Wathne - Foretaget af flere d. 15-11-2016

RB:

Det er i den situation, det muuuuuligvis kunne være lækkert at have mulighed for at zoome lidt ind.

NW:

Præcis, det tænkte jeg også på. Fordi nu nu var de placeret i hver ende af Danmarkskortet, og det kunne man tydeligt se. Og der er jeg enig med dig i, at der kunne man meget hurtigt blive. altså. Have brug for at zoome ind.

NPH:

Hvis nu knappenålene var meget mindre, så man først så dem når man kom ind på sådan den her størrelse agtigt. Tror du så det ville, øh, øge sådan oplevelses, eller opdagelses ... ?

NW:

Jeg tror det ville ... Jeg tror det gode ved den her er nu at man kan se med det samme hvad man skal. Øhm, og man ved at det her der skal man vælge. Hvis man ikke kan se noget på dem, så skal man først finde ud af at man kan zoome, og det fandt jeg ikke ud af. Øhm. Jeg tror det er rigtigt godt, at det skriger til en med det samme. Helt simpelt, hvad man skal gøre. Og, øhm, Der tror jeg det... Der hjalp det i hvert fald mig rigtigt meget de der røde pletter her. Fordi så kunne man, det var der man skulle vælge.

NPH:

Ja, hvis man bare så et kort og der ikke var noget til at starte med ...

NW:

Ja, så er det et rigtigt flot kort, men jeg ved ikke jeg skal vælge. Og øh.

RB:

Med den størrelse ville det måske også blevet lidt voldsomt, hvis de øh øh, når alle sitesne kommer på.

Bilag C. Interview

NW:

Arh, det ved jeg nu ikke. Det tror jeg heller ikke. Det tror jeg faktisk ville være fint. Fordi så tæt ligger de heller ikke. Øhm, men derfor kunne man måske altså zoome ind på dem alligevel. Hvis man ville finde ud af helt præcist hvor i Køgebugt, eller Askø, eller .. Men men øhm .. Men det er også fordi at så ligger de meget tættere. Når de er så langt fra hinanden, så er jeg faktisk lidt ligeglad, så vil jeg hellere ...

RB:

Ja, så er det jo ikke nødvendigt.

NW:

Men det fungerer rigtig godt synes jeg det der med at man kan vælge.

22:10 "interview" slut.

C.4 Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

Interviewer: Sebastian A. V. Jakobsen (SJ), Rune B. Barrett(RB), Patrick Østergaard(PØ) og Andreas O. Thomsen (AT).

Interviewee: Mette Jerbo(MJ).

Yderligere deltagende: Andreas K. Bloch (AB)

RB:

Og det er også sådan noget. Det kan muligvis godt være vi ændre det, fordi det ikke er så intuitivt.

MJ:

Når så jeg skal bare røre, jeg skal ikke gøre noget mere?

RB:

Ja, Ja så skal man jo holde den inde lidt, med en finger, før den virker og det, det skal vi nok også.

MJ:

Det skal man lige opdage.

RB:

Ja, ja det er det. Det skal vi måske have lavet nemmere at opdage.

MJ:

Ow shit, ej undskyld.

RB:

Yes, så er det faktisk den store skærm her, der kan styres med denne her.

MJ:

Hvad skal man?

RB:

Den der bold, det er ikke meningen den skal være der. Vi sad lige redigeret nogle ting, og glemte at slå nogle ting fra.

Bilag C. Interview

MJ:

Hva er jeg så nu i gang med at puste væk, eller ved at støvsuge?

RB:

Nu er du, nu er du. Ja, det det er. Du er i gang med at suge det op.

MJ:

Okay.

RB:

Ja nu er du, nu er du arkæologen nede i vandet, og har fundet et interessant site.

MJ:

Hmm.

MJ:

Kan man så gå over det hele eller hvad?

SJ:

Måske.

MJ:

Næh det er dåse, hvad skal man så suge den op?

RB:

Ja der skulle måske ske et eller andet når man finder en af dem også. Det er sådan et af de ikke interessante objekter.

MJ:

Man skal gøre det meget længe før der sker noget var?

SJ:

Den virker i virkeligheden rigtig godt, hvis man holder den 10 cm fra. Det ehm, det vidste du bare ikke.

MJ:

Nej. (rømmer sig)

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

SJ:

Når du så har fundet noget der måske kunne have en interesse. For eksempel guldmønter. Så kommer de frem deroppe.

MJ:

Nårh. Det skal man så også lige opdage.

SJ:

Jarh.

MJ:

Det kunne være nice, hvis det kom frem her for eksempel, fordi man er jo rimelig fokuseret her.

SJ:

Ja.

RB SPILLER DEN SYVENDE SYMFONI PÅ TASTERNE

MJ:

Kommer der så også noget om, hvad det er man finder?

RB:

Ja, det er tanken, at der skal være en meget minimal mængde information. eller muligvis slet ingen information om hvad det er. Men måske noget tekst på en måde.

MJ:

Ja, for jeg kan f.eks slet ikke gennemskue hvad det er jeg har fundet.

RB:

Nej.

RB:

Det er også meningen. Det kan man jo heller ikke når man ligger under vandet. men heldigvis.. det kan jo være der er nogle master arkæologer der kan komme med et kvalificeret gæt.

Bilag C. Interview

SJ:

Nu når du er i gang med det, og du har fundet nogle genstande der tydeligvis har relevans når de er blevet vist heroppe. Kan du så prøve og forklare lidt om fordybelsen?

MJ:

Det er ikke et ledende spørgsmål overhovedet kan jeg høre.

SJ:

Nej.

MJ:

Nej.

SJ:

Nej, men nu har du jo prøvet og finde forskellige ting.

MJ:

Ja, hvad spurgte du om undskyld?

SJ:

Om du kan prøve sådan, og forklare lidt om niveauet af fordybelse?

MJ:

Hmm, jeg tror ikke helt jeg forstår hvad du mener.

SJ:

Puha.

MJ:

Niveauet af fordybelse?

SJ:

Ja.

MJ:

Altså i genstanden? eller i aktiviteten?

SJ:

Ja i aktiviteten, det kan man godt sige.

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

AO:

Bliver man fanget af den?

MJ:

Æhh, nej. det, altså. det der med, at jeg ikke automatisk ved hvor langt den skal holdes fra, der støvsugede jeg meget længe, tænkte jeg, hvor jeg tænkte der sker ikke og en fis, og der havde jeg nok mistet tålmodigheden og gået min vej. Og tænkt, jeg gider ikke, altså.

SJ:

Ja.

RB:

Det tog for lang tid.

MJ:

Det tog for lang tid ja altså, men nej. og tænkt, øhh det gider jeg ikke bruge tid på. det virker ikke, det ville jeg nok have tænkt.

PØ:

Så fordi den ikke er så, altså fordi den ikke responderer så godt?

MJ:

Ja altså, jeg tænker at når jeg skal støvsuge noget, så gør jeg sådan her.

RB:

Ja altså tættere på.

MJ:

Og det kunne man så ikke, og så skete der ingenting. Og så bliver jeg bare, og så tænker jeg bare. Så virker lortet ikke, så går jeg videre til den næste. og det er sådan noget, når der er interaktive ting på museer så er min tålmodighed super kort, det skal bare virke eller være nemt at forstå. øhm.

PØ:

Jamen det er godt fordi, hvis det så er børn der skal bruge den så må vi gå ud fra at deres tålmodighed er endnu kortere.

Bilag C. Interview

MJ:

Det ved jeg sku ikke, det kan også være de fatter det bedre. Det ved jeg ikke, hvis de er vandt til sådan noget.

SJ:

Det tror jeg er, meget ja tilfældigt.

PØ:

Men altså meningen er at, faktisk Andreas vil gerne have der kommer så lidt information som muligt om tingene, så at man ligesom får lyst til at tage dem med videre. Det er meningen at der skal være andre installationer.

MJ:

Okay.

PØ:

Hvor man så kan få mere information tingene.

MJ:

Så man skal tage den her ting, på en eller anden måde?

PØ:

Så kommer de f.eks til at være der.

MJ:

Nårh.

PØ:

Fysisk.

MJ:

Ja okay fedt.

PØ:

Så kan man tage dem med videre, og så undersøge mere om tingene.

MJ:

Ja okay, så når man har fundet et eller andet, så tænker man arhh det var den der lå der

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

SJ:

Ja.

MJ:

Det er fedt at det bliver konkret, syntes jeg. og det bliver noget man kan agere med.

SJ:

Hvis man forestiller sig øhh.atdu havde fået en introduktion til hvor langt du skulle holde den fra og sådan noget.

MJ:

Mhm.

SJ:

Hvordan synes du så troværdigheden er i installationen, altså det med at man er under vand, og man finder ting?

MJ:

Det syntes jeg er meget fedt, jeg har bare ikke en fornemmelse af, ikke at jeg støvsuger, men at jeg puster det væk.

RB:

Ja.

SJ:

Ja.

PØ:

Ja.

MJ:

Men at jeg puster det væk. Jeg ved ikke om det er vigtigt eller.

RB:

Det er jo lidt vigtigt, og du er jo faktisk ikke den første der har sagt det.

MJ:

Nej okay, men nu også altså,nu kan jeg, jeg vil gerne puste på den der kant sådan,eller sådan væk med det der.

Bilag C. Interview

RB:

Ja.

MJ:

Men jeg skal her ind, før der sker noget. Eller sådan, og så tænker jeg igen. Hvorfor fanden sker der ikke noget?

RB:

Ja.

MJ:

Når jeg skal derind godt så, altså, forstår i hvad jeg mener?

SJ:

Mhm.

MJ:

Det er så fordi jeg tror at den puste, og ikke suger

PØ:

Jamen, det giver også meget god for den skulle helst(forsvundet i støj)

MJ:

Når jeg har fundet noget, det havde jeg ikke engang set.

MJ:

Det kan godt være, at det er en spande det der.

RB:

Det er faktisk ikke din skyld.at du ikke opdage det der skelethoved.

PØ:

Nej fordi, det ligger egentlig heroppe, gør det ikke?

RB:

Jo.

MJ:

Er det der en spande? Eller er det ikke noget?

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

PØ:

Der er.

MJ:

Der er et eller andet der.

PØ:

Det tror jeg måske bare er bunden.

RB:

Ja, det er bare, det er terræn.

PØ:

Der var måske lige før, men den er så forsvundet.

MJ:

Det der er da noget, men.

PØ:

Den kom frem lige før.

SJ:

Den ligger faktisk lige der.

MJ:

Jamen er det, der ligger et kranie der.

PØ:

Jamen det er. vi har haft noget med at få tingene fri.

RB:

Vi kan lige prøve og lave et hurtigt reset her. for at finde det der kranie igen måske.

SJ:

Jeg tror den kommer nu.

RB:

Jamen der er jo, der er jo faktisk sket en fejl her. Og det er jo ikke meningen at kraniet allerede skal være på skærmen. Det skal jo graves frem først.

Bilag C. Interview

MJ:

Jeg kan heller ikke nå det hele jo.

PØ:

Nej, det er også lidt svært nu.

MJ:

Den her er også lidt uhåndterlig, det er måske også sådan det er på havbunden, det ved jeg ikke

MJ:

Virk nu, sug så. Den er sådan lidt langsom. Jeg bliver utålmodig kan jeg mærke.

RB:

Ja.

SJ:

Pga. det der med, at det ikke er som du forventer af output.

MJ:

Ja, virk nu for helvede ha ha. Ja. Nu troede jeg lige jeg havde fundet ud af, at man bare skulle ind over, men man skal så bare, nu skal jeg endnu mere ind over, end jeg skulle før.

SJ:

Nu står skærmen også lidt i vejen.

RB:

Ja, det er ikke meningen at skærmen skal stå der, der er bare ikke kabler til at, den kan stå andre steder lige nu. Der ligger ihvertfald en ting mere lige der.

MJ:

Ja.

RB:

Men den er selvfølgelig også lidt svær at komme til.

MJ:

Men jeg tror ehh. Altså jeg tror jeg har brug for nogle flere forklaringer altså.

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

SJ:

Ja sådan indledende?

MJ:

Ja sådan, og når jeg så har fundet den der . Så skal jeg ligesom også opdage at jeg skal finde den der. eller sådan.

SJ:

Ja.

AB:

Ja, du skal faktisk gå over til det bord, hvor standen er.

MJ:

Når, det skal jeg jo ligesom vide på en eller anden måde ellers ville jeg måske bare gå videre og ikke opdage det.

AB:

Den der tekst, det kan også godt være der skal være noget lyd. der siger et eller andet om skærmen.

MJ:

Ja det tror jeg måske er federe, for jeg er jo fokuseret her. så kan man opdage der sker noget her.

RB:

Herovre fra info skærmen, eller hvad vi kalder den.

MJ:

For hvis der sker noget på en skærm, det opdager jeg ikke. Men hvis der er nogen der siger til mig: Hey du har fundet et kranie, fedt tag det over på bordet. Super, så gør jeg det.

PØ:

Så, få opmærksomheden væk fra.

MJ:

Ja altså. og nu er mit, nu er mit visuelle helt hernede.

SJ:

Mhm.

Bilag C. Interview

MJ:

Så. Nu ligger de så oven på hinanden, får jeg så ikke noget information eller hvad?

RB:

Nej. Ja det er, ja der er jo kun nogle enkelte af objekterne der kommer op. det er kun de interessante objekter der kommer op på skærmen.

MJ:

Jo, men den der pind. Er det ikke den der ligger derovre?

RB:

Den var faktisk oppe, men så dukkede der en dåse frem, som så optog pladsen. Og det er en lille, det er en fejl der ikke er.

MJ:

Okay.

RB:

Endnu en fejl. Ja ja.

MJ:

Godt det ikke er det færdige produkt jeg tester.

RB:

Ja.

PØ:

Nej den skal først være færdig om to måneder, eller sådan noget.

RB:

Ja.

MJ:

Ok, dog.

SJ:

Hvad så med troværdigheden over for de genstande, f.eks kanonkuglen?

MJ:

Når det er en kanonkugle, godt så.

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

SJ:

Ja, nu afslørede jeg det. Det er en kanonkugle.

RB:

Men det er jo heller ikke meningen at man skal vide det så.

SJ:

Nej det er det ikke.

MJ:

Altså det ved jeg ikke, det havde jeg måske gættet hvis jeg havde brugt tid på det.

SJ:

Ja det er så meningen man skal tage den med hen, for så har man jo fundet noget.

MJ:

Ja så har man jo opdaget noget.

RB:

Det her, det skulle så være den rigtige, eller næsten den rigtige.

MJ:

Det er så en nutidig en eller hvad?

AB:

Nej, det er en fra dannebrog.

MJ:

Når.

AB:

Den er god nok.

MJ:

Med den der tut på?

AB:

Ja

Bilag C. Interview

MJ:

ha, jamen det er jo sådan en rigtig bombe.

AB:

Ja, den har været sprængfyldt med sort krudt.

MJ:

Ha ha hvor sjovt. Det syntes jeg er sjovt. Hov.

RB:

Nej, det var mig.

MJ:

Når det var dig.

RB:

ja, jeg skulle bare lige.

MJ:

Øhm. altså jeg tror hvis jeg fik. Jeg er nok sådan en der har rigtig meget brug for at blive guidet lidt fordi, altså. Jeg finder jo de der genstande, men hvad det er, og hvad man skal med det. Det har jeg nok lidt brug for noget hjælp til tror jeg. Men noget auditivt, det ville være super tror jeg.

RB:

Ja.

MJ:

At man ligesom fik. Jeg, jeg var ude på lejre museum for ikke så længe siden de har fået en ny, ikke et nyt museum, men en ny udstilling og så var der nogle af montrene, når man gik derhen og bare stillede sig for at kigge på genstandene, så var der pludselig noget der snakkede til en. Og det fungerede ret fedt. Det var sådan lidt, wow fedt, så fik man lige en forklaring på hvad det var man så, og det der med at noget pludselig taler til en. Det syntes jeg er meget sjovt.

PØ:

Ja.

MJ:

Mhm, det kunne jo godt være en god ide, at gøre det. Ja.

C.4. Interview med formidler Mette Jehrbo - Foretaget af flere d. 15-11-2016

SJ:

Ja, jamen jeg tror at det er.

MJ:

Er der mere i skal vide?

RB:

Nej.

SJ:

Ja, måske dit navn.

MJ:

Ja, jeg hedder Mette Jerborg.

SJ:

Ja. Og din stilling?

MJ:

Jeg er formidler, her på museet ehm.

SJ:

Okay.

MJ:

Og underviser

PØ:

fedt.

SJ:

tak fordi vi måtte optage.

C.5 Interview med Marinarkæolog Andreas K. Bloch - Foretaget af Sebastian A. V. Jakobsen d. 15-12-2016

Transkribering d. 15/12 - 2016 - Interview med Andreas

24min 37sek

Interviewer: Sebastian Jacobsen

Interviewede: Andreas Kallmeyer Bloch

SJ:

Vil du starte med at sige dit navn og din stilling

AB:

Jeg hedder Andreas Bloch jeg er museumsinspektør og marinarkæolog på vikingeskibsmuseet

SJ:

Sweet, nu har du lige prøvet genstanden eller øøh, installationen og hvor troværdigt synes du så at undervands miljøet er?

AB:

Jeg synes det er (pause) jeg synes det er tilpas troværdigt

SJ:

Ja

AB:

Øøøøh, hvis man skal se på det i forhold til de reelle forhold vi har når vi graver så er haøbunden, det man kommer ned til lidt specielt, men jeg tænker som, altså som et spil og som det man kan forvente af en digital model af det, så synes jeg det er rigtigt tæt på.

SJ:

Ja så man kommer tæt på den oplevelse du har fået i felten

AB:

Ja altså man kunne godt ønske at der var mere af den der fornemmelse af sug, at sandet faktisk hvirvlede op mod suget, hvis øh man kunne lave det. Det ville...

SJ:

Ja så man ikke får fornemmelsen af sug frem for pust

AB:

Ja

SJ:

Ja og hvor fordybende den er øh når man bruger den? bliver man draget ind i universet om så man kan sige?

AB:

Ja det synes jeg man gør, ens fokus er meget på det altså man går ind i oplevelsen øhm og det også det der gør det så svært at når man finder sådan en genstand at man ikke lægger mærke til at den kommer op på væggen, fordi man er dybt inde det og koncentreret om at flytte sand øhm jeg tænker at øh at man skal have mulighed for så nemlig at kunne løsrive sig for så at komme over og se hvad det er man så har fundet og hvad man så skal bruge en til. Men jeg synes den er fordybende og jeg synes at øh det giver en fornemmelse af det vi gør med det der som vi snakkede om før, måske den der mere den sugende fornemmelse ville være god at finde frem til øhm jeg kunne godt tænke mig at der var mere sådan et konstant tempo på det man man graver efter fordi det kan virke forvirrende at den hopper en lille smule, og at man har en fornemmelse af at man graver i et konstant tempo.

SJ:

Ja også ville det måske også give en virkelighedsfornemmelse i det der med at den pludselig hakker eller hvad ved jeg ikke

AB:

Jo, altså jeg tror det der med en lidt mere, stadig kraftigere suge evne øhm og at man kan holde sit sug direkte på glaspladen og det så har den optimale effekt, det tror jeg er.. min fornemmelse er at det er det mest intuitive.

SJ:

Ja, jamen det tror jeg du har ret i, det har også vist sig andre gange. Tror du folk vil kunne få fornemmelsen af at være arkæolog på sådan en opdagelse i sådan en interaktion med installationen?

Bilag C. Interview

AB:

Det tror jeg sagtens, som sagt så tror jeg også at vi skal have de der hjælpe spor på øh for at give dem den oplevelse af at det de så finder det er noget de skal bruge til noget andet, så jo der er den der søgen efter guld eller hvad man skal kalde det altså fornemmelsen af at man leder efter noget.

SJ:

Ja sådan et eventyr agtigt

AB:

Ja, øh men nu har jeg jo set nogle af testene så jeg tror vi skal hjælpe folk lidt mere til en forståelse af hvad de skal

SJ:

Og jeg kommer også til en erkendelse af at øh det ved i jo egentlig også når i går ud at i har et område i skal finde noget og så skal i tilbage og gøre det som arkæologer gør

AB:

Ja

SJ:

Som vi så egentlig ikke viser og hvordan synes du det er det der med at vi viser den her lille del af arkæologien?

AB:

Jeg synes det givet et rigtigt godt indblik fordi det er jo, vi har selvfølgelig mange processer inden at vi går ud og er der hvor vi så er fordi kræver jo en del undersøgelse inde vi er der hvor vi er, det kræver jo at vi kigger sitescan data igennem og vi kigger på topografien og kan så se ud fra topografien hvor der er størst sandsynlighed for at vi har en bo plads eller et vrag eller hvad det er vi har øhm og derefter og så har vi nogle yderligere rekognosceringer og så er det først det sidste trin der er at vi suger, så noget af det sidste vi gør øhm men det ville fylde for meget som det er nu at have det med, det ville være for stor en mundfuld for at nå frem til det. Og så ville det ende med en installation man ville blive ved i 20 minutter for at finde sit (site), jeg er ikke i tvivl om at det kan laves men man ville simpelthen bruge for lang tid på det.

SJ:

Ja, jeg ved ikke om det er det rigtige ord at bruge men er det så okay at vise bare det?

AB:

Ja det synes jeg fordi, det må lægge til en anden installation at gøre det indledende arbejde fordi den her flade den egner sig rigtigt godt til at grave på og så kan man sige i fremtiden der kunne det være sjovt at tage de indledende processer i andre typer installationer sådan at man faktisk kan finde ud af "hvorfør når jeg frem til det her?". Det bliver igen kompliceret af at hvis det er et vrug så er det én måde, hvis det er en stenalderboplads er det en anden måde.

SJ:

Ja okay så det simpelthen en "Keep it simple sir"

AB:

Det er jo lige præcis her vi starter når det begynder at blive spændende så har vi færtten af at her er noget som vi gerne vil lede efter , og så suger vi så på den måde er det det mest interessante punkt i den process som så kan gøre interaktiv

SJ:

Tror du så også at det der med at man får noget information om objekterne det er godt hvis det bliver indledt på en god måde? sådan introduktions mæssigt?

AB:

Ja det er jo også det gerne skulle lægge op til det er jo bare at den skal give den der oplevelse af at man personligt har fundet den her genstand og fordi man selv har fundet den så er håbet jo også at får lyst til at undersøge hvilke metoder man så kan bruge for at få resultater ud af den her genstand, hvad bidrager den med at for at fortælle historien om det her sted den er gravet?

SJ:

Kunne man så føle sig lidt som sådan en opdagelsesrejsende på en eller anden måde?

AB:

Ja, det er jo lidt det jeg tænker at den skal give en fornemmelse af at man er den første der ser det her i tusind år og nu vil man gerne vide hvad det er, lige som når man graver rigtigt. så vil man gerne være den første der finder noget

Bilag C. Interview

SJ:

Ja som så ikke er blevet set i, hvad ved jeg, 700 år

AB:

Ja, jeg skal lige svare på at det er mig der henter børn

SJ:

Det skal du bare gøre

AB:

Nå, ja men jeg synes ikke den skal vise mere end hvad den gør andet at den skal vise folk igennem hvad de skal gøre med den nu men den skal ikke vise mere, fordi dens funktions skal også begrænses fordi der skal egentlig helst rigtig mange mennesker igennem så i og med at tilføje flere funktioner ville være forkert så kunne man lige så godt have det på en skærm der hjemme på sin computer

SJ:

Ja eller studere [Arkæologi?]

AB:

Ja

SJ:

Tror du man for fornemmelse naf at man har lært noget om selve arkæologien?

AB:

Ja jeg tror at det der med at folk som udgangspunkt har svært ved at forstille sig hvordan vi graver under vand så på den måde lærer folk noget om vores metode. og så forhåbentlig når man så tager sin genstand videre så lærer man også noget om selve genstandene men lige præcis her er det jo vores metode, at vi graver med et sug.

SJ:

Ja det praktiske

AB:

Ja, og det synes jeg i den her sammenhæng er ret vigtigt og så er det lidt den her der pirrer med at man selv er en del af at finde historien, så det fungerer den til. og så behøves den heller ikke at skulle gøre for meget, den må godt begrænses til at vise at jeg er med ti lat opdage historien og det jeg trækker op fra havbunde det kan jeg bruge til det her og det så finder man så i det videre forløb.

SJ:

Du nævner også selv det her med det tidsmæssige aspekt, som jeg faktisk synes var en rimelig god pointe, det at man kommer ond og skal ære her i noget tid men vil også gerne prøve det andet. lad være med at gøre det sværere end det er. Nu har du selvfølgelig været med i udviklingen men hvordan synes du om dén måde at formidle arkæologi på?

AB:

Jeg synes det er sjovt, jeg synes jo absolut også da hele projektet startede at arkæologi i sin grundkerne er dødhamrende kedeligt for almindelige mennesker, det er noget af det tørreste stof vi ligger inde med på museer, ej det kan godt være at madam blå kaffe kande også kan være i sit udgangspunkt kedelig men jeg tror ikke på at almindelige mennesker vil synes af hvis man bare bliver præsenteret af en genstand og der står noget tekst under den og siger at det her det er en kanon kugle fra vraget af dannebrog, så er det bare en kanonkugle fra vraget af dannebrog.

SJ:

Man er ikke tvunget til at tænke videre over det man får bare den og det er det du ser

AB:

Ja, så den måde at formidle marine arkæologi for mig gør at jeg tror på. og det så også det hele projektet skal prøve at munde ud i at det faktisk giver en bredere befolkning mere lyst til at gå dybere i historien.

SJ:

Forhåbentligt, jeg håber det ikke kommer til at læne sig for meget op ad det forrige spørgsmål men man synes du er det vigtigt at formidle omkring arkæologien når du nu siger at godt kan virke kedeligt for den generelle borger

AB:

Der er jo flere ting i det og det jo også enkelthistorien også den forståelse af at der lægger altså spændene historie under ens fødder og også når man er ude og svømme at der ligger gamle bopladser og vores forfædre for mange tusind år siden har boet ude i det der nu er vand, og den der fornemmelse af at man er et sted hvor folk i 14.000 år har langsomt bosat sig og så er det jo det igen med hvad har det for en betydning for folk hvor folk som udgangspunkt, eller mange er, ret lige glad med historien så tror jeg det det her kan hjælpe dem på vej ti lat tænke, nå ja det er jo egentlig ret spændene og det er også med til at retfærdiggøre mit arbejde og der er en grund til at vi synes det er spændene at fortælle hvad vores forfædre har lavet fordi det er jo med til definere hvem vi er og hvor vi er. så det tænker jeg at det er det meste af udstillingen.. en form for retfærdiggørelse af at vi får løn for at lægge og grave på haobunden.

SJ:

Og det er interessant at du lige selv kommer ind på det. opfatter du sådan at offentligheden eller samfundet har en indflydelse på dit job? måske implicit?

AB:

Det har det jo helt sikkert, det er jo også den vej at museer går. Vi udstiller jo ting fordi det er jo et løvkrav at vi udstiller det arbejde vi laver men mit indtryk, det er at rigtigt mange af dem jeg kender, også min familie stiller spørgsmålet om "er i ikke snart færdige?" "er der virkelig grund til at finde mere?" hvor at man nogle gange føler at man arbejder for måske 1-2% af befolkningen som overhovedet har en interesse i det her og der tænker jeg at formidlingen er jo netop en retfærdiggørelse af hvad vi gør fordi det er både forsknings artikler og lette formidlinger [pause] hvor jeg tænker at det her det er jo.. ja.. det retfærdiggør at vi har det job vi har, fordi det giver mening for rigtigt mange mennesker og desto flere mennesker vi kan få det til at give mening for kan jo understøtte at vi har det job vi har ellers kan vi lige så godt lade være. og det er der rigtigt mange mennesker der synes, selv min fætter synes det er det tåbeligste arbejde der findes der findes at blive ved med at grave efter det fordi det er jo altid i vejen for alt muligt andet.

SJ:

Men tror du så at, nu sagde du tung formidling og let formidling, tror du så i virkeligheden at det måske er bedre at det er den lette formidling man burde fokusere mere på i fremtiden?

AB:

Tja, man skal jo satse bredt, vi har jo en forpligtigelse til at lave de tunge både artikler og store bøger for at kunne komme videre, man kan sige at den lette formidling, og for at kunne komme videre i historien og for at kunne bygge ovenpå sådan at vi til sidst har det samlede puslespil og det der billede af, hvordan levede vores forfædre for 10.000 år siden og der skal vi jo lave den tunge formidling men jeg synes det er vigtigt at vi satser på den lette formidling fordi det jo også det igennem at vi får vagt folks interesse for faget og det er også der igennem at vi får nye arkæologer til at uddanne sig og til at have lyst til at lede efter tingene og hvis man rent faktisk kan skabe en skare af små arkæologer, hvis flere danskere eller flere mennesker følte et ansvar for deres fælles historie så ville jo også få langt flere indrapporteringer med ting der er at finde der ude fordi folk så rent faktisk forstå at hvis de fandt et stykke flint eller en eller anden genstand for vores vedkommende ude i hovedet, for ander på land så ville vi have bedre mulighed for at kunne lave et danmarks kort der kunne kortlægge vores forfædres verden, laden og gøren så hvis vi rent faktisk kunne uddanne små arkæologer.

SJ:

Lad os blive i samfundsområdet, selvom det er en smule flyvsk og svært at svare på men har du sådan kunne mærke ændringer i samfundet ud fra Film eller bøger er bliver udgivet?

(Arkæologisk emne i tankerne)

Bilag C. Interview

AB:

Det har det helt klart, man kan sige at serien, både "Vikings" og faktisk også "Game of Thrones" og "The last kingdom" er en medvirkende faktor i at der er en større interesse omkring vikinger og måske skræmmende nok også en nationalistisk følelse af at vi har en fremgang på museer fordi at vi fordi folk er bange for at vi bliver løbet over ande af fremmede kulturer. så det har en tydelig effekt, når det kommer til det som er den største udfordring det med at vores stenalderbefolkning og de første mennesker i danmark ikke har den store tiltrækningskraft i forhold til at det er viking og at de her serier er kommet spiller en stor rolle positivt i besøgstal men også interesse for arkæologi og det vi finder.

SJ:

Så på en måde kan man sige at når sådan noget sker ude i samfundet, såer det sådan implicit med til at brande arkæologien?

AB:

Ja, det er en positiv branding af arkæologien i at det også er medvirkende ikke at de bruger alt det vi rent faktisk har lært når de laver de her serier desværre men så giver det lige pludselig mening for folk, det er det der med at prøve og gøre historien nærværende for folk og give folk mulighed for at spejle sig i historien og spejle sig selv i vikingerne hvor vi har et efterslæb på stenaldersiden, der er det meget meget svært at få folk til at spejle sig selv og det er der hvor jeg tænker at sådan nogle installationer som denne her gør det muligt at se sig selv i en sammenhæng med en stenalder boplads fordi man selv kan være med til at finde og den er jo rørende uinteressant som bare de her genstande der kan ligge i en montre og man kan fortælle sig selv "nå ja der er nogle der har fundet" men hvorfor skulle det betyde noget for mig og så tror jeg at den her slags formidling og den her interaktive installation kan give den her spejling og give en følelse af at "jeg er med til det her" jeg er faktisk en del af historien, den her aktive deltagelse er utroligt vigtigt for at forstå, og overhovedet have en interesse. det er lidt nemmere med vikinger netop fordi det er fjernsynet.

SJ:

Ja så man bliver bevidst om det. Vi har dækket det her men hvordan synes du vores installation bidrager til arkæologien og er der overhovedet et behov for det?

AB:

Ja tilbage til det vi lige snakkede om med at spejle sig selv og kunne se sig selv, når jeg graver, jeg har ikke en bred interesse for alt der er arkæologisk, men hver gang jeg finder noget nyt selv, det er lidt mit udgangspunkt for at grave, hver gang jeg finder noget eller prøver at grave i en installation og finder noget som jeg ikke har set før eller som jeg ikke aner hvad er så får det lige en betydning for mig, det giver mig lyst til at finde ud af hvad, den her perle jeg har fundet hvordan hænger den sammen med den historie der var på det her sted og hvem har brugt den og hvordan passer den ind i historien? Det giver den der levendegørelse af historien som vi har svært ved at skabe altså personificeret historien med en enkelt genstand, det er nemmere hvis du har et lig og du kan se at den her person har haft pest eller eller have været spedalsk fordi at knoglerne er ædt op, det er en spændende person det her, men de her døde genstande, at bringe dem i live, det tror jeg kun man kan gøre ved at se sig selv i det her perspektiv som værende den der finder dem så det er en iscenesættelse af historien som gør sig meget moderne og op ad tiden, det gør mange af de store museer de prøver at iscenesætte historie sådan man faktisk kan sætte sig selv ind i den scene og det tænker jeg at jeres installation er med til at gøre, den er med til at sætte folk ind i den scene som er underlig diffus, hvad finder jeg under vandet, hvorfor finder jeg det overhovedet under vandet så en forklaring af hvorfor man finder det under vandet er jo spændene af finde frem til. og det er jo så vores udfordring i resten af udstillingen at få formidlet at det er, at det gør man på grund af det og det og det. Men jeres installation er med til at iscenesætte historien, og gøre det til en personligt oplevelse.

SJ:

Jeg har lige et sidste spørgsmål, jeg får lidt fornemmelsen af at der er et behov for at retfærdiggøre sit arbejde så der er en eller anden konstitution mellem arkæologien og samfundet, tror du arkæologien er nød til at appellere til samfundet for overhovedet at overleve?

Bilag C. Interview

AB:

Ja vil den jo være på sigt, hvis vi blev ved med at grave og aldrig fandt noget og aldrig ville kunne gøre det relevant for nogle så ville vi miste, specielt til søs fordi det er så dyrt at gøre det vi gør. Så ville vi hurtigt holde op med at gøre det og så ville folk sige at det er unødvendigt og så ville ikke finde de her stenalder bo pladser fordi de pr. kvadratmeter er ekstremt dyre at finde det ville have en effekt og det heldige ved vores det er at det ligger i museums hjørnet og det en statslig pligt at hvis man finder noget der er et fortidsminde at man sp skal undersøge det, og det heldige ved at være til søs er at de fleste projekter der bliver lavet til søs er finansieret af staten. Den er lidt sværere på land, hvor det kan være en privat mand der gerne vil have bygget noget og så stopper man hans projekt og det koster ham x-antal kroner, det er problematisk og så er det jo også problematisk at den person selv skal betale og det er jo den her økonomi der er det springene punkt om vi får lov ti lat blive ved med at gøre det vi gør og hvis ikke vi viser det i udstillinger så er det helt sikkert vi mister den opbakning der kan være både fra befolkningen og staten.

SJ:

Nu sagde du også noget spændene, det der med love, der er jo penge fra staten, i skal så også levere noget

AB:

Vi får faktisk ikke nogle penge fra staten man får få penge fra staten men det er betalt at bygge her det er så i vores tilfælde staten fordi det er anlæg til søs så det er staten der er bygherrer og skal betale for at bygge den ene og den anden bro og det er jo så kæmpe budgetter at vores store men ret små i de store anlægsomkostninger så på den måde er vi ikke finansieret af staten men af den enkelte bygherre, som så i vores tilfælde er staten tit, ikke altid. og man kan sige at hvis vi ikke leverer en formidling på det ene eller det andet plan så ja så man kan man sagtens forstille sig en regering der går ind og siger, det dur ikke og vi får ikke noget ud af det og så lukker vi ned. Og der kræver det også vi ved at lave udstillinger bibeholder den her forståelse af at det vi laver det giver faktisk mening og vi får resultater ud af det som kan være spændene for både stor og lille

SJ:

Og måske endda videreudvikler for at kunne appellere til så mange som muligt

AB:

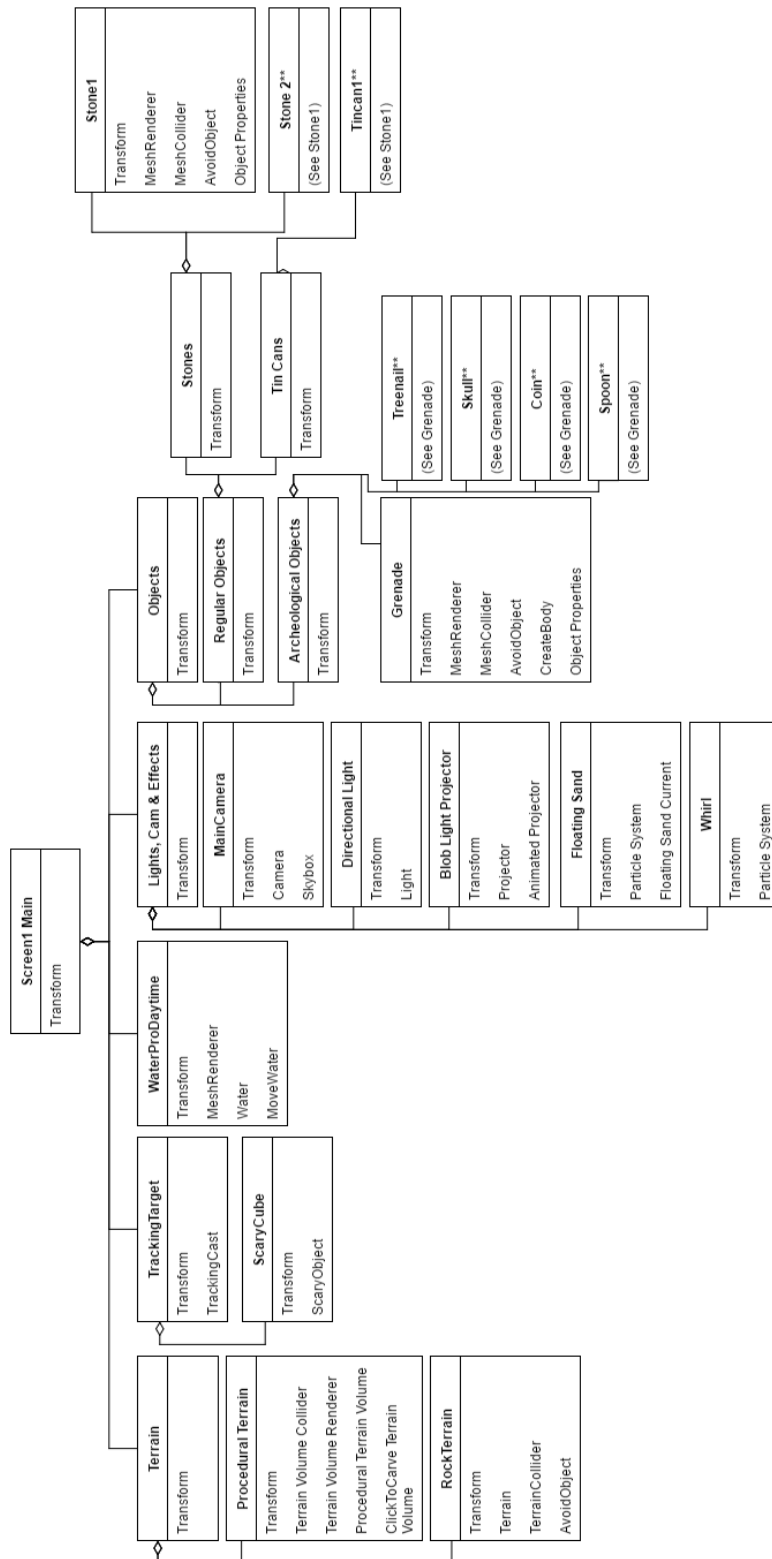
Ja, Det er også et samspil imellem at vi gerne vil have flere gæster for at udvikle et museum der er interessant for flere hvor vi har jo nogle udfordringer med roskilde borgere i forhold til at de jo har været her når de har været i skole og så kommer de faktisk ikke rigtigt igen men at vi kan have et museum der kan gøre at vi kan få flere af roskilde borgerne herved og blandet med roskilde og københavn i stedet for at vi har så mange udlændinge som vi har øhm er vigtigt fordi så kommer der selvfølgelig flere penge i kassen og det gør at museet rent faktisk kan udvikle sig og ikke bare går i stå og bliver til en død skal.

Bilag D

Diagrammer

D.1 Hierarki Diagram

D.1. Hierarki Diagram



Figur D.1: Overskrifter svarer til entiteter eller Gameobjects, Indhold svarer til komponenter eller klasser

Bilag E

Kilde kode

E.1 AddObjToWall.cs

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class AddObjToWall : MonoBehaviour {
6
7     GameObject allObjects;
8     void Start () {
9         allObjects = GameObject.FindGameObjectWithTag("AllObjects");
10        foreach (Transform child in allObjects.transform)
11            {
12                child.gameObject.SetActive(false);
13                //print(child.name);
14            }
15    }
16
17    // Update is called once per frame
18    void Update () {
19
20    }
21
22    public void setActiveObject(string name) {
23
24        foreach (Transform child in allObjects.transform)
25            {
26                if(!(child.tag == "AllObjects")) //If this is not the parent
27                    GO(GameObject) disable it.
28                    child.gameObject.SetActive(false);
29
30                if (child.name == name) { // Enable the GO with that name
31                    print("Success "+child.name + " " + name);
32                    child.gameObject.SetActive(true);
33                }
34            }
35    }
36 }
```

E.1. AddObjToWall.cs

```
33
34         if (child.parent.name == name) // Enable any potential child objects for
35             the current GO
36             child.gameObject.SetActive(true);
37     }
38 }
39 }
```

E.2 AnimatedProjector.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class AnimatedProjector : MonoBehaviour {
5
6     public float fps = 30.0f;
7     public Texture2D[] frames;
8
9     private int frameIndex;
10    private Projector projector;
11    // Use this for initialization
12    void Start () {
13        projector = GetComponent<Projector>();
14        NextFrame();
15        InvokeRepeating("NextFrame", 1 / fps, 1 / fps);
16    }
17
18    // Update is called once per frame
19    void NextFrame () {
20        projector.material.SetTexture("_ShadowTex", frames[frameIndex]);
21        frameIndex = (frameIndex + 1) % frames.Length;
22    }
23 }
```

E.3 AvoidObject.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class AvoidObject : MonoBehaviour {
5     /*At this point this script does not have any actual functionality by itself,
6     but acts as a component that will make other entities react when they are in
7     range.*/
8 }
```

E.4 BackButton.cs

```
1 using UnityEngine;
2 using UnityEngine.SceneManagement;
3
4 public class BackButton : MonoBehaviour {
5     public Camera cam;
6     RaycastHit hit;
7     void Update() {
8         if (Input.GetButton("Fire1")) {
9             Ray ray = cam.ScreenPointToRay(Input.mousePosition);
10            if (Physics.Raycast(ray.origin, ray.direction, out hit))
11            {
12                if (hit.transform.name.Contains("BackButton")) {
13                    loadSitefinder();
14                }
15            }
16        }
17    }
18 }
19 public void loadSitefinder() {
20     print("Loading SiteFinder");
21     SceneManager.LoadScene("SiteFinder");
22 }
23 }
```


E.5 Check1.cs

```

1  using UnityEngine;
2  using UnityEngine.UI;
3  using System.Collections;
4  using System;
5  using System.Runtime.InteropServices;
6
7  using System.Diagnostics;
8
9  [StructLayout(LayoutKind.Sequential, Pack = 1)]
10 public class tTouchData
11 {
12     public int m_x;
13     public int m_y;
14     public int m_ID;
15     public int m_Time;
16 };
17
18 [StructLayout(LayoutKind.Sequential, Pack = 1)]
19
20 public class Check1 : MonoBehaviour {
21     public Text t;
22     public int fingers = 0;
23     [Tooltip("IMPORTANT! This needs to be the same as the name or the build, or
24         touch will not work.")]
25     public String buildName;
26     public bool m_Initialised;
27     [DllImport("TouchOverlay", CharSet = CharSet.Auto, CallingConvention =
28         CallingConvention.StdCall)]
29     public static extern int Initialise(string Str);
30
31     [DllImport("TouchOverlay")]
32     public static extern int GetTouchPointCount();
33     [DllImport("TouchOverlay")]
34     public static extern void GetTouchPoint(int i, tTouchData n);
35
36     // Use this for initialization
37     void Start () {
38         m_Initialised = false;
39         t.text = "";
40     }
41
42     // Update is called once per frame
43     void Update () {
44
45     }
46
47     void OnGUI () {
48         //t.text = "OnGUI called "+m_Initialised;
49         string Str;
50         int NumTouch = 0;
51         if (!m_Initialised)
52         {

```

Bilag E. Kilde kode

```
51         //Str = "TouchTest";
52
53         Str = buildName;
54         if (Initialise(Str) < 0)
55             {
56                 //t.text = "ERROR";
57                 // ERROR STATE
58             }
59         //t.text = "init true";
60         m_Initialised = true;
61     }
62
63     NumTouch = GetTouchPointCount ();
64     fingers = NumTouch;
65     Str = "Number of Touch Points: " + NumTouch.ToString();
66     //t.text = "Number of Touch Points: " + NumTouch.ToString();
67     /*GUI.Label (new Rect (10,10,150,40), Str);
68     for (int p=0; p<NumTouch; p++)
69     {
70         tTouchData TouchData = new tTouchData();
71         GetTouchPoint (p, TouchData);
72         GUI.Label (new Rect (10,10 + (p+1) * 40, 200, 40),
73             "ID:" + TouchData.m_ID +
74             "Time:" + TouchData.m_Time.ToString() +
75             "(" + TouchData.m_x.ToString() + "," +
76                 TouchData.m_y.ToString() + ")");
77     }*/
78 }
79 }
```

E.6 ClickToCarveTerrainVolume.cs

```

1  using UnityEngine;
2  using System.Collections;
3
4  using Cubiquity;
5
6  public class ClickToCarveTerrainVolume : MonoBehaviour
7  {
8      private TerrainVolume terrainVolume;
9      public int range = 10;
10
11     // Bit of a hack - we want to detect mouse clicks rather than the mouse
12     // simply being down,
13     // but we can't use OnMouseDown because the voxel terrain doesn't have a
14     // collider (the
15     // individual pieces do, but not the parent). So we define a click as the
16     // mouse being down
17     // but not being down on the previous frame. We'll fix this better in the
18     // future...
19     private bool isMouseAlreadyDown = false;
20
21     // Use this for initialization
22     void Start ()
23     {
24         // We'll store a reference to the colored cubes volume so we can
25         // interact with it later.
26         terrainVolume = gameObject.GetComponent<TerrainVolume>();
27         if(terrainVolume == null)
28         {
29             Debug.LogError("This 'ClickToCarveTerrainVolume' script
30             // should be attached to a game object with a TerrainVolume
31             // component");
32         }
33     }
34
35     // Update is called once per frame
36     void Update ()
37     {
38         // Bail out if we're not attached to a terrain.
39         if(terrainVolume == null)
40         {
41             return;
42         }
43
44         // If the mouse button is down and it was not down last frame
45         // then we consider this a click, and do our destruction.
46         if(Input.GetMouseButton(1))
47         {
48             if(!isMouseAlreadyDown)
49             {
50                 // Build a ray based on the current mouse position
51                 Vector2 mousePos = Input.mousePosition;
52                 Ray ray = Camera.main.ScreenPointToRay(new

```

Bilag E. Kilde kode

```
46         Vector3(mousePos.x, mousePos.y, 0));
47
48         // Perform the raycasting.
49         PickSurfaceResult pickResult;
50         bool hit = Picking.PickSurface(terrainVolume, ray,
51             1000.0f, out pickResult);
52
53         // If we hit a solid voxel then create an explosion
54         // at this point.
55         if(hit)
56         {
57             DestroyVoxels((int)pickResult.volumeSpacePos.x,
58                 (int)pickResult.volumeSpacePos.y,
59                 (int)pickResult.volumeSpacePos.z, range);
60         }
61
62         // Set this flag so the click won't be processed
63         // again next frame.
64         //isMouseAlreadyDown = true;
65     }
66 }
67
68 public void DestroyVoxels(int xPos, int yPos, int zPos, int range)
69 {
70     yPos += (int)(range+1);
71     // Initialise outside the loop, but we'll use it later.
72     int rangeSquared = range * range;
73     MaterialSet emptyMaterialSet = new MaterialSet();
74
75     // Iterate over every voxel in a cubic region defined by the
76     // received position (the center) and
77     // the range. It is quite possible that this will be hundreds or
78     // even thousands of voxels.
79     for(int z = zPos - range; z < zPos + range; z++)
80     {
81         for(int y = yPos - range; y < yPos + range; y++)
82         {
83             for(int x = xPos - range; x < xPos + range; x++)
84             {
85                 // Compute the distance from the current
86                 // voxel to the center of our explosion.
87                 int xDistance = x - xPos;
88                 int yDistance = y - yPos;
89                 int zDistance = z - zPos;
90
91                 // Working with squared distances avoids
92                 // costly square root operations.
93                 int distSquared = xDistance * xDistance +
94                     yDistance * yDistance + zDistance *
95                     zDistance;
```

E.6. ClickToCarveTerrainVolume.cs

```
90
91         // We're iterating over a cubic region, but
           // we want our explosion to be spherical.
           // Therefore
92         // we only further consider voxels which are
           // within the required range of our
           // explosion center.
93         // The corners of the cubic region we are
           // iterating over will fail the following
           // test.
94         if(distSquared < rangeSquared)
95         {
96             terrainVolume.data.SetVoxel(x, y, z,
           emptyMaterialSet);
97         }
98     }
99 }
100 }
101
102     range += 2;
103
104     TerrainVolumeEditor.BlurTerrainVolume(terrainVolume, new Region(xPos
- range, yPos - range, zPos - range, xPos + range, yPos + range,
zPos + range));
105     //TerrainVolumeEditor.BlurTerrainVolume(terrainVolume, new
Region(xPos - range, yPos - range, zPos - range, xPos + range,
yPos + range, zPos + range));
106     //TerrainVolumeEditor.BlurTerrainVolume(terrainVolume, new
Region(xPos - range, yPos - range, zPos - range, xPos + range,
yPos + range, zPos + range));
107 }
108 }
```

E.7 ControlDiggers.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class ControlDiggers : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11     // Update is called once per frame
12     void Update () {
13
14     }
15 }
```

E.8 CreateBody.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class CreateBody : MonoBehaviour {
5
6     public float distanceToGround = 1.0f;
7     Rigidbody rb;
8     RaycastHit hit;
9     float mass, drag;
10    string name;
11    ObjectProperties op;
12    AddObjToWall activeObj;
13
14    float startTimer = 0;
15
16
17
18    void Start () {
19        activeObj =
20            GameObject.FindGameObjectWithTag("Global").GetComponent<AddObjToWall>();
21
22        rb = GetComponent<Rigidbody>();
23        op = GetComponent<ObjectProperties>();
24        mass = op.mass;
25        drag = op.drag;
26        name = op.name;
27    }
28
29    void Update () {
30        startTimer += Time.deltaTime;
31        if (rb == null && startTimer > 4) {
32            if (Physics.Raycast(transform.position, new Vector3(0,-1,0),
33                out hit))
34            {
35                if (hit.distance > distanceToGround &&
36                    hit.transform.name.Contains("OctreeNode"))
37                {
38                    addBody();
39                }
40            }
41        }
42
43    private void addBody()
44    {
45        gameObject.AddComponent<Rigidbody>();
46        rb = GetComponent<Rigidbody>();
47        rb.drag = drag;
48        rb.mass = mass;
49        activeObj.SetActiveObject(name);
```

Bilag E. Kilde kode

```
50         //rb.velocity = Vector3.zero;
51         //rb.angularVelocity = Vector3.zero;
52     }
53
54     /*
55     void OnCollisionEnter(Collision col) {
56         //print(col.gameObject);
57     }
58
59     void OnCollisionExit(Collision col) {
60         //print(gameObject.name);
61         print(gameObject.name + ", " + col.gameObject.name + " exit");
62     }
63
64
65     void OnCollisionStay(Collision col)
66     {
67         if(gameObject.name.Contains("Octree"))
68             print(gameObject.name + ", " + col.gameObject.name + " stay");
69     }
70     */
71
72
73 }
```


E.9 Current.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class Current : MonoBehaviour {
5     public float strLimit, step;
6     private float str = 0;
7     private float limitL, limitR;
8
9     public Vector3 dir;
10    //Rigidbody rb;
11    void Start () {
12        //rb = GetComponent<Rigidbody>();
13        SetVals();
14    }
15
16    void Update () {
17        if (str <= limitR)
18        {
19            step = -step;
20            SetVals();
21            str = limitR;
22        }
23
24        if (str >= limitL) {
25            step = -step;
26            SetVals();
27            str = limitL;
28        }
29        str += step;
30
31        //dir = new Vector3(0, 0, str);
32        dir = new Vector3(str, 0, 0);
33        //rb.AddForce(dir);
34    }
35
36    void SetVals() {
37        limitR = Random.Range(0, -strLimit);
38        limitL = Random.Range(0, strLimit);
39    }
40
41    public Vector3 GetDir() {
42        return dir;
43    }
44 }
```

E.10 DisplayScript.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class DisplayScript : MonoBehaviour
5 {
6     static bool started = false;
7     // Use this for initialization
8     void Start()
9     {
10         print(started);
11         Debug.Log("displays connected: " + Display.displays.Length);
12         // Display.displays[0] is the primary, default display and is always ON.
13         // Check if additional displays are available and activate each.
14         if (!started)
15         {
16             if (Display.displays.Length > 1)
17                 Display.displays[1].Activate();
18             if (Display.displays.Length > 2)
19                 Display.displays[2].Activate();
20
21             started = true;
22         }
23     }
24     // Update is called once per frame
25     void Update()
26     {
27
28     }
29 }
```

E.11 EnableCams.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class EnableCams : MonoBehaviour {
5     public Camera cam0, cam1; /*, cam2;*/           // Use this for initialization
6     void Start () {
7         cam0.enabled = true;
8         cam1.enabled = true;
9         Debug.Log("displays connected EnableCams: " + Display.displays.Length);
10        // Display.displays[0] is the primary, default display and is always ON.
11        // Check if additional displays are available and activate each.
12        /*if (Display.displays.Length > 1)
13        {
14            Display.displays[1].Activate();
15        }
16        if (Display.displays.Length > 2)
17            Display.displays[2].Activate();*/
18    }
19
20    // Update is called once per frame
21    void Update () {
22
23    }
24 }
```

E.12 Fish.cs

```
1 using UnityEngine;
2 using System.Collections;
3 //using UnityEditor.Animations;
4 //using Cubiquity;
5
6 public class Fish : MonoBehaviour
7 {
8     #region Variables
9     private GlobalFlock flock;           //Reference to the instantiated "parent"
10        script, Flock.cs
11     private FlockAIUtilities utilities;  //Own instance of utilities
12
13     private GameObject[] allFish;       //Contains all members of this flock
14     private ScaryObject[] scaryObjects;
15     private AvoidObject[] avoidObjects;
16     private bool instatiated = false;   //This becomes true when all members
17        have been retrieved from the Flock.cs instantiation.
18
19     private float initSpeed;            //Each members original speed. Not
20        changing.
21     private float initTurnTimer;        //Used for resetting the turntimer
22     private float initStateChangeTimer;
23     private float initRotateTimer;
24     //private float initInteractTimer;
25     //private Vector3 averageHeading;
26     //private Vector3 averagePosition;
27     private Vector3 scaredDirection;    //Stores the direction to move in when
28        scared
29     private Vector3 avoidDirection;     //Stores the direction to move in when
30        avoiding
31     private Vector3 avoidTerrainDirection;
32     private Vector3 headPos;
33
34     private GameObject goalPos;         //The overall position the member is
35        moving towards
36
37     private bool turning = false;       //Determines whether the fish recently
38        turned around from one of the sides
39     //private bool interacting = false;  //Determines whether the fish recently
40        interacted with a static object like terrain, or another fish
41
42     private enum States { flocking, resting, playing };
43     private States state = States.flocking;
44
45     private float exhausted;
46
47     private Vector3 targetPos;
48     private float targetSpeed;
49
50     Vector3 downSurfaceAngle;
```

E.12. Fish.cs

```
45  /* Public Variables Description
46  All of the below public variables, and the similar fields in GlobalFlock.cs,
47  change the behavior of the members. Be aware that many of them have
48  consequences on the effect of other variables as well when changed. As an
49  example, changing the amount of members in a given area without also changing
50  the neighborRange, will change the amount of flocking the members do. Another
51  example is how the speedModifier, if set to a high value, will reduce
52  flocking as well, because the members are then more likely to move out of
53  neighborRange.
54  */
55  public float speed = 2.8f;           //The approximate starting speed
56  public float speedModifier = 0.5f;   //The modifier used to make the member
57  move with varying speeds. (between 0-1f)
58  public float rotationSpeed = 2f;     //How fast the member turns
59  public float applyRulesFactor = 5f;  //Apply the 3 basic rules only one in
60  <applyRulesFactor> times. This allows fish to sometimes swim away from the
61  group, etc.
62  public float neighborRange = 5f;     //When is a member considered a
63  neighbor, and is thus eligible for grouping.
64  public float restingNeighborDistance = 5f;
65  public float restingGroundDistance;
66
67  public float tooCloseRange = 1f;     //When is another member too close.
68  public float groupSpeedReset = .7f;  //The approximate speed a group will
69  start with in the beginning of a frame. The average speed is then calculated
70  including this number.
71
72  public float scaredDist = 1.5f;      // How close to a scary object does the
73  member need to be, to start fleeing.
74  public float fleeSpeed = 5f;         //The speed modifier used for fleeing
75
76  public float avoidDist = .5f;        //How close to an avoidable object does
77  the member need to be, to swim away.
78  public float aloneSpeed = 1.5f;      //Move faster while alone
79
80  private float restDownSpeed = 0;
81
82  public float turnTimer = 2f;
83  public float stateChangeTimer = 3f;
84  public float rotateTimer;
85  //public float interactTimer = .2f;
86  public float viewDistance = 2f;
87
88  public string[] dontCollideWith;     //Strings with the names of objects you
89  dont want fish to collide with. Other fish is one example since the rules and
90  raycasts determine what to do when they get near. Glass sides are handled more
91  efficiently by RotationSwitch() as another example, and should therefore not
92  be included in CollisionDetection.
93  //public string[] visibleObjects;    //Names of objects that will be
94  processed by the fishVision function.
95  public string terrainNodeName = "OctreeNode"; // (Not needed with regular
96  terrains, use tag instead) Replace with the name (or part of it) of the
97  terrain or terrain nodes you want to hit with raycasting. Can be done simpler
98  with tags, but not when using a voxel terrain generated with Cubiquity.
99  public Transform head;
```

Bilag E. Kilde kode

```
81
82     private Animator anim;
83
84     public float exhaustionLimit = 100f;
85     public bool canRest;
86     private Vector3 restPosition;
87     private bool grounded;
88     private float restInRangeTimer;
89     private float neighborInRangeTimer;
90     private Vector3 restGoalPos;
91     private Quaternion targetRotation;
92
93
94
95     #endregion
96
97     void Start()
98     {
99         scaryObjects = (ScaryObject [])FindObjectsOfType(typeof(ScaryObject));
100        avoidObjects = (AvoidObject [])FindObjectsOfType(typeof(AvoidObject));
101
102        utilities = new
103            FlockAIUtilities(GameObject.FindGameObjectWithTag("GlobalFlock").transform);
104        GUITexture texture = (GUITexture)FindObjectOfType(typeof(GUITexture));
105
106        anim = GetComponent<Animator>();
107        //volume =
108            GameObject.FindGameObjectWithTag("Terrain").GetComponent<TerrainVolume>();
109
110        initSpeed = speed;
111        initTurnTimer = turnTimer;
112        initStateChangeTimer = stateChangeTimer;
113        initRotateTimer = rotateTimer;
114        //initInteractTimer = interactTimer;
115
116        //Set the speed to a value slightly higher or lower, so the fish will move
117        //with varying speeds.
118        speed = CalculateSpeed(1f, true);
119
120        //Choose a random initial goal.
121        goalPos = flock.getGoalPos();
122
123        exhausted = Random.Range(0, 90);
124    }
125
126    void Update()
127    {
128        //Fill the array of fish from the GlobalFlock instance. Can't do this in
129        //Start() since the array is not full yet until the last fish is
130        //instatiated. Every fish needs to do this of course
131        if (!instatiated)
132            InstatiateFish();
133
134        //What needs to happen before the state update
135        PreStateUpdate();
136    }
137
```

E.12. Fish.cs

```

132     //Apply rules depending on state.
133     if(!turning)
134         switch (state)
135         {
136             //State 1: Flocking
137             case States.flocking:
138                 FlockingUpdate();
139                 break;
140             //State 2: Resting
141             case States.resting:
142                 RestingUpdate();
143                 break;
144             //State 3: Playful
145             case States.playing:
146                 PlayingUpdate();
147                 break;
148         }
149
150     //Whatever needs to happen after the state update
151     PostStateUpdate();
152
153     //Move along the fish's local Z axis (Forward)
154     float diff = speed - targetSpeed;
155     if (diff < 0f)
156         speed += 0.1f;
157     if (diff > 0)
158         speed -= 0.1f;
159     //speed = targetSpeed;
160
161     transform.rotation = Quaternion.Slerp(transform.rotation, targetRotation,
162         rotationSpeed * Time.deltaTime);
163     transform.Translate(new Vector3(0, 0, speed * Time.deltaTime));
164     if(!(anim == null))
165         anim.speed = speed * .4f;
166 }
167
168 #region Pre & Post Updates
169 private void PreStateUpdate()
170 {
171     if (restInRangeTimer > 0)
172         restInRangeTimer -= Time.deltaTime;
173
174     if (stateChangeTimer > 0)
175         stateChangeTimer -= Time.deltaTime;
176
177     if (exhausted <= exhaustionLimit && state != States.resting)
178     {
179         exhausted += Time.deltaTime;
180     }
181
182     if (turning)
183     {
184         turnTimer -= Time.deltaTime;
185         if (turnTimer <= 0)
186         {

```

Bilag E. Kilde kode

```
187         turning = false;
188         turnTimer = initTurnTimer;
189     }
190 }
191 }
192
193 private void PostStateUpdate() //This will overwrite changes made by the current
    state.
194 {
195     //If the fish is out of allowed range, find a new target position within the
    area.
196     if (TurnAroundSwitch())
197     {
198         //Turn the fish around ..
199         //print("here");
200         Vector3 dir = utilities.setRandomPosInArea(flock.getSpawnArea(), .4f) -
            transform.position; //.. find a new target position
201         Rotate(dir);
202         CalculateSpeed(1f, true); // .. and randomize the speed
203         turning = true;
204     }
205
206     //If an avoidable object is nearby ..
207     if (AvoidObjectNearby())
208     {
209         Rotate(avoidDirection);
210         //CalculateSpeed(1f, true); // .. move away
211     }
212
213     //If a scary object is nearby ..
214     if (ScaryObjectNearby())
215     {
216         Rotate(scaredDirection);
217         CalculateSpeed(fleeSpeed, true); // .. flee fast
218     }
219
220     //Use raycasting to give the fish information on its surroundings
221     fishVision();
222
223     SwitchToRestStateIfExhausted();
224 }
225
226 private void SwitchToRestStateIfExhausted()
227 { //If exhausted or already resting
228     if ((exhausted >= exhaustionLimit * .85f || state == States.resting) &&
        canRest)
229     {
230
231         bool hitSomething = false;
232         float restGroundDistance = 1.7f;
233         RaycastHit hit;
234
235         Ray rayRestUp = new Ray(headPos, Vector3.up);
236         if (Physics.Raycast(rayRestUp, out hit, viewDistance *
            restGroundDistance))
237         {
```


E.12. Fish.cs

```
238         if (hit.transform.name.Contains(terrainNodeName) ||
239             hit.transform.tag.Equals("Prop"))
240         {
241             Debug.DrawRay(headPos, Vector3.up * viewDistance *
242                 restGroundDistance, Color.black, 2f);
243             hitSomething = true;
244         }
245     }
246     Ray rayRestDown = new Ray(headPos, -Vector3.up);
247     if (Physics.Raycast(rayRestDown, out hit, viewDistance *
248         restGroundDistance))
249     {
250         if (hit.transform.name.Contains(terrainNodeName) ||
251             hit.transform.tag.Equals("Prop"))
252         {
253             Debug.DrawRay(headPos, -Vector3.up * viewDistance *
254                 restGroundDistance, Color.black, 2f);
255             hitSomething = true;
256         }
257     }
258 }
259
260 if (hitSomething)
261 {
262     //If the fish needs to rest, switch to resting state
263     if (state != States.resting && stateChangeTimer <= 0)
264     {
265         state = States.resting;
266         stateChangeTimer = initStateChangeTimer;
267     }
268     if (restInRangeTimer < 3f)
269         restInRangeTimer = 3f;
270
271     //If there are a rest area nearby restGoalPos is set to that,
272     //otherwise it is set to transform.position
273     restGoalPos = flock.getRandomRestingPosInRange(transform.position,
274         7f) - transform.position;
275     bool rotate = false;
276     Vector3 dir = Vector3.zero;
277
278     //If the distance between this object and the potential goal is long
279     //enough modify the direction and set the rotate flag to true
280     if (restGoalPos != transform.position &&
281         Vector3.Distance(restGoalPos, transform.position) >
282         restingGroundDistance)
283     {
284         dir += restGoalPos;
285         rotate = true;
286     }
287     else
288         restGoalPos = transform.position;
289
290     //If the distance to the group or a surface above is
291     if (Vector3.Distance(transform.position, hit.point) >=
```

Bilag E. Kilde kode

```
284         restingGroundDistance && hit.point.y < transform.position.y)
285     {
286         dir += new Vector3(0, -.5f, 0);
287         rotate = true;
288     }
289     if (Vector3.Distance(transform.position, hit.point) >=
290         restingGroundDistance && hit.point.y > transform.position.y)
291     {
292         dir += new Vector3(0, .5f, 0);
293         rotate = true;
294     }
295
296     if (rotate)
297         Rotate(dir * Time.deltaTime);
298
299     }
300 }
301
302 #endregion
303
304 #region Collision
305 void OnCollisionEnter(Collision col)
306 {
307     //if(!interacting)
308     CollisionReaction(col, Color.green);
309 }
310
311 void OnCollisionStay(Collision col)
312 {
313     //if (!interacting)
314     CollisionReaction(col, Color.red);
315 }
316
317 private void CollisionReaction(Collision col, Color c)
318 {
319     foreach (ContactPoint contact in col.contacts)
320     {
321         bool collide = true;
322         foreach (string dontCol in dontCollideWith)
323         {
324             if (contact.otherCollider.transform.name.Contains(dontCol))
325                 collide = false;
326         }
327
328         if (collide)
329         {
330             //interacting = true;
331             //print(contact.otherCollider.transform.name);
332             Debug.DrawRay(contact.point, contact.normal, Color.green, 3f);
333             avoidTerrainDirection = contact.normal;
334             Rotate(avoidTerrainDirection);
335             CalculateSpeed(.9f, true);
336         }
337     }
```

E.12. Fish.cs

```

338         }
339     }
340 }
341 #endregion
342
343 #region Vision
344 private void fishVision()
345 {
346     RaycastHit hit;
347     headPos = head.transform.position;
348     grounded = false;
349     float drawTime = .3f;
350     //bool turned = false;
351
352     //Forward Ray
353     //Quaternion rot = transform.rotation;
354     Ray rayForward = new Ray(headPos, transform.forward);
355     if (Physics.Raycast(rayForward, out hit, viewDistance*.8f))
356     {
357
358         //Rotate to match the x and y eulerAngles of the part of the terrain hit.
359         if (hit.transform.name.Contains(terrainNodeName) ||
360             hit.transform.tag.Equals("Prop"))
361         {
362             Debug.DrawRay(headPos, transform.forward * viewDistance*.8f,
363                 Color.cyan, drawTime);
364             //print(hit.normal.x + " " +transform.position.normalized.x);
365             //print(hit.transform.tag + " or "+ hit.transform.name);
366             Rotate(new Vector3(hit.normal.x, 0, 0)); //hit.normal.x, hit.normal.z
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```

Bilag E. Kilde kode

```
380
381     bool turnedUpDown = false;
382     //Up Ray
383     Ray rayUp = new Ray(headPos, transform.up);
384     if (Physics.Raycast(rayUp, out hit, viewDistance*.5f))
385     {
386         if (hit.transform.name.Contains(terrainNodeName) ||
387             hit.transform.tag.Equals("Prop"))
388         {
389             Debug.DrawRay(headPos, transform.up * viewDistance*.5f,
390                 Color.red, drawTime);
391             Rotate(new Vector3(0, hit.normal.y, 0));
392             turnedUpDown = true;
393         }
394     }
395     //Down Ray
396     Ray rayDown = new Ray(headPos, -transform.up);
397     if (Physics.Raycast(rayDown, out hit, viewDistance*.5f) && !turnedUpDown)
398     {
399         if (hit.transform.name.Contains(terrainNodeName) ||
400             hit.transform.tag.Equals("Prop"))
401         {
402             grounded = true; //If the fish needs to rest, go in to resting state
403             Debug.DrawRay(headPos, -transform.up * viewDistance * .5f,
404                 Color.red, drawTime);
405             downSurfaceAngle = new Vector3(hit.normal.x, hit.normal.y,
406                 hit.normal.z);
407             Rotate(new Vector3(0, hit.normal.y, 0));//*.65f
408         }
409     }
410     bool turned = false;
411     //Right Ray
412     Ray rayRight = new Ray(headPos, transform.right);
413     if (Physics.Raycast(rayRight, out hit, viewDistance*0.5f))
414     {
415         if (hit.transform.name.Contains(terrainNodeName) ||
416             hit.transform.tag.Equals("Prop"))
417         {
418             Debug.DrawRay(transform.position, (transform.right) * viewDistance *
419                 0.5f, Color.blue, drawTime);
420             Rotate(new Vector3(hit.normal.x, 0, 0));
421             turned = true;
422         }
423     }
424     //Left Ray
425     Ray rayLeft = new Ray(transform.position, -transform.right);
426     if (Physics.Raycast(rayLeft, out hit, viewDistance*0.5f) && !turned)
427     {
428         if (hit.transform.name.Contains(terrainNodeName) ||
429             hit.transform.tag.Equals("Prop"))
430         {
431             Debug.DrawRay(transform.position, -transform.right * viewDistance *
432                 0.5f, Color.blue, drawTime);
```

E.12. Fish.cs

```
427         Rotate(new Vector3(hit.normal.x, 0,0));
428         //turned = true; interacting = true;
429     }
430 }
431 }
432 #endregion
433
434 #region State Updates
435 private void FlockingUpdate()
436 {
437     if (Random.Range(0, applyRulesFactor) < 1)
438     {
439         Vector3 thisPos = transform.position;
440         Vector3 anotherPos;
441
442         Vector3 groupCenter = Vector3.zero;
443         Vector3 avoidFishDir = Vector3.zero;
444
445         float groupSpeed = CalculateSpeed(groupSpeedReset, true);
446         float dist;
447
448         int groupSize = 1;
449
450         foreach (GameObject anotherFish in allFish)
451         {
452             if (anotherFish != gameObject &&
453                 anotherFish.GetComponent<Fish>().state == States.flocking)//If
454                 not this fish and the fish is in flocking state
455             {
456                 anotherPos = anotherFish.transform.position;
457                 dist = Vector3.Distance(thisPos, anotherPos);
458                 if (dist <= neighborRange) //If the current other fish is in
459                 'neighborRange'..
460                 {
461                     // .. add its' position to adjust the groupCenter, and
462                     increment the groupSize.
463                     groupCenter += anotherPos;
464                     groupSize++;
465
466                     if (dist < tooCloseRange) // If the other fish gets too
467                     close however, turn away from it.
468                     {
469                         avoidFishDir = avoidFishDir + (thisPos - anotherPos);
470                     }
471
472                     //Prepare to adjust the groupSpeed by adding in the other
473                     fish's speed.
474                     groupSpeed += anotherFish.GetComponent<Fish>().GetSpeed();
475                 }
476             }
477         }
478
479         if (groupSize > 1) // If in a group
480         {
481             //Calculate direction to turn towards the center of the group, ..
```

Bilag E. Kilde kode

```
477         groupCenter = groupCenter / groupSize;
478         // .. add in the direction of the goal..
479         groupCenter += (goalPos.transform.position - thisPos)*1.35f;
480         speed = groupSpeed / groupSize; //.. and change the speed of this
481             fish to the avarage speed of the group.
482         speed = utilities.slightlyRandomizeValue(speed, 0.3f);
483
484         //Turn towards groupCenter and the goalPos, and add in avoidFishDir
485             to turn away from nearby fish.
486         Vector3 dir = groupCenter + avoidFishDir;
487         if (dir != Vector3.zero)
488             Rotate(dir);
489     }
490     else { // If alone
491         CalculateSpeed(aloneSpeed, false);
492         Vector3 dir = groupCenter + goalPos.transform.position;
493
494         if (dir != Vector3.zero)
495             Rotate(dir);
496     }
497 }
498
499 private void PlayingUpdate()
500 {
501 }
502
503 private void RestingUpdate()
504 {
505     //Return to flocking if rested
506     exhausted -= Time.deltaTime * 5;
507     if (exhausted <= exhaustionLimit * 0.01f)
508     {
509         state = States.flocking;
510     }
511
512     //Dont rest if alone too long.
513     neighborInRangeTimer -= Time.deltaTime;
514
515     //Apply rest grouping rules
516     if (Random.Range(0, applyRulesFactor) < 1)
517     {
518         Debug.DrawRay(transform.position, Vector3.up, Color.magenta, .3f);
519         targetSpeed = utilities.slightlyRandomizeValue(initSpeed, .6f) * .4f;
520
521         //Resting group behavior
522         ArrayList restingGroup = new ArrayList();
523         Vector3 avoidFishDir = Vector3.zero;
524         foreach (GameObject otherFish in allFish)
525         {
526             if (otherFish.GetComponent<Fish>().state == States.resting &&
527                 otherFish != gameObject)
528             {
529                 float dist = Vector3.Distance(transform.position,
530                     otherFish.transform.position);
```

E.12. Fish.cs

```
529         if (dist < restingNeighborDistance)
530         {
531             restingGroup.Add(otherFish);
532             if (neighborInRangeTimer < 3f)
533                 neighborInRangeTimer = 3;
534
535             if (dist < tooCloseRange) // If the other fish gets too
536                 close however, turn away from it.
537             {
538                 avoidFishDir = avoidFishDir + (transform.position -
539                     otherFish.transform.position);
540             }
541         }
542
543         Vector3 groupCenter = Vector3.zero;
544         int size = 0;
545         foreach (GameObject restGroupMember in restingGroup)
546         {
547             groupCenter += restGroupMember.transform.position;
548             size++;
549         }
550         if (restGoalPos != transform.position)
551             groupCenter = groupCenter / size + restGoalPos - transform.position;
552         else
553             groupCenter = groupCenter / size - transform.position;
554
555         Debug.DrawRay(transform.position, groupCenter, Color.yellow, .3f);
556         //Debug.DrawRay(transform.position, groupCenter, Color.yellow, .3f);
557
558         //rotate towards groupCenter and avoidFishDir
559         Rotate(groupCenter + avoidFishDir * 1.5f);
560
561     }
562
563     if (restInRangeTimer <= 0.01 || neighborInRangeTimer <= 0.01 &&
564         stateChangeTimer <= 0.01)
565     {
566         state = States.flocking;
567         stateChangeTimer = initStateChangeTimer;
568     }
569 #endregion
570
571 #region Interactable Object Methods
572 private bool AvoidObjectNearby()
573 {
574     avoidDirection = Vector3.zero;
575     //Collider[] hitColliders = Physics.OverlapSphere(transform.position,
576         scaredDist);
577     foreach (AvoidObject ao in avoidObjects)
578     {
579         if (Vector3.Distance(ao.transform.position, transform.position) <
580             avoidDist)
581         {
```

Bilag E. Kilde kode

```
580         avoidDirection = -(ao.transform.position - transform.position);
581         state = States.flocking;
582         return true;
583     }
584
585 }
586 return false;
587 }
588
589 private bool ScaryObjectNearby()
590 {
591     scaredDirection = Vector3.zero;
592     foreach (ScaryObject so in scaryObjects)
593     {
594         if (Vector3.Distance(transform.position, so.transform.position) <
595             scaredDist)
596         {
597             scaredDirection = -(so.transform.position - transform.position);
598             state = States.flocking;
599             return true;
600         }
601     }
602     return false;
603
604
605
606     /*scaredDirection = Vector3.zero;
607     Collider[] hitColliders = Physics.OverlapSphere(transform.position,
608         scaredDist);
609     foreach (Collider c in hitColliders)
610     {
611         if (c.gameObject.GetComponent<ScaryObject>() != null)
612         {
613             scaredDirection = -(c.gameObject.transform.position -
614                 transform.position);
615             state = States.flocking;
616             return true;
617         }
618     }
619     return false;*/
620 }
621 #endregion
622
623 #region Utilities
624 private void Rotate(Vector3 dir)
625 {
626     if (dir != Vector3.zero)
627         targetRotation = Quaternion.Slerp(transform.rotation,
628             Quaternion.LookRotation(dir), rotationSpeed);
629     //transform.rotation = Quaternion.Slerp(transform.rotation,
630         Quaternion.LookRotation(dir), rotationSpeed * Time.deltaTime);
631 }
632 //
```


E.12. Fish.cs

```
631 private float CalculateSpeed(float burst, bool init)
632 { //The burst parameter in CalculateSpeed is a speed modifier. If means base
633   //speed, higher = faster, lower = slower.
634   if (init)
635     targetSpeed = initSpeed;
636   targetSpeed *= burst;
637   return utilities.slightlyRandomizeValue(targetSpeed, speedModifier);
638 }
639
640 private bool TurnAroundSwitch()
641 {
642   Vector3 x = new Vector3(transform.position.x, 0, 0);
643   Vector3 y = new Vector3(0, transform.position.y, 0);
644   Vector3 z = new Vector3(0, 0, transform.position.z);
645   Transform t = flock.SpawnArea.transform;
646
647   if (Vector3.Distance(x, Vector3.zero) > t.localScale.x / 2 ||
648       Vector3.Distance(y, Vector3.zero) > t.localScale.y / 2 ||
649       Vector3.Distance(z, Vector3.zero) > t.localScale.z / 2)
650     return true;
651   else
652     return false;
653 }
654
655 private void InstatiateFish()
656 {
657   allFish = flock.getAllFish();
658   instatiated = true;
659 }
660
661 #endregion
662
663 #region Accessors & Internal Utilities
664
665 public float GetSpeed()
666 {
667   return speed;
668 }
669
670 public void SetFlockReference(GlobalFlock f)
671 {
672   flock = f;
673 }
674
675 public GameObject GetGoal() {
676   return goalPos;
677 }
678
679 #endregion
680 }
```

E.13 FloatingSandCurrent.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class FloatingSandCurrent : MonoBehaviour {
5     Current c;
6     float curMaxStr;
7     // Use this for initialization
8     void Start () {
9         c = GameObject.FindGameObjectWithTag("Global").GetComponent<Current>();
10        curMaxStr = c.strLimit;
11    }
12
13    // Update is called once per frame
14    void Update () {
15        //print(c.GetDir().z + ", "+ map(c.GetDir().z, -curMaxStr, curMaxStr, -90,
16            90));
17        //float rotVal = map(c.GetDir().z, -curMaxStr, curMaxStr, -90, 90);
18        //transform.rotation = new Quaternion( 0,rotVal,0,0);
19        //transform.Rotate(0, c.GetDir().z, 0);
20        //transform.rotation.eulerAngles = new Vector3(0,90,0);
21        if(c.GetDir().x > .1f)
22            transform.rotation = Quaternion.Euler(new Vector3(0, 90, 0));
23        if (c.GetDir().x < -.1f)
24            transform.rotation = Quaternion.Euler(new Vector3(0, -90, 0));
25    }
26
27    public float map(float value, float  istart, float  istop, float  ostart, float
28        ostop)
29    {
30        return ostart + (ostop - ostart) * ((value - istart) / (istop - istart));
31    }
32 }
```

E.14 FlockAIUtilities.cs

```

1  using UnityEngine;
2  using System.Collections;
3
4  public class FlockAIUtilities
5  {
6      #region Used by Both
7      Transform t;
8      public FlockAIUtilities(Transform t) {
9          this.t = t;
10     }
11
12     public float slightlyRandomizeValue(float val, float modifier)
13     {
14         if (modifier > val)
15             modifier = val*.9f;
16         return val + val * Random.Range(-modifier, modifier);
17     }
18
19     public Vector3 setRandomPosInArea(GameObject area, float areaPercentage)
20     {
21
22         if (areaPercentage > 1)
23             areaPercentage = 1;
24
25         Transform t = area.transform;
26         Vector3 p = new Vector3(Random.Range(-t.localScale.x / 2 * areaPercentage,
27                                     Random.Range(-t.localScale.y / 2 * areaPercentage,
28                                     t.localScale.y / 2 * areaPercentage),
29                                     Random.Range(-t.localScale.z / 2 * areaPercentage,
30                                     t.localScale.z / 2 * areaPercentage));
31         return p/*+area.transform.position*/;
32     }
33
34     #endregion
35
36     #region Used by GlobalFlock.cs
37
38     public GameObject[] randomizeSize(GameObject[] members, float sizeMod)
39     {
40     {
41         if (sizeMod >= 1f)
42             sizeMod = 0.9f;
43
44         foreach (GameObject fish in members)
45         {
46             fish.gameObject.transform.localScale += new Vector3(
47                 slightlyRandomizeValue(fish.transform.localScale.x, sizeMod),
48                 slightlyRandomizeValue(fish.transform.localScale.y, sizeMod),
49                 slightlyRandomizeValue(fish.transform.localScale.z, sizeMod));
50         }
51         return members;
52     }
53 }

```

Bilag E. Kilde kode

```
50
51     #endregion
52
53     #region Used by Fish.cs
54     /*public float CalculateSpeed(float speed, float burst, float speedMod, float
55         initSpeed)
56     {
57         speed = initSpeed;
58         speed *= burst;
59         return slightlyRandomizeValue(speed, speedMod);
60     }*/
61
62     /*public float CalculateSpeed(float speed, float burst, float speedMod)
63     {
64         speed *= burst;
65         return slightlyRandomizeValue(speed, speedMod);
66     }*/
67     #endregion
68
69
70
71
72
73 }
```

E.15 GlobalFlock.cs

```

1  using UnityEngine;
2  using System.Collections;
3
4  public class GlobalFlock : MonoBehaviour
5  {
6      #region Refs
7      // Flocking reference, AI for Game Developers by Glenn Seemann, David M Bourg
8      // Chapter 4
9      // https://www.safaribooksonline.com/library/view/ai-for-game/0596005555/ch04.html
10     // The ApplyBasicFlockingRules() function handles the three basic flocking
11     // rules, and is originally based on Holistic3d's implementation described in a
12     // video by herself here:
13     // https://www.youtube.com/watch?v=eMpI1eCsIyM
14     #endregion
15     #region Description
16     /*
17     Description of the project
18     */
19     #endregion
20     //private Vector3 goalPos = Vector3.zero;
21     private FlockAIUtilities utilities;
22     private GameObject[] allFish; //Holds all members
23
24     public int numFish = 30; //How many members
25     public float changeGoalPosFreq = 2f; //Randomizes goalPos around
26     <changeGoalPosFreq> times in 1000 frames
27
28     public GameObject[] goals; //Contains the amount of goal meshes to
29     //use. Each goal is a common position that used to adjust the position of all
30     //members in one group. The amount of goals determines how many groups your
31     //flock can form.
32     public GameObject[] restingAreas; //Contains areas fit for resting. Use
33     //box meshes to define these.
34
35     public GameObject fishPrefab; //Whatever
36     //fish/bird/human/particle/bacteria you want. The Fish.cs needs to be attached
37     //to the prefab.
38     public GameObject SpawnArea; //The area members can spawn and move in.
39     public bool randomizePrefabSize = true; //Whether or not the size of the prefab
40     //should be randomized.
41     public float sizeModifier = .55f; //If size is randomized, this is how
42     //much they will differ from their original size
43
44     void Start()
45     {
46         utilities = new FlockAIUtilities(transform);
47         allFish = new GameObject[numFish];

```

Bilag E. Kilde kode

```
40
41     foreach (GameObject goal in goals)
42     {
43
44         goal.transform.position -= utilities.setRandomPosInArea(SpawnArea,
45             .7f);
46     }
47
48     //Instantiate all the fish at random positions, and store them in allFish[]
49     for (int i = 0; i < numFish; i++)
50     {
51         Vector3 pos = transform.position-utilities.setRandomPosInArea(SpawnArea,
52             .5f);
53         GameObject fishObj = Instantiate(fishPrefab, pos, Quaternion.identity)
54             as GameObject;
55         fishObj.GetComponent<Fish>().SetFlockReference(this);
56         fishObj.transform.parent = transform;
57         allFish[i] = fishObj;
58     }
59     //print(allFish.Length);
60
61     if (randomizePrefabSize) //Slightly modify the size of the model prefab if
62         chosen
63         allFish = utilities.randomizeSize(allFish, sizeModifier);
64 }
65
66 void Update()
67 {
68     //Change the position of each goal once in a while (Frequency determined by
69     'changeGoalPosFreq')
70     foreach (GameObject goal in goals)
71     {
72         if (Random.Range(0, 1000) < changeGoalPosFreq)
73         {
74             goal.transform.position = utilities.setRandomPosInArea(SpawnArea,
75                 .85f);
76         }
77     }
78
79     //DEBUG
80     /*string p = "";
81     foreach (GameObject f in allFish) {
82         p += (int)f.GetComponent<Fish>().exhausted + " ";
83     }
84     print(p);*/
85 }
86
87 public Vector3 setRandomPosShortcut(float mod)
88 {
89     return utilities.setRandomPosInArea(SpawnArea, mod);
90 }
91
92 public Vector3 getRandomRestingPosInRange(Vector3 pos, float detectDistance) {
93     //restingAreas = RandomizeArray(restingAreas);
94 }
```

E.15. GlobalFlock.cs

```

90     if (restingAreas.Length == 0)
91         return pos;
92     GameObject newPosGO = restingAreas[0];
93     bool found = false;
94     restingAreas = RandomizeArray(restingAreas);
95     foreach (GameObject area in restingAreas)
96     {
97         if (Vector3.Distance(pos, area.transform.position) <= detectDistance &&
98             Vector3.Distance(area.transform.position, pos) <
99             Vector3.Distance(newPosGO.transform.position, pos))
100     {
101         newPosGO = area;
102         found = true;
103     }
104     }
105
106     //return area; //+utilities.setRandomPosInArea(area,1f)*/
107     if (found)
108         return
109             newPosGO.transform.position;//utilities.setRandomPosInArea(newPosGO,
110             1f);
111     else
112         return pos;
113 }
114
115 static GameObject[] RandomizeArray(GameObject[] arr)
116 {
117     for (int i = arr.Length - 1; i > 0; i--)
118     {
119         int r = (int)Random.Range(0, i);
120         GameObject tmp = arr[i];
121         arr[i] = arr[r];
122         arr[r] = tmp;
123     }
124     return arr;
125 }
126
127 public GameObject[] getAllFish()
128 {
129     return allFish;
130 }
131
132 public GameObject getSpawnArea()
133 {
134     return SpawnArea;
135 }
136
137 public GameObject getGoalPos()
138 {
139     return goals[Random.Range(0, goals.Length)];
140 }
141
142 #region Deprecated
143 /*private GameObject[] randomizeSize(GameObject[] members, float
144     sizeMod)//allFish, sizeModifier
145 {

```

Bilag E. Kilde kode

```
142         if (sizeModifier >= 1f)
143             sizeModifier = 0.9f;
144
145         foreach (GameObject fish in members)
146         {
147             fish.gameObject.transform.localScale += new Vector3(
148                 slightlyRandomizeValue(fish.transform.localScale.x, sizeMod),
149                 slightlyRandomizeValue(fish.transform.localScale.y, sizeMod),
150                 slightlyRandomizeValue(fish.transform.localScale.z, sizeMod));
151         }
152         return allFish;
153     }*/
154     /*public float slightlyRandomizeValue(float val, float modifier)
155     {
156         return val + val * Random.Range(-modifier, modifier);
157     }*/
158     /*    public Vector3 setRandomPosInArea(float areaPercentage) {
159
160         if (areaPercentage > 1)
161             areaPercentage = 1;
162
163         Transform t = SpawnArea.transform;
164         Vector3 p = new Vector3(Random.Range(-t.localScale.x / 2 *
165             areaPercentage, t.localScale.x / 2 * areaPercentage),
166             Random.Range(-t.localScale.y / 2 *
167                 areaPercentage, t.localScale.y / 2 *
168                 areaPercentage),
169             Random.Range(-t.localScale.z / 2 *
170                 areaPercentage, t.localScale.z / 2 *
171                 areaPercentage));
172
173         return p;
174     }*/
175     #endregion
176 }
```


E.16 Goal.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class Goal : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11     // Update is called once per frame
12
13     void Update () {
14         RaycastHit hit;
15         Vector3 pos = transform.position;
16         if (Physics.Raycast(pos, -Vector3.up, out hit))
17         {
18             //print(hit.distance);
19         }
20         else {
21             transform.position = new Vector3(pos.x, pos.y+5, pos.z);
22             //GetComponent<MoveCube>();
23         }
24     }
25 }
```

E.17 LoadTimer.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class LoadTimer : MonoBehaviour {
5
6     public float time = 5;
7
8     // Update is called once per frame
9     void Update () {
10        time -= Time.deltaTime;
11        if (time <= 0)
12        {
13            print("loading screen done");
14            gameObject.SetActive(false);
15        }
16    }
17 }
```

E.18 MiddleMan.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class middleMan : MonoBehaviour
5 {
6     bool nData = false;
7
8     public void SetNData(bool d)
9     {
10         nData = d;
11     }
12
13     public bool GetNData()
14     {
15         return nData;
16     }
17
18
19 }
```

E.19 MoveGoal.cs

```
1 using UnityEngine;
2
3 public class MoveGoal : MonoBehaviour
4 {
5
6     public float speed = 30;
7     private float negative, positive;
8     private float limX, limY, limZ;
9     public float restrict = .8f;
10
11
12     public enum directions { x, y, z }
13     public directions dir;
14
15     public Transform SpawnArea;
16
17     void Start()
18     {
19         positive = -speed;
20         negative = speed;
21
22         //print(SpawnArea.localScale.x / 2 + " " + -SpawnArea.localScale.x / 2);
23
24         limX = SpawnArea.localScale.x * .5f * restrict;
25         limY = SpawnArea.localScale.y * .5f * restrict;
26         limZ = SpawnArea.localScale.z * .5f * restrict;
27     }
28
29     void Update()
30     {
31         //if (Random.Range(0, 5000) < .5f)
32         //speed = -speed;
33
34         switch (dir)
35         {
36             case directions.x:
37                 {
38                     if (transform.position.x >= limX)
39                         speed = -speed;
40                     else if (transform.position.x <= -limX)
41                         speed = -speed;
42
43                     transform.Translate(new Vector3(speed * Time.deltaTime, 0, 0));
44                     break;
45                 }
46             case directions.y:
47                 {
48                     if (transform.position.y >= limY)
49                         speed = -speed;
50                     else if (transform.position.y <= -limY)
51                         speed = -speed;
52

```

E.19. MoveGoal.cs

```
53         transform.Translate(new Vector3(0, -speed * Time.deltaTime, 0));
54         break;
55     }
56     case directions.z:
57     {
58         if (transform.position.z >= limZ)
59             speed = -speed;
60         else if (transform.position.z <= -limZ)
61             speed = -speed;
62
63         transform.Translate(new Vector3(0, 0, -speed * Time.deltaTime));
64         break;
65     }
66 }
67 }
68 }
```

E.20 MoveWater.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class MoveWater : MonoBehaviour
5 {
6     float counter = 0;
7
8     public Camera cam;
9     bool done = false;
10    public float startFieldOfView = 60;
11    float endFieldOfView, field;
12    public float goalFogDensity = 0.012f;
13    public float startFogDensity = 0.022f;
14
15    public float waterSpeed = 0.012f;
16    public float fogSpeed = 0.0001f;
17
18    float waterAcc = 0;
19
20    // Use this for initialization
21    void Start()
22    {
23        endFieldOfView = cam.fieldOfView;
24        field = startFieldOfView;
25        cam.fieldOfView = field;
26        RenderSettings.fogDensity = startFogDensity;
27
28    }
29
30    // Update is called once per frame
31    void Update()
32    {
33        if (!done) //Stop everything after 10 seconds (This script is only used in
34                //the beginning of the program)
35        {
36            //print(RenderSettings.fogDensity);
37            counter += Time.deltaTime;
38            if (counter > 1)
39            {
40                //print(RenderSettings.fogDensity - goalFogDensity);
41                if (RenderSettings.fogDensity > goalFogDensity && counter > 6)
42                {
43                    RenderSettings.fogDensity -= fogSpeed;
44                }
45                //print(RenderSettings.fogDensity + " " + goalFogDensity);
46                transform.position += new Vector3(0, waterAcc, 0);
47                if (waterAcc <= waterSpeed)
48                    waterAcc += Time.deltaTime*0.02f;
49                if (field > endFieldOfView)
50                {
51                    field -= Time.deltaTime;
```

E.20. MoveWater.cs

```
52         cam.fieldOfView = field;
53     }
54
55     if (transform.position.y > cam.transform.position.y - 10 && field <=
56         endFieldOfView && RenderSettings.fogDensity < goalFogDensity +
57         0.001f)
58     {
59         print("done");
60         done = true;
61     }
62 }
63 }
```

E.21 ObjectProperties.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class ObjectProperties : MonoBehaviour {
5     public float mass, drag;
6     public string name;
7     Current c;
8
9     void Start(){
10         c = GameObject.FindGameObjectWithTag("Global").GetComponent<Current>();
11     }
12
13     void Update() {
14
15         if (GetComponent<Rigidbody>()){
16             GetComponent<Rigidbody>().AddForce(c.GetDir());
17             //print(gameObject + "is moving towards " + c.GetDir());
18         }
19     }
20 }
```


E.22 PinSelected.cs

```

1  using UnityEngine;
2  using System.Collections;
3  using UnityEngine.SceneManagement;
4
5
6  public class PinSelected : MonoBehaviour {
7      public Transform radar;
8      public GameObject selectedObj, loadingObj;
9      public Camera cam;//, cam1;
10     public float initLoadSceneTimer = 3;
11
12     public float selectTimer = 0;
13     private bool selected = false;
14
15     private float loadSceneTimer = 0;
16
17
18     // Use this for initialization
19     void Start () {
20         loadSceneTimer = initLoadSceneTimer;
21     }
22
23     void Update () {
24         if (selected)
25         {
26             loadSceneTimer -= Time.deltaTime;
27             selectedObj.SetActive(true);
28             radar.LookAt(cam.transform.position);
29             float scale = map(loadSceneTimer,0,initLoadSceneTimer,0,1f);
30             //print(loadSceneTimer+", "+scale);
31             radar.transform.localScale = new Vector3(scale, scale, scale);
32         }
33         else {
34             selectedObj.SetActive(false);
35             loadSceneTimer = initLoadSceneTimer;
36         }
37
38         selectTimer -= Time.deltaTime;
39         if (selectTimer <= 0)
40         {
41             selected = false;
42         }
43         //print (selectTimer);
44         if (loadSceneTimer <= 0)
45         {
46             loadingObj.SetActive(true);
47             loadingObj.transform.LookAt(cam.transform.position);
48             SceneManager.LoadScene("SVS scene");
49
50             //cam.enabled = true;
51             //cam1.enabled = true;
52

```

Bilag E. Kilde kode

```
53     }
54
55 }
56
57 public void isSelected(bool isSelected) {
58     selectTimer = .3f;
59     selected = isSelected;
60
61 }
62
63 float map(float value, float istart, float istop, float ostart, float ostop)
64 {
65     return ostart + (ostop - ostart) * ((value - istart) / (istop - istart));
66 }
67 }
```

E.23 ProceduralTerrainVolume.cs

```

1  using UnityEngine;
2  using System.Collections;
3
4  using Cubiquity;
5
6  /**
7   * This class serves as an example of how to generate a TerrainVolume from code. The
8   * exact operation
9   * of the noise function(s) is not particularly important here as you will want to
10  * implement your own
11  * approach for your game, but you should focus on understanding how data is written
12  * into the volume.
13  * Please note, most of the 'magic numbers' in this code are simply found by trial
14  * and error as there
15  * is a lot of experimentation required to generate procedural terrains. Feel free
16  * to change them and
17  * see what happens!
18  */
19  [ExecuteInEditMode]
20  public class ProceduralTerrainVolume : MonoBehaviour
21  {
22      // Use this for initialization
23      void Start()
24      {
25          // The size of the volume we will generate
26          int width = 110;
27          int height = 55;
28          int depth = 70;
29
30          // FIXME - Where should we delete this?
31          /// [DoxygenSnippet-CreateEmptyTerrainVolumeData]
32          /// Create an empty TerrainVolumeData with dimensions width * height *
33          /// depth
34          TerrainVolumeData data =
35              VolumeData.CreateEmptyVolumeData<TerrainVolumeData>(new Region(0,
36              0, 0, width-1, height-1, depth-1));
37          /// [DoxygenSnippet-CreateEmptyTerrainVolumeData]
38
39          TerrainVolume volume = GetComponent<TerrainVolume>();
40          TerrainVolumeRenderer volumeRenderer =
41              GetComponent<TerrainVolumeRenderer>();
42
43          volume.data = data;
44
45          // This example looks better if we adjust the scaling factors on the
46          // textures.
47          volumeRenderer.material.SetTextureScale("_Tex0", new Vector2(0.062f,
48              0.062f));
49          volumeRenderer.material.SetTextureScale("_Tex1", new Vector2(0.03f,
50              0.03f));
51          volumeRenderer.material.SetTextureScale("_Tex2", new Vector2(0.03f,
52              0.03f));

```

Bilag E. Kilde kode

```
40
41 // At this point our volume is set up and ready to use. The remaining code
42 // is responsible
43 // for iterating over all the voxels and filling them according to our noise
44 // functions.
45
46 // This scale factor controls the size of the rocks which are generated.
47 //float rockScale = 36.0f;
48 float rockScale = 50f;//Random.Range(36f,42f);
49 float invRockScale = 1f / rockScale;
50
51 // Let's keep the allocation outside of the loop.
52 MaterialSet materialSet = new MaterialSet();
53
54 // Iterate over every voxel of our volume
55 for(int z = 0; z < depth; z++)
56 {
57     for(int y = height-1; y > 0; y--)
58     {
59         for(int x = 0; x < width; x++)
60         {
61             // Make sure we don't have anything left in
62             // here from the previous voxel
63             materialSet.weights[0] = 0;
64             materialSet.weights[1] = 0;
65             materialSet.weights[2] = 0;
66
67             // Simplex noise is quite high frequency. We
68             // scale the sample position to reduce this.
69             float sampleX = (float)x * invRockScale;
70             float sampleY = (float)y * invRockScale;
71             float sampleZ = (float)z * invRockScale;
72
73             // Get the noise value for the current
74             // position.
75             // Returned value should be in the range -1
76             // to +1.
77             float simplexNoiseValue =
78                 SimplexNoise.Noise.Generate(sampleX,
79                 sampleY, sampleZ);
80
81             // We want to fade off the noise towards the
82             // top of the volume (so that the rocks
83             // don't go
84             // up to the sky) and add extra material
85             // near the bottom of the volume (to create
86             // a floor).
87             // This altitude value is initially in the
88             // range from 0 to +1.
89             float altitude = (float)(y + 1) /
90                 (float)height;
91
92             // Map the altitude to the range -1.0 to
93             // +1.0...
94             altitude = (altitude * 2.0f) - 1.0f;
```

E.23. ProceduralTerrainVolume.cs

```
81 // Subtract the altitude from the noise.
82 // This adds
83 // material near the ground and subtracts it
84 // higher up.
85 simplexNoiseValue -= altitude*4f;
86 // After combining our noise value and our
87 // altitude we now have values between -2.0
88 // and 2.0.
89 // Cubiquity renders anything below the
90 // threshold as empty and anything above as
91 // solid, but
92 // in general it is easiest if empty space
93 // is completely empty and solid space is
94 // completely
95 // solid. The exception to this is the
96 // region near our surface, where a gentle
97 // transition helps
98 // obtain smooth shading. By scaling by a
99 // large number and then clamping we achieve
100 // this effect
101 // of making most voxels fully solid or
102 // fully empty except near the surface..
103 simplexNoiseValue *= 3.5f;
104 simplexNoiseValue =
105     Mathf.Clamp(simplexNoiseValue, -0.5f,
106         0.5f);
107 // Go back to the range 0.0 to 1.0;
108 simplexNoiseValue += 0.5f;
109 // And then to 0 to 255, ready to convert
110 // into a byte.
111 simplexNoiseValue *= 255;
112 // Write the final value value into the
113 // third material channel (the one with the
114 // rock texture).
115 // The value being written is usually 0
116 // (empty) or 255 (solid) except around the
117 // transition.
118 materialSet.weights[2] =
119     (byte)simplexNoiseValue;
120 // Lastly we write soil or grass voxels into
121 // the volume to create a level floor
122 // between the rocks.
123 // This means we want to set the sum of the
124 // materials to 255 if the voxel is below
125 // the floor height.
126 // We don't want to interfere with the rocks
127 // on the transition between the material so
128 // we work out
129 // how much extra we have to add to get to
130 // 255 and then add that to either soil or
```

Bilag E. Kilde kode

```
109         grass.  
byte excess = (byte)(255 -  
110         materialSet.weights[2]);  
111     if(y < 11)  
112     {  
113         // Add to soil material channel.  
materialSet.weights[1] = excess;  
114     }  
115     else if(y < 12)  
116     {  
117         // Add to grass material channel.  
118         materialSet.weights[0] = excess;  
119     }  
120  
121     // We can now write our computed voxel value into the volume.  
122  
123     data.SetVoxel(x, y, z, materialSet);  
124 }  
125 }  
126 }  
127 }  
128 }
```

E.24 Restarter.cs

```
1 using UnityEngine;
2 using UnityEngine.SceneManagement;
3
4 public class Restarter : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11    // Update is called once per frame
12    void Update () {
13        if (Input.GetKeyDown("r" )
14            {
15
16            SceneManager.LoadScene(Application.loadedLevelName);
17                //Application.LoadLevel (Application.loadedLevelName);
18
19            }
20    }
21 }
```

E.25 ScaryObject.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class ScaryObject : MonoBehaviour {
5     /*At this point this script does not have any actual functionality by itself, but
6         acts as a component that will make other entities react when they are in range.*/
7 }
```


E.26 SelectSite.cs

```

1  using UnityEngine;
2
3  public class SelectSite : MonoBehaviour {
4
5      public Camera cam;
6      public Check1 touch;
7      RaycastHit hit;
8
9      public bool hitting = false;
10
11
12     // Use this for initialization
13     void Start () {
14
15     }
16
17     // Update is called once per frame
18     void Update () {
19         //print("Touchcount: "+Input.touchCount);
20         if (/*Input.GetButton("Fire1") || */touch.fingers == 1)
21         {
22             Ray ray = cam.ScreenPointToRay(Input.mousePosition);
23             if (Physics.Raycast(ray.origin, ray.direction, out hit))
24             {
25                 hitting = false;
26                 if (hit.transform.name.Contains("pinPrefab"))
27                 {
28                     hitting = true;
29                     hit.transform.GetComponent<PinSelected>().isSelected(true);
30
31                 }
32                 if (hit.transform.name.Contains("Radar"))
33                 {
34                     hitting = true;
35
36                     hit.transform.parent.parent.parent.transform.GetComponent<PinSelected>().selectTime
37                         = .3f;
38                 }
39             }
40         }
41     }
42 }

```

E.27 SimplexNoise.cs

```
1 // SimplexNoise for C#
2 // Author: Heikki Tormala
3
4 //This is free and unencumbered software released into the public domain.
5
6 //Anyone is free to copy, modify, publish, use, compile, sell, or
7 //distribute this software, either in source code form or as a compiled
8 //binary, for any purpose, commercial or non-commercial, and by any
9 //means.
10
11 //In jurisdictions that recognize copyright laws, the author or authors
12 //of this software dedicate any and all copyright interest in the
13 //software to the public domain. We make this dedication for the benefit
14 //of the public at large and to the detriment of our heirs and
15 //successors. We intend this dedication to be an overt act of
16 //relinquishment in perpetuity of all present and future rights to this
17 //software under copyright law.
18
19 //THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
20 //EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
21 //MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
22 //IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR
23 //OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
24 //ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
25 //OTHER DEALINGS IN THE SOFTWARE.
26
27 //For more information, please refer to <http://unlicense.org/>
28
29
30 namespace SimplexNoise
31 {
32     /// <summary>
33     /// Implementation of the Perlin simplex noise, an improved Perlin noise
34     /// algorithm.
35     /// Based loosely on SimplexNoise1234 by Stefan Gustavson
36     /// <http://staffwww.itn.liu.se/~stegu/aqsis/aqsis-newnoise/>
37     /// </summary>
38     public class Noise
39     {
40         /// <summary>
41         /// 1D simplex noise
42         /// </summary>
43         /// <param name="x"></param>
44         /// <returns></returns>
45         public static float Generate(float x)
46         {
47             int i0 = FastFloor(x);
48             int i1 = i0 + 1;
49             float x0 = x - i0;
50             float x1 = x0 - 1.0f;
```

E.27. SimplexNoise.cs

```

51         float n0, n1;
52
53         float t0 = 1.0f - x0*x0;
54         t0 *= t0;
55         n0 = t0 * t0 * grad(perm[i0 & 0xff], x0);
56
57         float t1 = 1.0f - x1*x1;
58         t1 *= t1;
59         n1 = t1 * t1 * grad(perm[i1 & 0xff], x1);
60         // The maximum value of this noise is 8*(3/4)^4 = 2.53125
61         // A factor of 0.395 scales to fit exactly within [-1,1]
62         return 0.395f * (n0 + n1);
63     }
64
65     /// <summary>
66     /// 2D simplex noise
67     /// </summary>
68     /// <param name="x"></param>
69     /// <param name="y"></param>
70     /// <returns></returns>
71     public static float Generate(float x, float y)
72     {
73         const float F2 = 0.366025403f; // F2 = 0.5*(sqrt(3.0)-1.0)
74         const float G2 = 0.211324865f; // G2 = (3.0-Math.sqrt(3.0))/6.0
75
76         float n0, n1, n2; // Noise contributions from the three corners
77
78         // Skew the input space to determine which simplex cell we're in
79         float s = (x+y)*F2; // Hairy factor for 2D
80         float xs = x + s;
81         float ys = y + s;
82         int i = FastFloor(xs);
83         int j = FastFloor(ys);
84
85         float t = (float)(i+j)*G2;
86         float X0 = i-t; // Unskew the cell origin back to (x,y) space
87         float Y0 = j-t;
88         float x0 = x-X0; // The x,y distances from the cell origin
89         float y0 = y-Y0;
90
91         // For the 2D case, the simplex shape is an equilateral triangle.
92         // Determine which simplex we are in.
93         int i1, j1; // Offsets for second (middle) corner of simplex in (i,j)
94             coords
95         if(x0>y0) {i1=1; j1=0;} // lower triangle, XY order: (0,0)->(1,0)->(1,1)
96         else {i1=0; j1=1;} // upper triangle, YX order: (0,0)->(0,1)->(1,1)
97
98         // A step of (1,0) in (i,j) means a step of (1-c,-c) in (x,y), and
99         // a step of (0,1) in (i,j) means a step of (-c,1-c) in (x,y), where
100         // c = (3-sqrt(3))/6
101
102         float x1 = x0 - i1 + G2; // Offsets for middle corner in (x,y) unskewed
103             coords
104         float y1 = y0 - j1 + G2;
105         float x2 = x0 - 1.0f + 2.0f * G2; // Offsets for last corner in (x,y)
106             coords

```

Bilag E. Kilde kode

```
104         float y2 = y0 - 1.0f + 2.0f * G2;
105
106         // Wrap the integer indices at 256, to avoid indexing perm[] out of
107         // bounds
108         int ii = i % 256;
109         int jj = j % 256;
110
111         // Calculate the contribution from the three corners
112         float t0 = 0.5f - x0*x0-y0*y0;
113         if(t0 < 0.0f) n0 = 0.0f;
114         else {
115             t0 *= t0;
116             n0 = t0 * t0 * grad(perm[ii+perm[jj]], x0, y0);
117         }
118
119         float t1 = 0.5f - x1*x1-y1*y1;
120         if(t1 < 0.0f) n1 = 0.0f;
121         else {
122             t1 *= t1;
123             n1 = t1 * t1 * grad(perm[ii+i1+perm[jj+j1]], x1, y1);
124         }
125
126         float t2 = 0.5f - x2*x2-y2*y2;
127         if(t2 < 0.0f) n2 = 0.0f;
128         else {
129             t2 *= t2;
130             n2 = t2 * t2 * grad(perm[ii+1+perm[jj+1]], x2, y2);
131         }
132
133         // Add contributions from each corner to get the final noise value.
134         // The result is scaled to return values in the interval [-1,1].
135         return 40.0f * (n0 + n1 + n2); // TODO: The scale factor is preliminary!
136     }
137
138     public static float Generate(float x, float y, float z)
139     {
140         // Simple skewing factors for the 3D case
141         const float F3 = 0.3333333333f;
142         const float G3 = 0.1666666667f;
143
144         float n0, n1, n2, n3; // Noise contributions from the four corners
145
146         // Skew the input space to determine which simplex cell we're in
147         float s = (x+y+z)*F3; // Very nice and simple skew factor for 3D
148         float xs = x+s;
149         float ys = y+s;
150         float zs = z+s;
151         int i = FastFloor(xs);
152         int j = FastFloor(ys);
153         int k = FastFloor(zs);
154
155         float t = (float)(i+j+k)*G3;
156         float X0 = i-t; // Unskew the cell origin back to (x,y,z) space
157         float Y0 = j-t;
158         float Z0 = k-t;
```

E.27. SimplexNoise.cs

```

159     float x0 = x-X0; // The x,y,z distances from the cell origin
160     float y0 = y-Y0;
161     float z0 = z-Z0;
162
163     // For the 3D case, the simplex shape is a slightly irregular
164     // tetrahedron.
165     // Determine which simplex we are in.
166     int i1, j1, k1; // Offsets for second corner of simplex in (i,j,k) coords
167     int i2, j2, k2; // Offsets for third corner of simplex in (i,j,k) coords
168
169     /* This code would benefit from a backport from the GLSL version! */
170     if(x0>=y0) {
171         if(y0>=z0)
172             { i1=1; j1=0; k1=0; i2=1; j2=1; k2=0; } // X Y Z order
173         else if(x0>=z0) { i1=1; j1=0; k1=0; i2=1; j2=0; k2=1; } // X Z Y
174             order
175         else { i1=0; j1=0; k1=1; i2=1; j2=0; k2=1; } // Z X Y order
176     }
177     else { // x0<y0
178         if(y0<z0) { i1=0; j1=0; k1=1; i2=0; j2=1; k2=1; } // Z Y X order
179         else if(x0<z0) { i1=0; j1=1; k1=0; i2=0; j2=1; k2=1; } // Y Z X order
180         else { i1=0; j1=1; k1=0; i2=1; j2=1; k2=0; } // Y X Z order
181     }
182
183     // A step of (1,0,0) in (i,j,k) means a step of (1-c,-c,-c) in (x,y,z),
184     // a step of (0,1,0) in (i,j,k) means a step of (-c,1-c,-c) in (x,y,z),
185     // and
186     // a step of (0,0,1) in (i,j,k) means a step of (-c,-c,1-c) in (x,y,z),
187     // where
188     // c = 1/6.
189
190     float x1 = x0 - i1 + G3; // Offsets for second corner in (x,y,z) coords
191     float y1 = y0 - j1 + G3;
192     float z1 = z0 - k1 + G3;
193     float x2 = x0 - i2 + 2.0f*G3; // Offsets for third corner in (x,y,z)
194     // coords
195     float y2 = y0 - j2 + 2.0f*G3;
196     float z2 = z0 - k2 + 2.0f*G3;
197     float x3 = x0 - 1.0f + 3.0f*G3; // Offsets for last corner in (x,y,z)
198     // coords
199     float y3 = y0 - 1.0f + 3.0f*G3;
200     float z3 = z0 - 1.0f + 3.0f*G3;
201
202     // Wrap the integer indices at 256, to avoid indexing perm[] out of
203     // bounds
204     int ii = Mod(i, 256);
205     int jj = Mod(j, 256);
206     int kk = Mod(k, 256);
207
208     // Calculate the contribution from the four corners
209     float t0 = 0.6f - x0*x0 - y0*y0 - z0*z0;
210     if(t0 < 0.0f) n0 = 0.0f;
211     else {
212         t0 *= t0;
213         n0 = t0 * t0 * grad(perm[ii+perm[jj+perm[kk]]], x0, y0, z0);
214     }

```

Bilag E. Kilde kode

```
208
209     float t1 = 0.6f - x1*x1 - y1*y1 - z1*z1;
210     if(t1 < 0.0f) n1 = 0.0f;
211     else {
212         t1 *= t1;
213         n1 = t1 * t1 * grad(perm[ii+i1+perm[jj+j1+perm[kk+k1]]], x1, y1, z1);
214     }
215
216     float t2 = 0.6f - x2*x2 - y2*y2 - z2*z2;
217     if(t2 < 0.0f) n2 = 0.0f;
218     else {
219         t2 *= t2;
220         n2 = t2 * t2 * grad(perm[ii+i2+perm[jj+j2+perm[kk+k2]]], x2, y2, z2);
221     }
222
223     float t3 = 0.6f - x3*x3 - y3*y3 - z3*z3;
224     if(t3<0.0f) n3 = 0.0f;
225     else {
226         t3 *= t3;
227         n3 = t3 * t3 * grad(perm[ii+i+perm[jj+j+perm[kk+1]]], x3, y3, z3);
228     }
229
230     // Add contributions from each corner to get the final noise value.
231     // The result is scaled to stay just inside [-1,1]
232     return 32.0f * (n0 + n1 + n2 + n3); // TODO: The scale factor is
        preliminary!
233 }
234
235 public static byte[] perm = new byte[512] { 151,160,137,91,90,15,
236     131,13,201,95,96,53,194,233,7,225,140,36,103,30,69,142,8,99,37,240,21,10,23,
237     190,
238     6,148,247,120,234,75,0,26,197,62,94,252,219,203,117,35,11,32,57,177,33,
239     88,237,149,56,87,174,20,125,136,171,168,
240     68,175,74,165,71,134,139,48,27,166,
241     77,146,158,231,83,111,229,122,60,211,133,230,220,105,92,41,55,46,245,40,244,
242     102,143,54, 65,25,63,161, 1,216,80,73,209,76,132,187,208,
243     89,18,169,200,196,
244     135,130,116,188,159,86,164,100,109,198,173,186,
245     3,64,52,217,226,250,124,123,
246     5,202,38,147,118,126,255,82,85,212,207,206,59,227,47,16,58,17,182,189,28,42,
247     223,183,170,213,119,248,152, 2,44,154,163, 70,221,153,101,155,167,
248     43,172,9,
249     129,22,39,253, 19,98,108,110,79,113,224,232,178,185,
250     112,104,218,246,97,228,
251     251,34,242,193,238,210,144,12,191,179,162,241,
252     81,51,145,235,249,14,239,107,
253     49,192,214, 31,181,199,106,157,184, 84,204,176,115,121,50,45,127,
254     4,150,254,
255     138,236,205,93,222,114,67,29,24,72,243,141,128,195,78,66,215,61,156,180,
256     151,160,137,91,90,15,
257     131,13,201,95,96,53,194,233,7,225,140,36,103,30,69,142,8,99,37,240,21,10,23,
258     190,
259     6,148,247,120,234,75,0,26,197,62,94,252,219,203,117,35,11,32,57,177,33,
260     88,237,149,56,87,174,20,125,136,171,168,
261     68,175,74,165,71,134,139,48,27,166,
262     77,146,158,231,83,111,229,122,60,211,133,230,220,105,92,41,55,46,245,40,244,
```

E.27. SimplexNoise.cs

```

253         102,143,54, 65,25,63,161, 1,216,80,73,209,76,132,187,208,
           89,18,169,200,196,
254         135,130,116,188,159,86,164,100,109,198,173,186,
           3,64,52,217,226,250,124,123,
255         5,202,38,147,118,126,255,82,85,212,207,206,59,227,47,16,58,17,182,189,28,42,
256         223,183,170,213,119,248,152, 2,44,154,163, 70,221,153,101,155,167,
           43,172,9,
257         129,22,39,253, 19,98,108,110,79,113,224,232,178,185,
           112,104,218,246,97,228,
258         251,34,242,193,238,210,144,12,191,179,162,241,
           81,51,145,235,249,14,239,107,
259         49,192,214, 31,181,199,106,157,184, 84,204,176,115,121,50,45,127,
           4,150,254,
260         138,236,205,93,222,114,67,29,24,72,243,141,128,195,78,66,215,61,156,180
261     };
262
263     private static int FastFloor(float x)
264     {
265         return (x > 0) ? ((int)x) : (((int)x) - 1);
266     }
267
268     private static int Mod(int x, int m)
269     {
270         int a = x % m;
271         return a < 0 ? a + m : a;
272     }
273
274     private static float grad( int hash, float x )
275     {
276         int h = hash & 15;
277         float grad = 1.0f + (h & 7); // Gradient value 1.0, 2.0, ..., 8.0
278         if ((h & 8) != 0) grad = -grad; // Set a random sign for the
           gradient
279         return ( grad * x ); // Multiply the gradient with the distance
280     }
281
282     private static float grad( int hash, float x, float y )
283     {
284         int h = hash & 7; // Convert low 3 bits of hash code
285         float u = h<4 ? x : y; // into 8 simple gradient directions,
286         float v = h<4 ? y : x; // and compute the dot product with (x,y).
287         return ((h&1) != 0 ? -u : u) + ((h&2) != 0 ? -2.0f*v : 2.0f*v);
288     }
289
290     private static float grad( int hash, float x, float y , float z ) {
291         int h = hash & 15; // Convert low 4 bits of hash code into 12 simple
292         float u = h<8 ? x : y; // gradient directions, and compute dot product.
293         float v = h<4 ? y : h==12||h==14 ? x : z; // Fix repeats at h = 12 to 15
294         return ((h&1) != 0 ? -u : u) + ((h&2) != 0 ? -v : v);
295     }
296
297     private static float grad( int hash, float x, float y, float z, float t ) {
298         int h = hash & 31; // Convert low 5 bits of hash code into 32 simple
299         float u = h<24 ? x : y; // gradient directions, and compute dot product.
300         float v = h<16 ? y : z;
301         float w = h<8 ? z : t;

```

Bilag E. Kilde kode

```
302         return ((h&1) != 0 ? -u : u) + ((h&2) != 0 ? -v : v) + ((h&4) != 0 ? -w
303             : w);
304     }
305 }
```


E.28 StartProcessingExe.cs

```
1 using UnityEngine;
2 using System.Diagnostics;
3
4
5 public class StartProcessingExe : MonoBehaviour {
6     public bool finishedBuild;
7     private bool alreadyStarted = false;
8
9     void Start () {
10        if (finishedBuild && !alreadyStarted)
11        {
12            Process p = new Process();
13            p.StartInfo.FileName = "SVSProcessingUnity2.exe";
14            if (p.Start())
15                alreadyStarted = true;
16            else
17                print("Could not start processing app");
18        }
19    }
20 }
```

E.29 camZoom.cs

```
1 using UnityEngine;
2 using UnityEngine.EventSystems;
3
4 //using System.Windows;
5 //using System.Windows.Input;
6
7 //using System.Windows.Media;
8 //using System.Windows.Shapes;
9
10 public class camZoom : MonoBehaviour
11 {
12     public float rotSpeed = 1;
13     public float zoomSpeed = 4;
14     public float maxZoom = 4.5f;
15     int fingers;
16     public Check1 c;
17     public SelectSite ss;
18
19     float minZoom;
20     float step = 1;
21     Camera cam;
22
23     void Start()
24     {
25         cam = GetComponent<Camera>();
26         minZoom = cam.fieldOfView;
27
28     }
29
30     void Update()
31     {
32         fingers = c.fingers;
33         RaycastHit hit;
34         if (/*Input.GetButton("Fire1") || Input.GetButton("Fire2") ||
35             Input.GetButton("Fire3") || */fingers != 0)
36         {
37             step = map(cam.fieldOfView, maxZoom, minZoom, 0.1f, 1.5f);
38             Ray ray = cam.ScreenPointToRay(Input.mousePosition);
39             if (Physics.Raycast(ray.origin, ray.direction, out hit, 100))
40             {
41                 if (!ss.hitting) {
42                     Quaternion targetRot = Quaternion.LookRotation(hit.point -
43                         transform.position);
44                     transform.rotation = Quaternion.Slerp(transform.rotation,
45                         targetRot, rotSpeed * Time.deltaTime);//Rotate towards click
46                         point
47                 }
48                 if (cam.fieldOfView > maxZoom && fingers == 2/* ||
49                     Input.GetButton("Fire2")*/)//Zoom in
50                     cam.fieldOfView = Mathf.Lerp(cam.fieldOfView, cam.fieldOfView -
51                         step, zoomSpeed * Time.deltaTime);
52                 if (cam.fieldOfView < minZoom && fingers == 3 /*||
```

E.29. camZoom.cs

```
47         Input.GetButton("Fire3")*//Zoom out
         cam.fieldOfView = Mathf.Lerp(cam.fieldOfView, cam.fieldOfView +
         step, zoomSpeed * Time.deltaTime);
48     }
49 }
50 }
51
52 float map(float value, float istart, float istop, float ostart, float ostop)
53 {
54     return ostart + (ostop - ostart) * ((value - istart) / (istop - istart));
55 }
56 }
```

E.30 rotateObject.cs

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class rotateObject : MonoBehaviour {
5     [Tooltip("Rotation speed")]
6     public float speed;
7     [Tooltip("How many frames should pass before a new target destination is
8         found.")]
9     public int destFrequency;
10    int count = 0;
11    Quaternion targetRot;
12    // Use this for initialization
13    void Start () {
14        targetRot = Random.rotation;
15    }
16
17    // Update is called once per frame
18    void Update () {
19
20        count++;
21        if (count > destFrequency) {
22            targetRot = Random.rotation;
23            count = 0;
24        }
25
26        transform.rotation = Quaternion.Slerp(transform.rotation, targetRot, speed *
27            Time.deltaTime);
28    }
29 }
```

E.31 trackingCast.cs

```

1  using UnityEngine;
2  using System.Collections;
3  using Cubiquity;
4
5  public class trackingCast : MonoBehaviour {
6      RaycastHit hitInf;
7      GameObject terrain;
8      ClickToCarveTerrainVolume carve;
9      middleMan mm;
10     private TerrainVolume terrainVolume;
11     int finalY;
12
13     int range;
14     public float differenceLimit = 2f;
15     public float angle = .5f;
16     public float length = 5;
17     public float wsDisplace = 4;
18     public int finalYDisplace = 2;
19     Vector3 p;
20
21     // Use this for initialization
22     void Start () {
23         terrain = GameObject.FindGameObjectWithTag("ProTerrain");
24         mm =
25             GameObject.FindGameObjectWithTag("Global").GetComponent<middleMan>();
26         carve = terrain.GetComponent<ClickToCarveTerrainVolume>();
27         terrainVolume = terrain.GetComponent<TerrainVolume>();
28         range = carve.range;
29     }
30
31     // Update is called once per frame
32     void Update () {
33         //print (mm.GetNData ());
34         if (!mm.GetNData ())
35             return;
36         mm.SetNData (false);
37         //Vector2 mousePos = Input.mousePosition;
38         Ray ray = new Ray(transform.position, Vector3.down);
39         PickSurfaceResult pickResult;
40         bool hit = Picking.PickSurface(terrainVolume, ray, 1000.0f, out pickResult);
41
42         if (hit)
43         {
44             /*Check whether there is a too large difference in the distances of
45             rays, casted at an angle in all four sideways
46             directions and down. If there is, move the digging point up. This is
47             done to avoid violently digging into the
48             sides of shallow spots, or "craters"*/
49
50             Vector3 wsPos = pickResult.worldSpacePos + Vector3.up * wsDisplace;
51             ArrayList dists = new ArrayList();

```

Bilag E. Kilde kode

```
50 //Draw rays
51 Debug.DrawRay(wsPos, new Vector3(angle, -1, 0) * length, Color.green, 5,
    true); //west
52 Debug.DrawRay(wsPos, new Vector3(-angle, -1, 0) * length, Color.green,
    5, true); //east
53 Debug.DrawRay(wsPos, new Vector3(0, -1, angle) * length, Color.green, 5,
    true); //north
54 Debug.DrawRay(wsPos, new Vector3(0, -1, -angle) * length, Color.green,
    5, true); //south
55 Debug.DrawRay(wsPos, new Vector3(0, -1, 0) * length, Color.green, 5,
    true); //down
56
57 //find distances and add to an array
58 if (Physics.Raycast(wsPos, new Vector3(angle, -1, 0), out hitInf, length))
59     dists.Add(hitInf.distance);
60 if (Physics.Raycast(wsPos, new Vector3(-angle, -1, 0), out hitInf))
61     dists.Add(hitInf.distance);
62 if (Physics.Raycast(wsPos, new Vector3(0, -1, angle), out hitInf, length))
63     dists.Add(hitInf.distance);
64 if (Physics.Raycast(wsPos, new Vector3(0, -1, -angle), out
    hitInf, length))
65     dists.Add(hitInf.distance);
66 if (Physics.Raycast(wsPos, new Vector3(0, -1, 0), out hitInf, length))
67     dists.Add(hitInf.distance);
68
69 /*print it
70 string printer = "";
71
72 foreach (float d in dists) {
73     printer += d + " ";
74 }
75 print(printer); */
76
77 //check for large differences
78 dists.Sort();
79 finalY = (int)pickResult.volumeSpacePos.y;
80
81 if (dists.Count > 0){
82     float biggestD = (float)dists[dists.Count - 1] - (float)dists[0];
83     if (biggestD >= differenceLimit){
84         print("Difference: " + biggestD);
85
86         //finalY += (int)biggestD*finalYDisplace; //this is supposed to
            be used instead of the if in the try catch.
87     }
88
89 //Call the vacuum script
90 try
91 {
92     if ((float)dists[dists.Count - 1] - (float)dists[0] <
        differenceLimit) //not very pretty!
93         carve.DestroyVoxels((int)pickResult.volumeSpacePos.x,
            finalY, (int)pickResult.volumeSpacePos.z, range);
94 }
95 catch (CubiquityException e)
96 {
```

E.31. trackingCast.cs

```
97         print("Cubiquity exception" + e);
98         //This is cast if we try to call destroyVoxels on a y value that
           is out of the voxelvolume
99     }
100 }
101
102
103 }
104 /* if (Physics.Raycast(transform.position, Vector3.down, out hit)) {
105     print("Dig here: " + hit.point);
106     p = hit.point;
107     Debug.DrawRay(transform.position, Vector3.down, Color.green, 3, true);
108     carve.DestroyVoxels((int)p.x, (int)p.y, (int)p.z,range);
109 }*/
110 }
111
112 void SetRange(int r) {
113     range = r;
114 }
115 }
```

E.32 OSCListener.js

```
1 private var UDPHost : String = "127.0.0.1";
2 private var listenerPort : int = 8000;
3 private var broadcastPort : int = 57131;
4 private var oscHandler : Osc;
5 private var mm : middleMan;
6
7
8 private
9 private var eventName : String = "";
10 private var eventData : String = "";
11 private var posX : int = 0;
12 private var posZ : int = 0;
13 private var area : int = 0;
14 private var messages;
15 private var temp;
16 private var whirl;
17 //public var output_txt : UnityEngine.UI.Text;
18
19 public function Start ()
20 {
21     messages = new Array();
22     temp = new Array();
23
24     var udp : UDPPacketIO = GetComponent("UDPPacketIO");
25     mm = GameObject.FindGameObjectWithTag("Global").GetComponent("middleMan");
26     udp.init(UDPHost, broadcastPort, listenerPort);
27     oscHandler = GetComponent("Osc");
28     oscHandler.init(udp);
29
30     //oscHandler.SetAddressHandler("/eventTest", updateText);
31     oscHandler.SetAddressHandler("/positionData", positionData);
32     whirl = GameObject.Find("Whirl");
33 }
34 Debug.Log("Running");
35
36 function Update () {
37     /*
38     var cube = GameObject.Find("trackingTarget");
39     var x:int = posX;
40     var z:int = posZ;
41     var str:int = area;
42     //cube.transform.localScale = Vector3(boxWidth,5,boxHeight);
43     cube.transform.position = new Vector3(x,35,z);
44     cube.GetComponent("trackingCast").SetRange(str);
45     */
46     if(messages.length > 0)
47     {
48         temp = messages.pop();
49         var cube = GameObject.Find("trackingTarget");
50
51         var x:int = temp.mPosX;
52         var z:int = temp.mPosZ;
```


E.32. OSCListener.js

```
53     var str:int = temp.mArea;
54     var id:int = temp.mId;
55
56     //cube.transform.localScale = Vector3(boxWidth,5,boxHeight);
57     cube.transform.position = new Vector3(x,-3,z);
58     whirl.transform.position = cube.transform.position;
59
60     //print("ID: "+id);
61     cube.GetComponent("trackingCast").SetRange(str);
62
63 }
64 else
65 {
66     if(Random.Range(0,100) == 1)
67     {
68         whirl.transform.position = new Vector3(-100,0,-100);
69     }
70 }
71 }
72 public function positionData(oscMessage : OscMessage) : void
73 {
74     mm.SetNData(true);
75     //print("sending true");
76     Osc.OscMessageToString(oscMessage);
77     posX = oscMessage.Values[0];
78     posZ = oscMessage.Values[1];
79     area = oscMessage.Values[2];
80     id = oscMessage.Values[3];
81
82     var mes = new m(posX, posZ, area, id);
83     messages.push(mes);
84     //print(posX + " " + posZ);
85 }
86
87 class m{
88     var mPosX : int;
89     var mPosZ : int;
90     var mArea : int;
91     var mId : int;
92
93     function m(mPosX : int, mPosZ : int, mArea : int, mId : int){
94         this.mPosX = mPosX;
95         this.mPosZ = mPosZ;
96         this.mId = mId;
97         this.mArea = mArea;
98     }
99 }
100
101 /*public function updateText(oscMessage : OscMessage) : void
102 {
103     eventName = Osc.OscMessageToString(oscMessage);
104     eventData = oscMessage.Values[0];
105 } */
```

E.33 Blob.pde

```
1  /**
2   * Blob Class
3   *
4   * Based on this example by Daniel Shiffman:
5   * http://shiffman.net/2011/04/26/opencv-matching-faces-over-time/
6   *
7   * @author: Jordi Tost (@jorditost)
8   *
9   * University of Applied Sciences Potsdam, 2014
10  */
11
12  class Blob
13  {
14      private PApplet parent;
15
16      // Contour
17      public Contour contour;
18
19      // Am I available to be matched?
20      public boolean available;
21
22      // Should I be deleted?
23      public boolean delete;
24
25      // How long should I live if I have disappeared?
26      private int initTimer = 20; //127;
27      public int timer;
28
29      // Unique ID for each blob
30      int id;
31
32      // Make me
33      Blob(PApplet parent, int id, Contour c)
34      {
35          this.parent = parent;
36          this.id = id;
37          this.contour = new Contour(parent, c.pointMat);
38
39          available = true;
40          delete = false;
41
42          timer = initTimer;
43      }
44
45      // Show me
46      void display()
47      {
48          Rectangle r = contour.getBoundingBox();
49
50          float opacity = map(timer, 0, initTimer, 0, 127);
51          fill(0, 0, 255, opacity);
52          stroke(0, 0, 255);
```

E.33. Blob.pde

```
53     rect(r.x, r.y, r.width, r.height);
54     fill(255, 2*opacity);
55     textSize(26);
56     text(""+id, r.x+10, r.y+30);
57 }
58
59 // Give me a new contour for this blob (shape, points, location, size)
60 // Oooh, it would be nice to lerp here!
61 void update(Contour newC)
62 {
63     contour = new Contour(parent, newC.pointMat);
64     timer = initTimer;
65 }
66
67 // Count me down, I am gone
68 void countDown()
69 {
70     timer--;
71 }
72
73 // I am deed, delete me
74 boolean dead()
75 {
76     if (timer < 0)
77         return true;
78     return false;
79 }
80
81 public Rectangle getBoundingBox()
82 {
83     return contour.getBoundingBox();
84 }
85 }
```

E.34 IRTracking.pde

```
1  import oscP5.*;
2  import netP5.*;
3  import processing.video.*;
4  import gab.opencv.*;
5  import java.awt.Rectangle;
6  import java.awt.geom.AffineTransform;
7  import java.awt.image.AffineTransformOp;
8  import java.awt.image.BufferedImage;
9
10 //Osc
11 Capture cam1, cam2, cam3, cam4;
12 OscP5 oscP5;
13 NetAddress myBroadcastLocation;
14
15 //OpenCV
16 OpenCV opencv;
17
18 ArrayList<Contour> contours;
19 ArrayList<Contour> newBlobContours; // List of detected contours parsed as blobs
    (every frame)
20 ArrayList<Blob> blobList;
21
22 int blobSizeThreshold = 10;
23 int threshold = 175;
24 int blurSize = 4;
25 int blobCount = 0;
26
27 void setup()
28 {
29     size(1728, 960);
30     blobList = new ArrayList<Blob>();
31     //frameRate(60);
32     oscP5 = new OscP5(this, 12000);
33     myBroadcastLocation = new NetAddress("127.0.0.1", 8000);
34
35     String[] cameras = Capture.list();
36
37     if (cameras.length == 0)
38     {
39         println("There are no cameras available for capture.");
40         exit();
41     } else
42     {
43         println("Available cameras:");
44         printArray(cameras);
45
46         cam1 = new Capture(this, 864, 480, "Cam1", 30);
47         cam1.start();
48
49         cam2 = new Capture(this, 864, 480, "Cam2", 30);
50         cam2.start();
51
```

E.34. IRTracking.pde

```
52     cam3 = new Capture(this, 864, 480, "Cam3", 30);
53     cam3.start();
54
55     cam4 = new Capture(this, 864, 480, "Cam4", 30);
56     cam4.start();
57 }
58 opencv = new OpenCV(this, 1728, 960);
59 noStroke();
60 smooth();
61 }
62
63 void draw()
64 {
65     if (cam1.available() && cam2.available() && cam3.available() && cam4.available())
66     {
67         cam1.read();
68         cam2.read();
69         cam3.read();
70         cam4.read();
71
72         PImage c1 = cam1;
73         PImage c2 = cam2;
74         PImage c3 = cam3;
75         PImage c4 = cam4;
76
77         PGraphics output = createGraphics(1728, 960, JAVA2D);
78
79         output.beginDraw();
80         output.image(c1, 864, 0, 864, 480);
81         output.image(c2, 0, 0, 864, 480);
82         output.image(c3, 864, 480, 864, 480);
83         output.image(c4, 0, 480, 864, 480);
84         output.endDraw();
85
86         image(output, 0, 0);
87
88         opencv.loadImage(output.get());
89
90         opencv.threshold(threshold);
91
92         detectBlobs();
93
94         displayBlobs();
95
96         for (int i = 0; i < blobList.size(); i++)
97         {
98             if (!blobList.get(i).dead())
99             {
100                 Rectangle r = blobList.get(i).getBoundingBox();
101                 sendMessage(r.x, r.y, r.width*r.height, blobList.get(i).id);
102             }
103         }
104         noFill();
105         strokeWeight(3);
106
107         for (Contour contour : contours)
```

Bilag E. Kilde kode

```
108     {
109         stroke(0, 255, 0);
110         stroke(255, 0, 0);
111         beginShape();
112         for (PVector point : contour.getPolygonApproximation().getPoints())
113         {
114             vertex(point.x, point.y);
115         }
116         endShape();
117     }
118
119     text("Framerate: " + int(frameRate), 10, 450);
120     line(width*.5, 0, width*.5, height);
121     line(0, height*.5, width, height*.5);
122 }
123 }
124
125 void sendMessage(int xx, int zz, int area, int id)
126 {
127     //float mappedX = map(xx, 1728, 0, -39 + (random(3)-1), 33 + (random(3)-1));
128
129     float mappedX = map(xx, 1728, 0, -39, 34);
130
131     //float mappedZ = map(zz, 0, 960, 20 + (random(3)-1), -27 + (random(3)-1));
132     float mappedZ = map(zz, 0, 960, 20, -27);
133
134     float mappedArea = map(area, 3000, 30000, 5, 8); //check med blobs
135
136     OscMessage myOscMessage = new OscMessage("/positionData");
137     myOscMessage.add(mappedX);
138     myOscMessage.add(mappedZ);
139     myOscMessage.add(mappedArea);
140     myOscMessage.add(id);
141     oscP5.send(myOscMessage, myBroadcastLocation);
142
143     println("FPS = " + int(frameRate) + " ID: " + id + " X = " + xx + " (" + mappedX
144             + ")" + " Y (Mapped) = " + zz + " (" + mappedZ + ")" + " Area = " +
145             mappedArea);
146 }
147
148 void displayContoursBoundingBoxes()
149 {
150     for (int i=0; i<contours.size(); i++)
151     {
152         Contour contour = contours.get(i);
153         Rectangle r = contour.getBoundingBox();
154
155         if (r.width < blobSizeThreshold || r.height < blobSizeThreshold)
156             continue;
157
158         stroke(255, 0, 0);
159         fill(255, 0, 0, 150);
160         strokeWeight(2);
161     }
```

E.34. IRTracking.pde

```
162     rect(r.x-r.width, r.y, r.width, r.height);
163   }
164 }
165
166 //////////////////////////////////////////////////
167 // Blob Detection
168 //////////////////////////////////////////////////
169
170 void detectBlobs()
171 {
172   // Contours detected in this frame
173   // Passing 'true' sorts them by descending area.
174   contours = opencv.findContours(true, true);
175
176   newBlobContours = getBlobsFromContours(contours);
177
178   // Check if the detected blobs already exist are new or some has disappeared.
179
180   // SCENARIO 1
181   // blobList is empty
182   if (blobList.isEmpty())
183   {
184     // Just make a Blob object for every face Rectangle
185     for (int i = 0; i < newBlobContours.size(); i++)
186     {
187       println("+++ New blob detected with ID: " + blobCount);
188       blobList.add(new Blob(this, blobCount, newBlobContours.get(i)));
189       blobCount++;
190     }
191
192     // SCENARIO 2
193     // We have fewer Blob objects than face Rectangles found from OpenCV in this
194     // frame
195   } else if (blobList.size() <= newBlobContours.size())
196   {
197     boolean[] used = new boolean[newBlobContours.size()];
198     // Match existing Blob objects with a Rectangle
199     for (Blob b : blobList)
200     {
201       // Find the new blob newBlobContours.get(index) that is closest to blob b
202       // set used[index] to true so that it can't be used twice
203       float record = 50000;
204       int index = -1;
205       for (int i = 0; i < newBlobContours.size(); i++)
206       {
207         float d = dist(newBlobContours.get(i).getBoundingBox().x,
208                       newBlobContours.get(i).getBoundingBox().y, b.getBoundingBox().x,
209                       b.getBoundingBox().y);
210         if (d < record && !used[i])
211         {
212           record = d;
213           index = i;
214         }
215       }
216       // Update Blob object location
217       used[index] = true;
218     }
219   }
220 }
```

Bilag E. Kilde kode

```
215     b.update(newBlobContours.get(index));
216 }
217 // Add any unused blobs
218 for (int i = 0; i < newBlobContours.size(); i++)
219 {
220     if (!used[i])
221     {
222         println("+++ New blob detected with ID: " + blobCount);
223         blobList.add(new Blob(this, blobCount, newBlobContours.get(i)));
224         //blobList.add(new Blob(blobCount, blobs[i].x, blobs[i].y, blobs[i].width,
225             blobs[i].height));
226         blobCount++;
227     }
228 }
229 // SCENARIO 3
230 // We have more Blob objects than blob Rectangles found from OpenCV in this frame
231 } else
232 {
233     // All Blob objects start out as available
234     for (Blob b : blobList)
235     {
236         b.available = true;
237     }
238     // Match Rectangle with a Blob object
239     for (int i = 0; i < newBlobContours.size(); i++)
240     {
241         // Find blob object closest to the newBlobContours.get(i) Contour
242         // set available to false
243         float record = 50000;
244         int index = -1;
245         for (int j = 0; j < blobList.size(); j++)
246         {
247             Blob b = blobList.get(j);
248             float d = dist(newBlobContours.get(i).getBoundingBox().x,
249                 newBlobContours.get(i).getBoundingBox().y, b.getBoundingBox().x,
250                 b.getBoundingBox().y);
249             if (d < record && b.available)
250             {
251                 record = d;
252                 index = j;
253             }
254         }
255         // Update Blob object location
256         Blob b = blobList.get(index);
257         b.available = false;
258         b.update(newBlobContours.get(i));
259     }
260     // Start to kill any left over Blob objects
261     for (Blob b : blobList)
262     {
263         if (b.available)
264         {
265             b.countDown();
266             if (b.dead())
267             {
```


E.34. IRTracking.pde

```
268         b.delete = true;
269     }
270 }
271 }
272 }
273 }
274
275 ArrayList<Contour> getBlobsFromContours(ArrayList<Contour> newContours)
276 {
277     ArrayList<Contour> newBlobs = new ArrayList<Contour>();
278     // Which of these contours are blobs?
279     for (int i=0; i<newContours.size(); i++)
280     {
281         Contour contour = newContours.get(i);
282         Rectangle r = contour.getBoundingBox();
283
284         if (r.width < blobSizeThreshold || r.height < blobSizeThreshold)
285         {
286             continue;
287         }
288         newBlobs.add(contour);
289     }
290     return newBlobs;
291 }
292
293 void displayBlobs()
294 {
295     for (Blob b : blobList)
296     {
297         strokeWeight(1);
298         b.display();
299     }
300 }
```

E.35 svsDebugger.pde

```
1 import oscP5.*;
2 import netP5.*;
3
4 OscP5 oscP5;
5 NetAddress myBroadcastLocation;
6 int x, z, sZ, sX;
7 int str = 7000;
8 int counter = 0;
9 void setup() {
10   size(300, 300);
11   background(0);
12   textSize(32);
13
14   oscP5 = new OscP5(this, 12000);
15   myBroadcastLocation = new NetAddress("127.0.0.1", 8000);
16   x = (int)random(0, 640);
17   z = (int)random(0, 420);
18   sZ = 1;
19   sX = -1;
20 }
21
22 void draw() {
23
24
25   if (keyPressed) {
26     if (key == 's') {
27       x+= sX;
28       z+= sZ;
29       //print((int)random(100)%17, "");
30       if ((random(100)%17) == 1) {
31         sZ = -sZ;
32         print(sZ);
33       }
34       if ((random(100)%17) == 1)
35         sX = -sX;
36       if (x > 640 )
37         x = (int)random(640);
38       if (z > 420)
39         z = (int)random(420);
40       sendMessage(x, z, str, 0);
41     }
42     //Corner checks -----
43     if (key == '1')
44       sendMessage(0, 0, str, 0);
45     if (key == '2')
46       sendMessage(640, 0, str, 0);
47     if (key == '3')
48       sendMessage(0, 420, str, 0);
49     if (key == '4')
50       sendMessage(640, 420, str, 0);
51
52     //IJKL Controls -----
```

E.35. svgsDebugger.pde

```
53     if (key == 'j') {
54         x--;
55         sendMessage(x, z, str, 0);
56     }
57     if (key == 'l') {
58         x++;
59         sendMessage(x, z, str, 0);
60     }
61     if (key == 'i') {
62         z--;
63         sendMessage(x, z, str, 0);
64     }
65     if (key == 'k') {
66         z++;
67         sendMessage(x, z, str, 0);
68     }
69
70     //Test Multiple
71     if (key == 'd'){
72         if(counter%2 == 1)
73             sendMessage(170,220,str,0);
74         else
75             sendMessage(470,220,str,0);
76         counter++;
77     }
78 }
79 if (x > 640 || x < 0 )
80     x = (int)random(640);
81 if (z > 420 || x < 0)
82     z = (int)random(420);
83
84
85 }
86
87 void sendMessage(int xx, int zz, int area, int num) {
88     /*float mappedX = map(xx, 640, 0, 0, 90);
89     float mappedZ = map(zz, 0, 480,60, 10);
90     */
91     float mappedX = map(xx, 640, 0, 10, 85);
92     float mappedZ = map(zz, 0, 480,50, 4);
93     float mappedArea = map(area, 500, 16000, 1, 10); //check med blobs
94     drawText(mappedX, mappedZ, mappedArea, xx, zz);
95
96     OscMessage myOscMessage = new OscMessage("/positionData");
97     myOscMessage.add(mappedX);
98     myOscMessage.add(mappedZ);
99     myOscMessage.add(mappedArea);
100    //myOscMessage.add(num);
101    oscP5.send(myOscMessage, myBroadcastLocation);
102 }
103
104 void drawText(float w1, float w2, float w3, float xx, float zz) {
105     background(0);
106     fill(0, 102, 153);
107     text("mX:"+w1, 10, 30);
108     text("X:"+xx, 10, 60);
```

Bilag E. Kilde kode

```
109     text("mY:"+w2, 10, 90);  
110     text("Y:"+zz, 10, 120);  
111     text("A:"+w3, 10, 290);  
112 }
```

E.36 testCams.pde

```
1 import processing.video.*;
2
3 Capture cam1, cam2, cam3, cam4;
4
5 void setup()
6 {
7     fullScreen();
8     String[] cameras = Capture.list();
9
10    if (cameras.length == 0)
11    {
12        println("There are no cameras available for capture.");
13        exit();
14    } else
15    {
16        println("Available cameras:");
17        printArray(cameras);
18
19        cam1 = new Capture(this, 864, 480, "Cam1", 30);
20        cam1.start();
21
22        cam2 = new Capture(this, 864, 480, "Cam2", 30);
23        cam2.start();
24
25        cam3 = new Capture(this, 864, 480, "Cam3", 30);
26        cam3.start();
27
28        cam4 = new Capture(this, 864, 480, "Cam4", 30);
29        cam4.start();
30    }
31 }
32 void draw()
33 {
34    if (cam1.available() && cam2.available() && cam3.available() && cam4.available())
35    {
36        cam1.read();
37        cam2.read();
38        cam3.read();
39        cam4.read();
40
41        PImage c1 = cam1;
42        PImage c2 = cam2;
43        PImage c3 = cam3;
44        PImage c4 = cam4;
45
46        PGraphics output = createGraphics(width, height, JAVA2D);
47
48        output.beginDraw();
49        output.image(c1, 0, 0, width*0.5, height*0.5);
50        output.image(c2, width*0.5, 0, width*0.5, height*0.5);
51        output.image(c3, 0, height*0.5, width*0.5, height*0.5);
52        output.image(c4, width*0.5, height*0.5, width*0.5, height*0.5);
```

Bilag E. Kilde kode

```
53     output.endDraw();
54
55     strokeWeight(2);
56
57     drawLines("Not really cam1", 0, 0.5, 0.25, 0, 0.5, 0.25);
58     drawLines("Not really cam2", 0.5, 1, 0.25, 0, 0.5, 0.75);
59     drawLines("Not really cam3", 0, 0.5, 0.75, 0.5, 1, 0.25);
60     drawLines("Not really cam4", 0.5, 1, 0.75, 0.5, 1, 0.75);
61
62     strokeWeight(5);
63     stroke(255, 0, 0);
64     line(width*0.5, 0, width*0.5, height);
65     line(0, height*0.5, width, height*0.5);
66 }
67 }
68
69 void drawLines(String name, float horiX1, float horiX2, float horiY, float vertiY1,
70               float vertiY2, float vertiX)
71 {
72     fill(255);
73     text(name, width*horiX1+10, height*vertiY1+30);
74
75     stroke(0, 150, 255);
76     line(width*horiX1, height*horiY, width*horiX2, height*horiY);
77
78     stroke(150, 255, 0);
79     line(width*vertiX, height*vertiY1, width*vertiX, height*vertiY2);
80 }
```