



Høgskulen  
på Vestlandet

# BACHELOROPPGAVE

## VR-simulator for trening til korrekt gjenopplivning av nyfødte

VR training simulator for the application of correct  
resuscitation techniques for newborns

Joakim Andås Johesan  
Grim Thomas Hammerstad  
Eirik Alvestad

Dataingeniør

Fakultet for ingeniør- og naturvitenskap  
Institutt for datateknologi, elektroteknologi og realfag  
Veileder: **Sven-Olai Høyland**

02.06-2020

*Jeg bekrefter at arbeidet er selvstendig utarbeidet, og at referanser/kildehenvisninger til alle kilder som er brukt i arbeidet er oppgitt, jf. Forskrift om studium og eksamen ved Høgskulen på Vestlandet, § 10.*

<i>Rapportens tittel:</i> VR-simulator for trening til korrekt gjenopplivning av nyfødte	<i>Dato:</i> 02.06 - 2020
<i>Forfatter(e):</i> Joakim Andås Johesan, Eirik Alvestad, Grim Thomas Hammerstad	<i>Antall sider u/vedlegg:</i> 40 <i>Antall sider vedlegg:</i> 19
<i>Studieretning:</i> Dataingeniør	<i>Antall disketter/CD-er:</i> 0
<i>Kontaktperson ved studieretning:</i> Sven-Olai Høyland	<i>Gradering:</i> Ingen
<i>Merknader:</i>	

<i>Oppdragsgiver:</i> Master i jordmorfag, HVL Bergen	<i>Oppdragsgivers referanse:</i>
<i>Oppdragsgivers kontaktperson:</i> Katrine Aasekjær, Ivar Rosenberg	<i>Telefon:</i> 93617784 (Ivar)

*Sammendrag / Summary:*

Jordmorutdanningen ved HVL ønsker å kunne tilby mer trening på gjenopplivning av nyfødte til sine studenter. De utfører i dag simuleringer med dukke. Med et VR-spill som tillater simulering av gjenopplivning og korrekte teknikker knyttet til dette, kan studentene øve på gjenopplivning uten instruktør, og uten behov for nøye planlegging. I prosjektet prøver gruppen å lage en prototype av VR spillet, med fokus på at studenten utfører gjenopplivning etter Norsk Resuscitasjonsråd sin algoritme. Simulatoren presenterer et nyfødt spedbarn som ikke har respirasjon, lav hjerterefrekvens, med akutt behov for livreddende hjerte- og lungeredning.

Prototypen som blir laget simulerer de fleste nødvendige parameterne som oppdragsgiver har behov for, og den oppleves brukervennlig i bruk. Simulatoren viser seg å være en god plattform å bygge videre på og kan med små forandringer brukes i undervisning ifølge oppdragsgiver. Det er hvor tett simulatoren følger NRR algoritmen som er behov for å justere, slik at den tvinger brukerne til å følge NRR algoritmen helt korrekt.

The midwifery subject at HVL wish to be able to offer more comprehensive training to their students in the task of resuscitation of newborn children. They wish to accomplish this with a VR game that simulates the resuscitation scenario to enable students to learn the correct application of the Norwegian Resuscitation Advisory board algorithm. The project task is to create a prototype simulator. The simulator presents a newborn without respiration, low heart rate, and an acute need for resuscitation.

The prototype offers the needed functionality and is evaluated to be user-friendly. The employer suggests the prototype could be used for instruction with small adjustments. The game needs adjustments to medical correctness and strict following of the NRR algorithm before use in class.

*Stikkord:*

VR, Simulator, Treningsverktøy, Unity, Jordmor, C#, Nyfødt, Gjenopplivning

# Forord

Prosjektet vi valgte og startet ble ikke som vi forventet på grunn av Covid-19 pandemien, men vi fikk likevel mye ut av det. Vi lærte oss å bruke Unity, å utvikle for VR plattformer, hvordan nyfødte blir håndtert, og prosessen som sørger for at de som trenger hjelp i starten på livet får det. Mest av alt har vi lært om prosessen i et prosjekt fullt av variabler, planlegging, kommunikasjon og samarbeid.

Gruppen ønsker å takke veileder Sven-Olai Høyland for god hjelp med prosjektet, rapporten, og tre gode år med en av de beste foreleserne på Datalinjen hos HVL. Vi ønsker å takke oppdragsgiver Katrine Aasekjær og Ivar Rosenberg for en interessant og god utfordring, og for hjelpen deres med oppgaven.

Takk til Kristoffer Erdal for hans gjennomgang og kommentering av oppgaven. Takk til Sandra Birkeland for det samme.

# Innholdsfortegnelse

<b>Forord</b>	<b>3</b>
<b>Innholdsfortegnelse</b>	<b>4</b>
<b>1 Innledning</b>	<b>7</b>
1.1 Motivasjon og mål	7
1.2 Kontekst	7
1.3 Avgrensninger	8
1.4 Ressurser	9
1.5 Oppbygging av rapporten	9
1.6 Etske problemstillinger knyttet til oppgaven	9
<b>2 Prosjektbeskrivelse</b>	<b>11</b>
2.1 Prosjekteier	11
2.2 Initielle krav	12
<b>3 Design av prosjektet</b>	<b>14</b>
3.1 Forslag til løsning	14
3.1.1 VR simulator basert på Unity, simulering av gjenopplivning	14
3.1.2 AR løsning ved hjelp av AR headset og dukke	15
3.1.3 App for trening på resuscitasjonsalgoritmen	15
3.1.4 Diskusjon av alternativene	15
3.2 Valg av verktøy	16
3.3 Prosjektmetodikk	16
3.3.1 Utviklingsmetodikk	16
3.3.2 Risikovurdering	17
3.4 Evalueringsplan	19
3.4.1 Unit-testing	19
3.4.2 Brukertesting	19
3.4.3 Akseptansetesting	19
3.4.4 Vitenskapelig test av effektivitet	20
3.4.5 Planlagt evalueringsmetode	20
<b>4 Detaljert design</b>	<b>21</b>
4.1 Funksjonalitet i simulatoren	21
4.1.1 Tidsinformasjon til bruker	21
4.1.2 Varmelampe	22
4.1.3 Pulsdisplay	22
4.1.4 Det nyfødte barnet	23

4.1.5 Ventilasjonsbag, Lærdalsbag	24
4.1.6 VR Interaksjoner	25
4.2 Modeller, teksturer, rigger og lyssetting	25
4.3 Oppbygging av prosjektet i Unity	27
4.4 Samhandlingen av kode i Unity	27
4.5 Oppbygging av scenarioet	28
4.5.1 Kodemessig oppbygging i prosjektet	28
4.5.2 Scenario	29
4.6 Bygge for Oculus Quest	30
<b>5 Evaluering</b>	<b>32</b>
5.1 Evalueringsmetode	32
5.2 Oppdragsgivers underveisevaluering	32
<b>6 Resultater</b>	<b>34</b>
<b>7 Diskusjon</b>	<b>36</b>
7.1 Utviklingsmetodikk, Github og Unity Collab	36
7.2 Begrensninger knyttet til lav ytelse på Oculus Quest	36
7.3 Utdfordringer knyttet til Covid-19 pandemien	37
7.4 Innkjøp av 3d-ressurser, sammenlignet med å lage de selv	38
7.5 Etikk i prototypen	38
7.6 Utdfordringer med Unity VR og Oculus VR pakke	39
<b>8 Konklusjon og videre arbeid</b>	<b>40</b>
8.1 Måloppnåelse og konklusjon	40
8.2 Utelatt funksjonalitet	40
8.3 Videre arbeid	41
<b>9 Referanser</b>	<b>42</b>
9.1 Kildeliste	42
9.2 Figurliste	43
<b>10 Vedlegg</b>	<b>44</b>
10.1 Risikoliste	44
10.2 Gantt diagram	45
10.3 Brukermanual	46
10.4 Ordliste	48
10.5 Scenario foreslått av Katrine Aasekjær	49
10.6 Evaluering av første prototype fra oppdragsgiver 20.05.2020	51
10.7 Evaluering fra oppdragsgiver av andre prototype den 27.05.2020	53
10.8 Original prosjektbeskrivelse før valg av prosjekt	55



# 1 Innledning

Se prosjektbloggen her: <https://joakimjohesan.no/blogg.html>

Se prosjektet sitt Github repository her: <https://github.com/182719/Bachelor-Oppgave>

Ordlister finnes i kapittel [10.4](#), der vil du finne definisjoner for uvanlige eller tekniske ord.

## 1.1 Motivasjon og mål

Jordmorstudenter ved Høgskolen på Vestlandet (HVL) trener i dag på gjenopplivning av nyfødte etter Norsk Resuscitasjonsråd (NRR) sin algoritme for gjenopplivning <sup>[1]</sup> (NRR, 2015). Algoritmen fungerer som et flytdiagram hvor man går fra node til node avhengig av parametere og respons hos den nyfødte, mot en slutt node. Hver node har visse parametere inn, og foreslått teknikk eller reaksjon til disse. Algoritmen setter veldig spesifikke tidsbegrensninger og faktorer som må tas hensyn til. Under trening med dukke opplever instruktører at studentene glemmer algoritmen grunnet den stressende trenings situasjonen. Dette skjer selv om studentene i normale situasjoner kan algoritmen utenat.

Se vedlegg [10.8](#) for en detaljert forklaring av NRR algoritmen.

Trening med dukke gjøres sjeldnere enn ønsket siden dette krever mye planlegging, forarbeid og tid. Det er også begrenset hvor mange studenter man kan ha med samtidig på disse simuleringene. Derfor ønsker jordmor-utdanningsprogrammet ved HVL en simulator som kan brukes hyppigere, som trygt og effektivt kan brukes til trening av studentene.

Målet med oppgaven er å lage en prototype VR simulator som minimum støtter ett scenario med gjenopplivning av en nyfødt. Med ett scenario menes det en simulering av en nyfødt som trenger gjenopplivning. Simulatoren kan ha ulike kliniske sensor og tilbakemeldings-forløp som presenteres for studenten. Studenten må reagere på eller løse disse på riktig måte i henhold til NRR algoritmen.

Å lage en simulator innebærer å lage en applikasjon i Unity spillmotoren med støtte for ett VR headset. Dette spillet vil minimum kunne brukes for å trene på ett scenario med gjenopplivning av en nyfødt. Hvis det blir tid, er det ønskelig å utvide simulatoren med flere scenarioer.

## 1.2 Kontekst

Under studieforløpet til en jordmorstudent er ofte den eneste muligheten for å trene på gjenopplivning av nyfødte i et klasserom med en instruktør og en dukke. Studentene får ofte lov til å være med på fødsler, men i en situasjon hvor gjenoppliving kreves vil studenten bli bedt om å gå ut på gangen. Dette gjør at opptreningen for gjenopplivning er noe begrenset.

Jordmorstudentene på HVL har tidligere fått prøve anatomiske spill med VR briller. Det viste seg at det tok kort tid for studentene å bli kjent med brillene, selv for de som aldri hadde prøvd VR før. Erfaringen herfra gjør at instituttet ønsker å utforske alternative treningsmetoder for gjenopplivning ved bruk av VR.

## 1.3 Avgrensninger

Oppgaven har et stort potensial som kan utvides i mange retninger, blant annet generell trening på gjenopplivning av mennesker i alle aldre. Det kunne blitt implementert flere scenarioer, slik at brukerne kan trene på flere ting. Derfor er det viktig å begrense prosjektets omfang med tanke på tiden vi har til rådighet. Prosjektet er i utgangspunktet avgrenset til å simulere ett scenario med gjenopplivning av nyfødte barn.

Det er flere avgrensninger når det gjelder teknologien vi bruker. Ett av scenarioene som kan være aktuelt er hjertekompresjon av den nyfødte. Hjertekompresjoner i et VR-miljø har flere utfordringer og restriksjoner. De viktigste elementene ved hjertekompresjoner av nyfødte er fingerplassering og hvor mye krefter man skal bruke. VR brillene vi planlegger å bruke, bruker håndkontrollere for å flytte hendene til studenten i spillet. Dette gir mulighet for å plassere hendene på riktig sted, men ikke hvor fingrene skal plasseres. Riktig fingerplassering er viktig i en reell situasjon og derfor er det ikke gunstig at simulatoren ikke kan brukes til å trene på dette.

I et VR-miljø er det umulig å simulere at brukeren fysisk tar på noe, der brukeren føler "kontakten". Det er derfor ikke mulig å lage simulatoren slik at en bruker får øve seg på hvor mye trykk som skal til for hver hjertekompresjon på den nyfødte. I teorien kunne man laget et augmented reality (AR) spill som hadde brukt en dukke i virkeligheten til å simulere barnet, overlagt med virtuell grafikk.

Oculus Quest er et VR headset med begrenset regnekapasitet både generelt og grafisk. Det er bygget på maskinvare for mobiltelefoner og kjører en modifisert versjon av Android operativsystemet. Dette vil begrense grafisk realisme og muligheter for å kjøre tung kode. Eksempler på tung kode kan være realistiske støyfunksjoner og generative algoritmer for å simulere tilfeldig bevegelser, lyder, og oppførsel. Et headset som kun tilbyr et audiovisuelt grensesnitt mot en kraftigere datamaskin vil kunne ha muligheter for høyere grafisk realisme, oppdateringsfrekvens og tung kode. Begrensninger her kan føre til at simuleringen ikke oppleves realistisk, og at den ikke klarer å formidle realistiske tegn i animasjon og oppførsel av nyfødte. Dette kan begrense simulatorens treningseffekt og korrelasjon mot virkeligheten.

Kun en student kan bruke VR simulatoren om gangen. Dette begrenser muligheten for samarbeid i simulatoren. Det er vanlig at flere jordmødre jobber samtidig med gjenopplivning. Det kan være mulig å implementere muligheten for at flere headset kan delta i samme scenario samtidig, men det vil trolig være utenfor hva det er mulig å få tid til i prosjektperioden nå. Et moment knyttet til dette er at en eventuell instruktør ikke kan delta for å hjelpe eleven. En måte vi kan korrigere for dette på er at instruktøren kan få se en videostrøm fra headsettet på en pc.



## 1.4 Ressurser

- Spillutviklingsplattformen Unity 2019.3.5<sup>[2]</sup> (Unity, 2020)
- 3d modellering og animasjonsverktøy
  - Autodesk 3ds Max 2018
  - Blender 2.80
- 2d grafikk verktøy: Adobe Photoshop
- NRR sin gjenopplivnings algoritme <sup>[1]</sup>(NRR, 2015)
- Oculus Quest VR briller <sup>[3]</sup> (Oculus, 2020)
- Personlige datamaskiner med høy ytelse i GPU og CPU
- Kontaktpersoner: Katrine Aasekjær (HVL) og Ivar Rosenberg (HVL)
- Veileder Sven-Olai Høyland (HVL)

## 1.5 Oppbygging av rapporten

Rapporten vil gå igjennom en prosjektbeskrivelse, hvor vi legger frem initielle krav og løsninger til problemstillingen. Den vil så gå igjennom ulike løsninger og tenkt design av valgt løsning, valg av verktøy knyttet til dette, risikovurdering, og evalueringsplan. Videre vil det blir forklart hvordan løsningen er laget, og hvilke funksjoner den har i detaljert design kapittelet. Dette blir fulgt av kap. 5, hvor vi legger frem evalueringer vi har gjort og fått i prosjektet. Disse blir oppsummert i kap. 6 resultater med hensyn på oppdragsgivers sluttevaluering. Kap. 7 diskuterer resultatene og prosjektet i sin helhet. Kapittel 8 inneholder konklusjon og forslag til videre arbeid. Kapittel 9 inneholder alle referanser, og kap. 10 alle vedlegg i rapporten.

## 1.6 Etske problemstillinger knyttet til oppgaven

I virkeligheten er en gjenopplivningssituasjon en svært alvorlig og stressende situasjon. Den kan oppleves traumatisk for de involverte partene, og kan trolig ha lengre ettervirkninger på psyken til de involverte. Oppdragsgiver ved Katrine Aasekjær har forklart hvor alvorlig studenter tar temaet og hvordan de har reagert med gråt når de ikke har fått til gjenopplivningen.

Grunnet den alvorlige situasjonen er det viktig at simulatoren oppleves som trygg for studentene. Det er viktig at vi prøver å unngå å skape dårlige minner eller traumer fra bruk av simulatoren. For å prøve å unngå dette bør ikke simulatoren gjøre det mulig at den nyfødte kan dø eller få forverret tilstand, i forhold til studentene sine handlinger. Dette kan trolig oppleves svært negativt og traumatisk.

I virkeligheten fører dessverre ikke alle gjenopplivningsforsøk frem.

“If there are no signs of life after ten minutes of effective cardiopulmonary resuscitation, it is appropriate to consider discontinuing resuscitation. One study showed that, in cases where this limit was exceeded, 88 out of 94 (94 %) patients either died or had severe motor impairments.” <sup>[16]</sup> (Boldingh, Solevåg, Nakstad, 2018, avsnitt 12)

Med tanke på dette bør det vurderes hvor lenge man skal tillate et gjenopplivningsforsøk å fortsette i simulatoren, er det riktig å ha et scenario hvor en nyfødt vil vise livstegn og

reaksjon etter lang tid? Bør det være opp til studentene selv å velge å avslutte resuscitering eller bør simulatoren avslutte scenarioet selv?

Det er essensielt at studentene blir trent riktig med tanke på arbeidet deres når de er ferdigutdannet, hvor feil valg og teknikk kan få fatale konsekvenser. Derfor er det svært viktig at treningen simulatoren tilbyr er helt korrekt og har høy kvalitet for at den skal tas i bruk. Ingen i gruppen er kvalifisert til å vurdere dette, så det er svært viktig at oppdragsgiver kvalitetssikrer, risikovurderer og redegjør for om simulatoren er forsvarlig. Hvis simulatoren fører til at studenter senere i arbeidslivet gjør feiltrinn i gjenopplivningssituasjoner vil det være veldig alvorlig.

## 2 Prosjektbeskrivelse

I dette kapitlet beskriver vi bakgrunnen til prosjektet, hvem prosjekteier er, og hvilke initielle krav de hadde til prosjektet.

### 2.1 Prosjekteier

Prosjektets eier er instituttet for helse- og omsorgsvitenskap.

Instituttet tilbyr bachelorutdanning, videreutdanning og masterutdanning innen sykepleie, masterutdanning i jordmorfag og tverrfaglige videreutdanninger og masterutdanninger innen helsefag. Instituttet samarbeider tett med kommunene og helseforetakene i regionen både når det gjelder utdanning og forskning. Forskningen har klar relevans for fagfeltet og arbeidslivet de utdanner til. Instituttet holder til i Førde, Bergen, Stord og Haugesund og har 2400 studenter og om lag 210 tilsette. <sup>[13]</sup>(HVL, 2020)

Instituttet har åtte fagseksjoner, der en av disse er Master i jordmorfag. I praksis vil det være denne fagseksjonen som vil være eier av prosjektet. Kontaktpersonene våre jobber i denne fagseksjonen og er initiativtakerne for hele prosjektet. Før prosjektet ble presentert som et bachelorprosjekt for oss, arbeidet kontaktpersonene med prosjektet på kveldstid. De fant fort ut at dette ikke var mulig å gjennomføre selv og valgte derfor å foreslå det som et bachelorprosjekt.

Fagseksjonen Master i jordmorfag utdanner sykepleiere til å bli jordmødre. Faget har som mål å forberede studentene på alle arbeidsoppgavene en jordmor har. Dette innebærer gjenopplivning av nyfødte. Som nevnt tidligere er det begrenset hvor mye trening det er mulig å få til i løpet av utdanningsforløpet.

Treningen på gjenoppliving fram til nå består av kurs som blir holdt av instruktører med mye erfaring innen feltet. Kursene er tidkrevende og blir derfor ikke holdt ofte. Det fører til at studentene ikke får ønsket mengdetrening.

Ifølge oppdragsgiver ved Katrine Aasekjær er det hennes erfaring at under øving med dukke tar studentene oppgaven veldig på alvor, og mange blir veldig stresset, og at de kanskje gjør mer feil på grunn av dette. I en reell situasjon kan det få katastrofale følger om jordmor gjør feil valg i en gjenopplivningssituasjon. En mulig forbedring på problemet er å gi studentene mengdetrening.

Ved å tilby studentene et VR-spill der de kan øve på diverse scenarioer med gjenopplivning kan dette åpne muligheten for mengdetrening. VR brillene er portable og trenger ikke noe ekstra utstyr, slik som en PC. Dette gjør at studentene vil kunne øve nesten hvor som helst. Det frigjør også tiden til instruktørene, ettersom det ikke kreves noen instruktør for å starte eller gjennomføre spillet.

I ett VR-miljø kan man simulere situasjoner som er sentrale for en jordmor å observere, som kan være vanskelig eller umulig å få til med dukke. Dette kan være endring i hudfargen til babyen (blir en baby blålig i huden, så kan det tyde på lite oksygen), lyd eller

muskelspenning. Ved å simulere dette i spillet vil det gjøre det mulig for studenten til å oppdage symptomene av seg selv, uten å få tips fra en instruktør.

Prosjektet er viktig for prosjekteieren fordi de ser et behov for mer mengdetrening hos sine studenter når det gjelder gjenopplivning av nyfødte. Ved å få til dette prosjektet vil kostnaden gå ned og kvaliteten vil bli bedre.

## 2.2 Initiale krav

### **Ikke-funksjonelle krav:**

Oppdragsgiver sine initiale ikke-funksjonelle krav er at et spill skal utvikles i Unity, dette skal primært utvikles for VR bruk.

### **Funksjonelle krav:**

De initiale hovedkravene til prosjekteier er å lage et VR-spill som brukes for å trene studenter i gjenopplivning av nyfødte. Spillet skal ta utgangspunkt i NRR sin algoritme for gjenopplivning av nyfødte. <sup>[1]</sup>(NRR, 2015). Barnet i spillet skal kunne vise alle kriteriene til en Apgar Score. <sup>[4]</sup> (SNL, 2018).

Spillet skal bli laget slik at studenten observerer babyen sine symptomer, og gjør riktige tiltak ut ifra disse symptomene. De riktige tiltakene er definert av NRR og ved å utføre disse tiltakene vil spillet gå videre. Hensikten med dette er at studentene skal lære seg NRR algoritmen ved hjelp av spillet. Ved at studentene spiller mange ganger kan det føre til at de vil huske bedre hva som skal gjøres i forskjellige situasjoner. Tid er en veldig viktig del av algoritmen, dermed kan det redde liv i en reell situasjon at jordmødrene har algoritmen i “ryggmargen”.

I spillet skal det være mulighet for studenten å starte en stoppeklokke. I praksis skal jordmoren starte stoppeklokke før gjenopplivningen starter. Dette er viktig for å kunne følge NRR sin algoritme presist, da den inneholder tids-spesifikke veivalg.

Et av ønskene til oppdragsgiver er å ha en sluttevaluering for studenten. Dette skal være en oversikt over hvordan det gått underveis i spillet. Her skal det vises hva studenten har gjort, om det var riktig eller galt. I tilfelle studenten har gjort noe feil vil det bli formidlet hva som var det riktige å gjøre i den aktuelle situasjonen. Det kan være flere fordeler med å implementere en sluttvurdering i spillet med tanke på læringsutbytte. Dette vil fortelle studenten hva som ble gjort riktig, og hva studenten må øve mer på. Uten en form for vurdering vil studenten være uvitende om sine feil og får svært lite læringsutbytte.

Det er også et krav om at studenten aldri skal forvolde skade på barnet, og at spillet på en eller annen måte vil hjelpe eller ta over for studenten om valg som kan forvolde skade skjer. Dette er for å begrense frykten og eventuelle traumer studenten kan oppleve av å ikke klare korrekt gjenopplivning. Et forslag for å unngå at ting kan gå galt er at spillet kan stoppe/fryse hvis studenten ikke har gjort det riktige valget. Da får studenten tid til å tenke seg om og prøve seg frem for å klare å ta det riktige valget. Det kan diskuteres om dette er det best mulige måten å designe spillet på. Fordelen er at studenten får rom til å tenke seg om, og spillet vil ikke være

like “vanskelig” og dermed vil terskelen være lavere for å spille på nytt. Ulempen kan være at spillet ikke blir like nær realiteten. I virkeligheten vil ikke jordmoren ha tid til å stoppe opp å tenke, og ved å implementere stopp vil det gjøre at studenten ikke får trene på å jobbe under tidspress. Med oppdragsgiver ble det diskutert om simulatoren skulle ha ulike modus for læring med gradert vanskelighetsgrad.

At studenten får tips underveis i spillet er også et av de initielle kravene til prosjekteier. Hvis studenten har glemt å trykke på en knapp, kan den for eksempel begynne å blinke. Dette kan hjelpe hvis studenten glemmer hva som er det riktige å gjøre.

Prosjekteier ønsker i tillegg til sluttvurderingen at studenten skal gjennomføre en egenvurdering på slutten av spillet. Ved å implementere dette ønsker prosjekteier at studenten selv skal reflektere over hva som er blitt gjort, og hva som kunne gått bedre. Tanken bak dette er for å få maksimal læring ut av hver spillscenario. Trolig kan dette gjøres på papir etter bruk av simulatoren.

# 3 Design av prosjektet

## 3.1 Forslag til løsning

### 3.1.1 VR simulator basert på Unity, simulering av gjenopplivning



Figur 1: Medisinsk asfyksibord, Oculus Quest

Løsning 1 består av å lage en VR-simulator i Unity sin plattform for spillutvikling <sup>[2]</sup> (Unity, 2020), der vi tar i bruk Oculus quest VR briller <sup>[3]</sup> (Oculus, 2020), med grunnlag i NRR sin gjenopplivningsalgoritme. <sup>[1]</sup>(NRR, 2015). Simulatoren skal kunne simulere ulike scenario knyttet til gjenopplivning av en nyfødt.

Studenten blir presentert med et bord som er likt det asfyksibordet som blir brukt på sykehus for gjenopplivning av nyfødte. Bordet består av flere knapper og utstyr som er nødvendig for gjenopplivning. Det vil være en stoppeklokke som studenten skal starte før noen annet gjøres. Det vil også være mulighet for studenten å tilføre oksygen til babyen. Dette vil skje med en slange festet til bordet, med en maske i enden som kan tas over munnen på babyen. Ved bruk av slangen må studenten skru på en knapp på bordet for øke oksygenmengden til babyen. Det skal også være en varmelampe som er plassert over bordet som studenten kan aktivere for å tilføre varme.

Babyen skal framstille symptomer som studenten må observere. Det er flere symptomer babyen skal kunne vise. Symptomet på oksygenmangel skal være at babyen forandrer farge. Den skal kunne vise grad av tonus ved hvordan den beveger armer og ben. Babyen skal kunne lage lyd, eller skrike for å vise om den er ved bevissthet.

Ved å ta i bruk Oculus Quest VR briller vil studenten bruke håndkontrollere for å kunne bevege armene, eller ta på babyen i spillet. Det er knapper på kontrollene som vil bli tatt i bruk for at studenten skal kunne gjøre ulike håndbevegelser, som for eksempel gripe.

Spillet skal starte ved at studenten blir satt i et rom med en livløs baby liggende på asfyksibordet. Babyen vil presentere symptomer som studenten skal kunne gjenkjenne. Spillet vil ikke gå videre med mindre studenten utfører de korrekte handlingene som definert i NRR sin algoritme for gjenopplivning av nyfødte. <sup>[1]</sup> (NRR, 2015)

Disse handlingene kan være:

- Tilføre oksygen til babyen fra tilførselen på bordet
- Gni babyen med et håndkle for å stimulere den
- Tørke babyen og bytte håndklær slik at den ikke blir nedkjølt
- Utføre hjerte- og lungeredning

På slutten av spillet vil studenten bli presentert med en sjekkliste der studenten skal evaluere seg selv for hver del av spillet. Etter sin egen evaluering vil studenten bli presentert med en evaluering av hvordan det faktisk gikk, ut ifra hva som ble gjort underveis i spillet.

### **3.1.2 AR løsning ved hjelp av AR headset og dukke**

Ved hjelp av en AR-brille eller VR-brille med støtte for AR funksjonalitet kunne man laget en simulator som kan kombineres med en medisinsk treningsdukke. Simulatoren kan da gjøre det mulig å simulere korrekt gjenopplivning med fysiske grep som brystkompresjoner. En annen mulighet her er å få data fra dukken inn i simulatoren også, og dermed kunne ha to-veis kommunikasjon og justering mellom dukke og simulator. Såvidt vi vet er de fleste dukker proprietære med lukkede løsninger, som vil vanskeliggjøre kommunikasjon uten samarbeid fra en leverandør. Hensikten med simulatoren er hovedsakelig å trene studenter i NRR algoritmen, ikke i korrekt brystkompresjonsteknikk, dermed blir kanskje dukke simulering en avsporing.

Det er en del begrensninger knyttet til en slik løsning, Oculus Quest headsettet støtter ikke AR på en god måte, og tilbyr kun sort-hvitt bilde av omgivelser. Omgivelsene må spores nøyaktig for at den virtuelle verden og virkeligheten hele tiden er synkronisert. Dette er teknisk vanskelig å få til gitt teknologien vi har tilgjengelig i dag.

### **3.1.3 App for trening på resuscitasjonsalgoritmen**

Et tredje alternativ til VR spillet kan være å lage en app som tester studentene mot NRR sin algoritme og gitte scenarioer med en nyfødt. Det kan f.eks. være et bilde av en nyfødt, med ulike parametere som er spesifisert i en oppgavetekst. Så er det opp til studenten og velge riktige gjøremål og metoder for å gjenopplive babyen. Etter endt sesjon så blir studenten presentert med en sluttrapport som viser hva som ble gjort, og hva som er anbefalt å gjøre. Ved flere ulike scenarioer kan studentene over tid bli flinkere til å ta riktige valg i gjenopplivningssituasjoner.

### **3.1.4 Diskusjon av alternativene**

Hensikten med prosjektet er å forberede jordmødre på veldig stressende og vanskelige situasjoner rundt gjenopplivning av nyfødte, noe det er ønske om at de skal få mer trening i. Med tanke på dette vil sannsynligvis VR løsningen gi best uttelling for studentene. AR løsningen har mange begrensninger knyttet til implementasjon og et bredere omfang enn den tenkte VR løsningen. App løsningen vil trolig ha lite ekstra effekt i forhold til treningen jordmorstudentene allerede driver med. Ifølge oppdragsgiver ved Katrine Aasekjær er de

fleste studentene veldig dyktig på NRR sin algoritme, og har lite problemer med å følge denne i situasjoner med normalt stressnivå og normal sensorisk inntrykksmengde. Det er først i realistiske og stressende situasjoner at studentene gjør feiltrinn i forhold til algoritmen.

Sett i lys av dette og gitt at oppdragsgiver allerede hadde utforsket og klargjort problemstillingen for en VR-løsning, er det denne løsningen vi har gått videre med.

## 3.2 Valg av verktøy

Ressursene som blir brukt i prosjektet består av spillutvikling plattformen Unity<sup>[2]</sup> (Unity, 2020), NRR sin algoritme for gjenoppliving av nyfødte<sup>[1]</sup> (NRR, 2015), Oculus Quest VR briller<sup>[3]</sup> (Oculus, 2020), og 3d arbeidsprogrammet Autodesk 3ds Max.

Unity er en spillutvikling plattform som tillater brukeren å utvikle spill til mange forskjellige systemer. I Unity har man mulighet til å lage todimensjonal-, tredimensjonal-, virtuell (VR) og utvidet realitet (AR) spill.

Prosjekteier hadde fra før erfaring med Unity, og et ønske om å kunne fortsette på prosjektet senere. Det var derfor naturlig å velge Unity for prosjektet. I tillegg til dette har Unity ekstra funksjonalitet tilgjengelig for studenter som registrerer seg med sin student epost.

Målet med hele prosjektet er å trene studenter på NRR sin algoritme for gjenoppliving. Den er blitt utviklet for utføre gjenoppliving på et nyfødt barn etter dagens beste anbefalte teknikk. Algoritmen består av ulike tiltak som skal utføres ut fra hvilke symptomer babyen har.

Prosjekteier har tatt valget om å bruke Oculus Quest VR briller. Disse brillene består av selve headsettet og to håndkontrollere. Brillene er ikke de kraftigste brillene på markedet, men passer bra for prosjektet. Grunnen til at det passer bra er at det ikke er fysisk stort. Det trenger ikke å koble seg opp til en datamaskin for å kjøre fordi spillet vil bli kjørt på selve brillene. Dette gjør at spillet kan bli spilt nesten hvor som helst, så lenge det er nok fysisk plass tilgjengelig.

Autodesk 3ds Max er et program som har blitt brukt til spillutvikling i de aller fleste moderne spill. Det blir sett på som en industristandard i spillindustrien. Det gjør det mulig å lage svært detaljerte 3D-modeller, materialer, animasjoner, og rigger for animasjon. I prosjektgruppen er det kun Joakim Andås Johesan som har erfaring med programmet fra tidligere. Det var naturlig å bruke det sett mot for eksempel Blender med tanke på at en i gruppen var kjent med det fra før. Programmet er gratis for studenter, men har vanligvis en lisenskostnad i prissegmentet 20 000 kr per lisens i året.<sup>[7]</sup> (Autodesk, 2020)

## 3.3 Prosjektmetodikk

### 3.3.1 Utviklingsmetodikk

Vi har valgt å ta i bruk utviklingsmetoden Kanban<sup>[14]</sup> (Agile Alliance, 2020) i vårt prosjekt. Kanban er en variant av Lean metoden. Lean handler om å minimere sløsing av ressurser, og fokusere mer på hva kunden ser på som verdifullt. Hovedfokuset i denne metoden er å skape



merverdi ved bruk av minst mulig ressurser. Kanban er en metode som handler om å begrense antall arbeidsoppgaver som blir igangsatt i et prosjekt. Det er et køsystem der det blir definert et maks antall jobber som er tillatt å starte. Hvis man skal starte flere er man nødt å fullføre en av de allerede oppstartede jobbene. I prosjektet løste vi dette ved at vi til enhver tid ikke hadde mer enn en oppgave pågående per person.

Grunnen til å ta i bruk Kanban er at det er begrenset tid for å fullføre prosjektet, og det er derfor en fordel å ha en slank “produksjon”. Det er også en fordel å følge Kanban med tanke på at vi ikke har mye kompetanse innen VR utvikling. Det er da viktig å få til fundamental funksjonalitet før man går videre med mer avanserte deler av spillet. Kanban er en form for agil utvikling, men den krever ikke direkte at man lager iterasjoner av produktet.

I praksis vil det å implementere Kanban være å ta i bruk Github sin funksjon for prosjekter. Der kan man definere kolonner som f.eks. “to do”, “in progress” og “done”. Dermed kan brukere legge inn arbeidsoppgaver som trengs å gjøres, og flytte de etter hvert som de forandrer status. På denne måten vil alle i prosjektet lett få oversikt over hva som status i prosjektet.

Senere i prosjektfasen valgte vi å bytte ut Github Projects med Trello boards. Se kapittel 7.1 for mer informasjon.

### **3.3.2 Risikovurdering**

Risikovurdering er et forsøk på å forutse fremtidige problemer og arbeide for å forhindre problemene med høyest risiko. (Risiko = Konsekvens×Sannsynlighet). I dette kapittel prøver vi å kartlegge risikoene i prosjektet.

I dette prosjektet har vi i utgangspunktet mange risikoelementer, spesielt knyttet til utviklingsdelen. Ingen av oss har erfaring med utvikling med Unity. Joakim har noe erfaring med 3d-modellering og grafikk, men ingen har erfaring med utvikling mot Oculus Quest. Sett i forhold til det medisinske, er det heller ingen tidligere erfaring i gruppen.

Gitt disse elementene er trolig risikoen undervurdert allerede fra starten av. Rett før kick-off i prosjekt ble også Covid-19 deklart en global pandemi av WHO. Underveis i prosjektet har vi forsøkt å kontrollere risiko så mye som mulig. Samtidig har vi opplevd at flere av punktene vi har vurdert som risikoer tydelig viste seg i prosjektet. Dette blir diskutert i [Kapittel 7](#).

## Risikoliste (#)

1. Skole og arbeidsplasser blir stengt
2. Unity er betraktelig vanskeligere å arbeide med enn forutsatt
3. Vanskelig å få lagd nødvendige modeller og materialer til simulatoren
4. Koden / ressurser er for ineffektive til å kjøre bra på Quest headsettet
5. Animasjon kan være veldig vanskelig
6. Kode kvalitet for gjenbrukbarhet og lett utvidelse
7. Sykdom i gruppen

#	Sannsynlighet	Konsekvens	Risiko	Tiltak
1	1. Har aldri tidligere skjedd i moderne tid	3. Lavere arbeidseffektivitet, lang utviklingstid	3	Klar plan for arbeid hjemmefra
2	2. Grim Thomas har tidligere erfaring fra andre spillmotorer	3. Lang utviklingstid, kort prosjektid	6	Undersøke tidligere VR arbeid i Unity, lese seg opp på utvikling i god tid, spørre om hjelp fra instituttet
3	1. Joakim har tidligere erfaring med å lage modeller og materialer, samt bruk i spillmotor three.js	3. Lang utviklingstid, dårlig resultat	3	Ivar kunne bidra med modellering og utvikling. Undersøke krav og metodikk til 3d assets på forhånd
4	2. Ingen tidligere erfaring med optimalisering for spill i gruppen	2. Sannsynligvis lett å skalere tilbake ressurser / features i kode	4	Bli kjent med begrensninger i Quest headsettet slik at vi er klar over disse tidlig.
5	4. Ingen tidligere erfaring i gruppen med animasjon av rigger.	1. Lite behov for animasjon av baby, da denne er livløs for det meste	4	Redusere mengden animasjon
6	2. Koden kan bli vanskelig å vedlikeholde eller utvide i etterkant eller mot slutten av prosjektet	1. Koster tid om det må refaktoreres	2	Refaktoring enten i etterkant eller mot prosjekt slutt.
7	3. Vanlig å bli syk, ikke vanlig med langvarig sykdom.	2. Ved kort forløp bør det gå greit. Ved langt forløp bør prosjektet justeres ned i omfang	6	Resten av gruppen må yte litt mer i sykdomsperiode. Ved lang sykdomsperiode vurdere å kutte ned på prosjektstørrelse

Sannsynlighet fra 1-4, Konsekvens fra 1-4. 1 er lavest, 4 er høyest

## 3.4 Evalueringsplan

Ved hjelp av evalueringer underveis og i slutten av prosjektet ønsker vi å få klarhet i om simulatoren fungerer som planlagt, om brukerne kan bruke den som planlagt, og om den oppfyller prosjektkravene til oppdragsgiveren. For å gjøre evalueringen er det behov for flere forskjellige evalueringsmetoder, disse vil spesifikt bli drøftet i underkapitlene som følger, med fordeler og ulemper relatert til planlagt evalueringsmetode.

### 3.4.1 Unit-testing

Unity støtter unit testing med Unity Test Framework <sup>[5]</sup> (Unity, 2020), dette benytter underliggende NUnit <sup>[6]</sup> (NUnit, 2020) rammeverket som er laget for testing av .NET kode.

#### Fordeler:

- Bedre kontroll på at koden vår fungerer som forventet
- Dokumentasjon på forventet oppførsel av metoder
- Raskere oppdaging av bugs, og fiksing av disse
- Reduserer sjansen for regresjons bugs

#### Ulemper:

- Gruppen har lite erfaring med UTF, NUnit og generelt Unity
- Erfaringen i gruppen er at vi ofte skriver tester som ikke tester relevant kode, samt at dette ikke fører til forbedret kvalitet i koden vi skriver.
- Erfaringsmessig fra gruppen tar det å skrive tester tar lang tid, ofte lengre tid enn selve koden som skal testes
- Testing mot VR interaksjoner kan bli ekstremt vanskelig å lage

### 3.4.2 Brukertesting

Ved brukertesting kunne vi funnet problemer relatert til interaksjon, flyt, og hendelsesforløp i simulatoren; og verifisert at dette fungerte tilfredsstillende.

Grunnet Covid-19 situasjonen er det usannsynlig at vi kan få til tester med brukere.

Det er også en mulighet å teste lokalt og vise video av dette til arbeidsgiver. Dette kan gjøres enten som en enkel video eller strømmes under et videomøte. Det er mest fordeler med å gjøre dette i et møte og da kan vi få spørsmål om hvordan ting fungerer “live”, og vise fram ting som eventuelt er aktuelt.

### 3.4.3 Akseptansetesting

Målet med evaluering her er å se om simulatoren oppfyller minimumskravene fra oppdragsgiver. Dette kan vi gjøre ved å få oppdragsgiver til å teste simulatoren, og gi skriftlig tilbakemelding innen rimelig tid før rapporten skal inn. Deretter kan vi også prøve å korrigere eventuelle feil eller problemer.

### **3.4.4 Vitenskapelig test av effektivitet**

En vitenskapelig metode for å evaluere effekten av simulatoren kunne vært å la en klasse med studenter bruke simulatoren på forhånd av en dukke-simulering og la en annen klasse opptre som kontrollgruppe, uten å ha brukt simulatoren på forhånd. Ved å la instruktøren kvalitativt vurdere resultat, evne, og korrekthet i simuleringen med dukke kunne vi vurdert om simulatoren i VR hadde ønsket effekt på studentene. Gitt at vi fikk inn flere klasser / grupper kunne vi begynt å se empiriske mønstre. Å vurdere disse resultatene er noe over vårt nåværende vitenskapelige- og statistiske kunnskapsnivå. Vi kunne trolig sett resultater, om indikatorene fra simuleringen med dukke var tilstrekkelig forbedret eller forverret.

### **3.4.5 Planlagt evalueringemetode**

I dette prosjektet planlegger vi å bruke brukertesting ved hjelp av oss selv, og akseptansetesting med oppdragsgiver. Brukertesting vil foregå ved at vi bruker simulatoren og prøver å oppdage problemer. Samt at vi strømmer bruk av simulatoren til oppdragsgiver og hverandre i videosamtaler for å få tilbakemeldinger. Akseptansetestingen vil bli gjort ved å levere en ferdig bygget applikasjon prototype som oppdragsgiver kan teste over et par dager, før vi vil gjøre et intervju med dem om hvordan de opplevde det samt diskutere forbedringer.

Det er sannsynlig at andre former for testing med flere testpersoner enn oss selv og oppdragsgiver vil gi bedre og klarere resultater. Grunnet Covid-19 pandemien ser vi oss nødt til å kun forholde oss til disse to metodene.

## 4 Detaljert design

### 4.1 Funksjonalitet i simulatoren

Det ble bestemt sammen med oppdragsgiver at Katrine Aasekjær skulle produsere et basis-scenario som vi skulle implementere i simulatoren. [Se [Vedlegg 10.5](#)] Dette krevde et sett med minimumsfunksjonalitet i simulatoren som vil beskrives i underkapitlene videre.

#### 4.1.1 Tidsinformasjon til bruker



Figur 2: Klokke på veggen og stoppeklokke i simulatoren

I simulatoren er det en veggklokke som viser klokkeslettet, samt en stoppeklokke på asfyksibordet som skal startes når gjenopplivningen starter. Disse to er tenkt å være tilstrekkelig for å informere brukeren om tiden som har gått i simuleringen. Det var nødvendig å gjøre disse elementene større enn normalt slik at de er lett å tolke i VR miljøet, grunnet den begrensede oppløsningen på bildet i brillen.

Veggklokken er implementert ved å modellere en analog klokke som henger på veggen i simulatoren. Time- og minuttviserne blir styrt ved hjelp av et script. Scriptet henter nåværende tid fra enheten den blir kjørt på, og flytter viserne til den riktige vinkelen i forhold til hva klokken er.

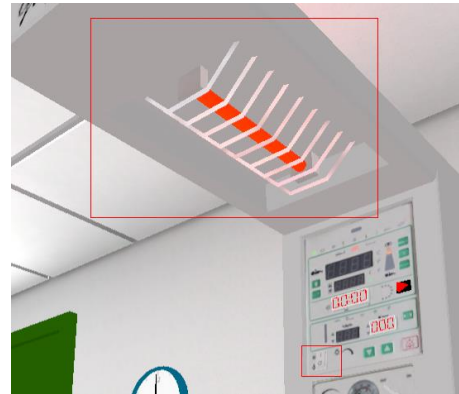
Stoppeklokken er modellert som en digital klokke. Klokken er en del av asfyksibordet med en start/stop knapp ved siden av, slik som det er i virkeligheten. Klokken har fire tall som skal representere minutter og sekunder. Med denne implementasjonen er det da mulig å vise tid fra 00:00 min:sek opp til 99:99 min:sek. Det er mulig å starte eller stoppe klokken både med å trykke på knappen i VR, eller ved å trykke på «B» knappen på tastaturet. Scriptet som styrer klokken har en variabel som forteller antall sekunder som har forløpt så langt. For hver frame sjekker den alltid om klokken er aktiv eller ikke (om studenten har startet eller pauset klokken). Hvis klokken er aktiv, vil scriptet inkrementere tidsvariabelen og oppdatere displayet etter hvert sekund den har vært aktiv. I tidligere versjoner av prosjektet var klokken

modellert som et fysisk korrekt display med geometri for hvert tall. Dette gikk vi bort fra da det krevde veldig mange tegnekall, og UI elementene som Unity tilbyr ga bedre ytelse, og så tilnærmet lik ut ved bruk av en «spesialisert» font.

### 4.1.2 Varmelampe

Asfyksibordet er utstyrt med en varmelampe som er montert øverst og peker ned mot madrassen. Det er tenkt at denne lampen kan stå på hele tiden under gjenopplivningen, men at studenten skal kunne slå den av eller på. Dette er relatert til kroppstemperaturen til den nyfødte, siden denne ikke må bli for lav eller for høy.

I simulatoren er varmelampen modellert som et vanlig lysrør i toppen av asfyksibordet. Det er mulig å slå på eller av lampen i VR ved å trykke på en knapp på bordet, eller ved å trykke på «H» knappen på tastaturet. Ved å slå på lampen vil materialet på selve lysrøret forandre seg slik at det er tydelig for studenten om den er av eller på.



Figur 3: Varmelampe i simulatoren

### 4.1.3 Pulsdisplay

Et av de viktigste parameterne studenten skal følge med på er pulsen til babyen. Asfyksibordet er utstyrt med en pulsmåler og et display som viser pulsen til babyen. Pulsmåleren kan bli festet på babyen, og pulsen vil da bli vist ved hjelp av det digitale displayet på bordet. Displayet er modellert som et digitalt display med tall, slik at man kan vise en verdi mellom 0-999. Dette skal være tilstrekkelig ettersom en normal puls er mellom 0-200. Displayet er styrt av et script. Dette scriptet gjør det mulig å simulere en puls i et gitt område (f.eks. 50-60 slag i minuttet). Den gjør dette ved hjelp av en tilfeldighetsgenerator som for hver frame bestemmer om pulsen skal gå opp, ned eller bli der den er.

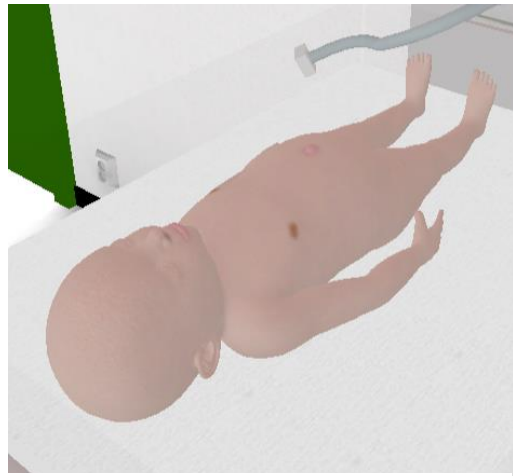


Figur 4: Kontrollpanel asfyksibord i simulatoren

#### 4.1.4 Det nyfødte barnet

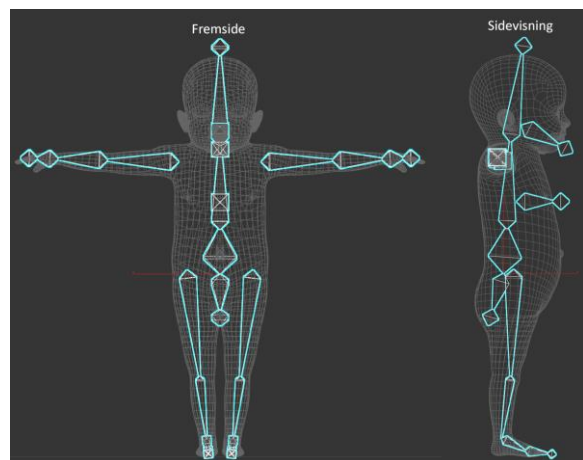
Et av hovedkravene til prosjektereier var at den nyfødte skulle kunne vise alle delene av en APGAR score. “Apgarscore er et system for rask vurdering av allmenntilstanden hos nyfødte.”<sup>[4]</sup>(Nesheim, 2018, avsnitt 1) For at studentene raskt skal kunne gjøre en Apgar vurdering av den nyfødte må tegn som: hjertefunksjon, respirasjon, muskeltonus, reflekser og hudfarge vises realistisk og tydelig.

Den nyfødte er i simulatoren rigget for bevegelse med et skjelett. Dette brukes for å flytte modellen, skape bevegelse, og modellere ledd og muskler. I starten av et scenario vil skjelettet og musklene være helt avslappet, og barnet vil se tilsynelatende bevisstløs ut. Skjelettet er satt opp som en ragdoll-body i fysikkmotoren til Unity, dette gjør at man kan flytte modellen realistisk og lemmene dens vil bevege seg naturlig som et slapt menneske. Ved korrekt gjenopplivning vil den nyfødte komme til live, stramme muskler og ledd, og puste selv.



Figur 5: Nyfødt barn i simulatoren

Skjelettet er bygget opp av bein som i virkeligheten. Disse er satt sammen i et hierarki, slik at en barnenode vil følge foreldrenoden sine bevegelser. Måten dette fungerer på er ved transformasjonsmatriser, og lokale koordinatsystemer for hvert enkelt bein. Koordinatsystemet definerer hvor beinet eksisterer i lokal-rommet ved hjelp av noder, og disse nodene kan transformeres rundt origo i koordinatsystemet. Origo kalles i praksis for pivot punktet. Ved å plassere pivot punktet til et barne-bein i endepunktet av et foreldre-bein vil en transformasjon av foreldre-beinet sitt koordinatsystem kunne ganges med lokal-matrisen til barne-beinet. Beina sin nye plassering vil synkronisert flytte seg, og pivot punktet til barne-beinet vil fortsatt være i den nye endeplasseringen av foreldre-beinet.



Figur 6: Barnemodellen i 3ds Max med rig

Unity sin fysikkmotor bruker “collidere” for å beregne omfanget til objekter, og sjekke om disse kolliderer med hverandre. Ved å legge collidere over beina i skjelettet vil disse kunne kolliderer med objekter i scenen og hverandre. Samtidig har beina plassert ledd i pivot punktene sine, og disse kan simulere ekte ledd, som kneledd, og skulderledd. Disse justeres slik at de får begrenset bevegelse og det settes fjærer i de som simulerer motstand, og muskler.

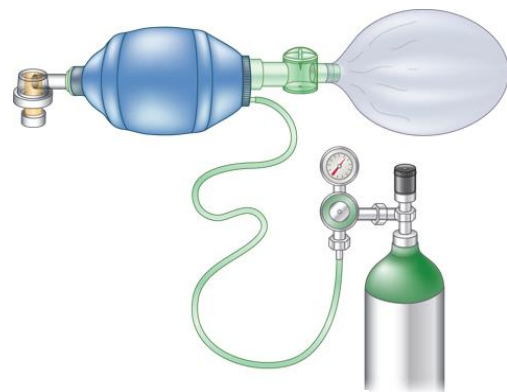
Ved å skalere lengden på beina i skjelettet i x, y eller z retning kan vi påvirke kroppsdeler slik at de blir lengre, høyere, eller bredere. Dette bruker vi for å simulere at brystkassen hever og senker seg ved pust. Vi endrer skaleringsfaktoren løpende for hver frame slik at dette blir en flytende og naturlig bevegelse.

For å simulere at hudfargen på den nyfødte endrer seg, bruker vi en standard Unity shader med en tekstur som definerer hudfargen. Ved å spesifisere en albedo farge annerledes fra hvitt, vil denne brukes for å endre fargenyansen på teksturen. I Unity er en bi-effekt av å endre albedo at hovedfargen blir mørkere. Vi oppnår likevel et godt resultat med en svak blåfarge. I script endrer vi denne prosentvis mellom hvitt og valgte blåfargen, slik at det kan være en jevn overgang mellom fargene.

En måte å vise hjerterytme hos barnet uten pulsoksymeter og display, gjøres ved å ha en collider som detekterer om hånden til studenten i VR blir plassert nært brystkassen til den nyfødte. Hvis hånden er innenfor området til collideren, vil håndkontrolleren til studenten vibrere i takt med hjerterytmen til den nyfødte.

#### 4.1.5 Ventilasjonsbagg, Lærdalsbagg

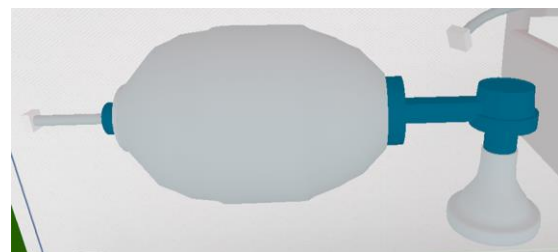
En ventilasjonsbagg er et verktøy helsepersonell bruker for å tilføre luft ned i lungene til pasienten manuelt. For å tilføre luft til pasienten må man presse på belgen slik at luft blir dyttet ut munnstykket. Når man slipper opp trykket på belgen vil den bli fylt opp med luft på nytt fra omgivelsene. Det er også mulig å feste på lufttilførsel på bagen slik at man kan ha en mer kontrollert tilførsel av luft.



Figur 7: Lærdalsbagg illustrasjon

Ventilasjonsbagen er utstyret studenten er nødt til å ta i bruk kunne bruke for å tilføre oksygen til babyen. For at studenten skal kunne feste ventilasjonsbagen på babyen må de holde kjevegrep og holde munnstykket i nærheten av babyen sin munn. Hvis studenten gjør dette vil det komme opp «tooltip» slik at de blir informert om hvilken knapp som skal trykkes på for å feste ventilatoren til babyen.

Når ventilasjonsbagen er festet til babyen er det mulig for studenten å trykke på baggen. Ved å trykke på baggen for å presse luft ned i lungene på babyen vil brystet til babyen heve seg. Dette er slik for å gi en indikasjon til studenten om at barnet får i seg luft.



Figur 8: Lærdalsbagg i simulatoren

Det er også mulighet å tilføre høyere oksygenprosent til babyen gjennom masken.

Dette gjøres ved å skru på ett hjul på asfyksibordet. Det er plassert en måler ved siden av dette hjulet som viser hvor høy oksygenprosent luften som blir tilført inneholder.



### 4.1.6 VR Interaksjoner

Interaksjoner i VR er hovedsakelig satt opp med “trigger colliders”, områder der kollisjon mellom to objekter ville foregått, men uten noen fysisk kollisjon. Hendene har hver sin sfære for både interaksjoner og for å kunne plukke opp gjenstander. Objekter som ønsker interaksjon med disse sfærene kan definere metoder som er håndtert av Unity, disse er “OnTriggerEnter” og “OnTriggerExit”. Disse blir automatisk kalt av Unity når to av disse volumene overlapper og når de slutter å overlapse.

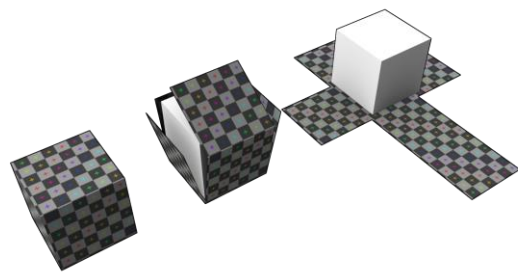
Knappene på bordet og babyen bruker dette systemet for å håndtere forskjellige interaksjoner. For eksempel er det en liten sfære i munnen på babyen som bruker dette for å sjekke om hånden er nær nok til å utføre et kjevegrep. Så lenge hånden er i dette område har brukeren mulighet til å holde nede en knapp for å holde kjevegrepet.

Mange av interaksjonene har også “tooltips” som er relevant til hva som skal gjøres. Disse er vist ved siden av hånden som er relevant for interaksjonen. Dersom brukeren strekker ut høyrehånden for å skru på varmelampen vil det vises tekst til høyre for hånden. Teksten vil forklare hvilken knapp som er relevant for å utføre handlingen, og hvilken handling som blir utført. Noen av interaksjonene, som å holde en gjenstand, krever at knappen holdes nede. Skruknappen på bordet krever at knappen holdes nede, samt at brukeren roterer hånden i retningen som ønskes.

## 4.2 Modeller, teksturer, rigger og lyssetting

Alle modellene i rommet har blitt modellert med poly-modellering <sup>[8]</sup>(Autodesk, 2016). Poly-modellering brukes som regel for å fremstille modeller med relativt lav oppløsning. Dette er en vanlig teknikk å bruke for å modellere til spillmotorer, på grunn av at spillmotorer har begrensninger knyttet til antallet polygoner de kan fremstille i sanntid. Poly-modellering er en tidkrevende og kunstnerisk prosess, som krever mange års erfaring for å produsere gode resultater.

Etter modellering må modellen og overflatene få teksturkoordinater, dette kalles UV mapping <sup>[9]</sup> (Autodesk, 2016). Alle modellene trenger UV mapping, selv om modellen kanskje kun skal ha en enkel farge. Dette er grunnet at «Global illumination» systemet til Unity er basert på lightmaps. Det betyr at indirekte lys beregnes på forhånd i editoren; deretter lagres dette i teksturer, som så legges over meshene for å belyse disse korrekt. UV mapping er i seg selv en veldig tidkrevende prosess hvor man lager sømmer i meshen som blir klippet, flatet ut og lagt i et todimensjonalt plan. Det krever en del finjustering og logisk tankegang på hvordan en tredimensjonal overflate kan brettes og klippes ut for å legges i et plan.



Figur 9: UV utbretting av en kube

Alle modeller som skal inneholde animasjon og deformasjon (bevegelse) av meshen trenger en rig. En rig består av bein som kan flyttes i forhold til meshen, og som påvirker posisjonen til nodene i meshen. <sup>[10]</sup> (Concept Art Empire, 2020). Etter at skjelettet er laget, må modellen “skinnes” og vektet i forhold til beina. Skinning betyr “å legge huden over beina”.

Konseptuelt forklart brukes meshen som huden til riggen, og huden følger riggen sine bevegelser. Dette er en tidkrevende prosess som skjer på node nivå. Hver node må ha sin påvirkning fra et bein vektet (verdi fra 0-1). For eksempel i kneområdet på baby modellen er det nødvendig at lårbeinet, og leggbeinet påvirker nodene gradvis mot kneet. Hver node i Unity kan maksimalt påvirkes av fire bein, men for Oculus Quest anbefales maksimalt to. <sup>[18]</sup> (Unity, 2019) Dette begrenser muligheten for myke og realistiske overganger. Prosessen helhetlig kalles rigging og utføres i en animasjonsprogramvare. I simulatoren er det den nyfødte og ventilatoren som må rigges, siden disse trenger flytende bevegelse og animasjon.

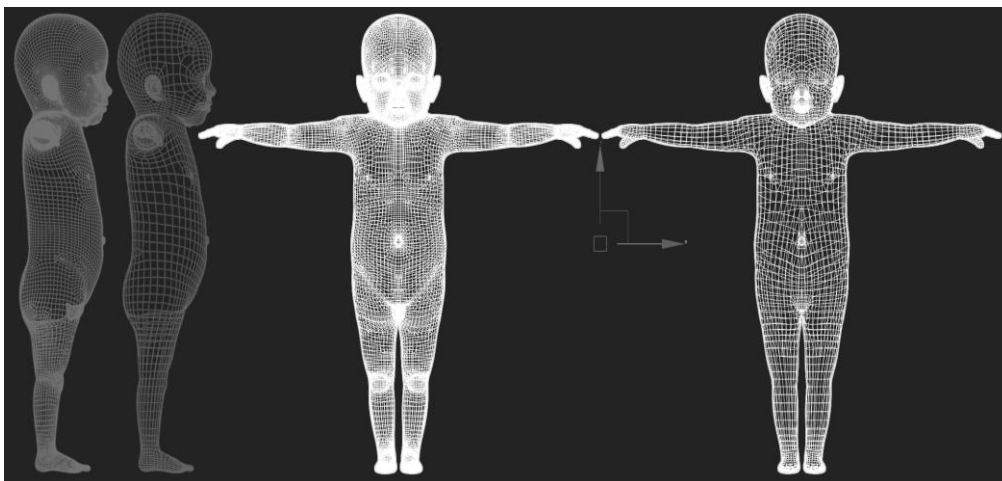
I Unity finnes det to typer lyssetting, direkte lys og indirekte lys. Direkte lys kalkuleres på nytt for hver bildetegning på skjermen. Indirekte lys bakes inn i teksturer som legges opp på modellene, og kalkuleres på forhånd. I virkeligheten er indirekte lys ofte like tydelig som direkte lys, det er dermed viktig å ha indirekte lys i simulatoren for å øke realismen. Direkte lys lager også veldig tydelige skygger og overganger som bør dempes.



Figur 10: Rommet i simulatoren i en tidlig versjon

Simulator scenen bestod av både direkte og indirekte lys, som vinduslys, lysarmatur, og spotlights. Sammen lagde disse en grei tilnærming til innendørs belysning på et sykehus. Ved kjøring på Oculus Quest ble vi oppmerksom på at ytelsen var såpass dårlig at vi måtte kutte ut alt direkte lys, grunnet at det har relativt høy kalkulasjonskostnad, sammenlignet med indirekte lys.

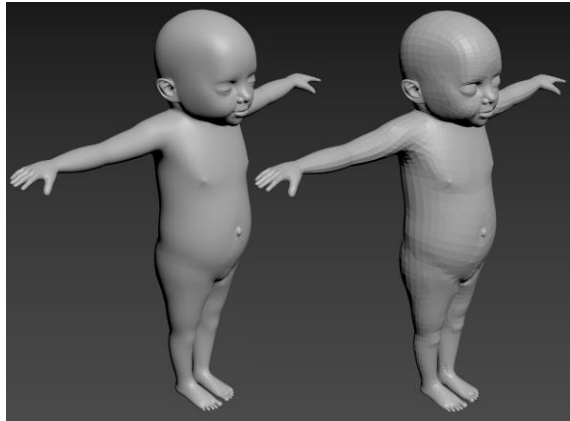
Ved nærmere ytelsestesting var det også tydelig at modellen av den nyfødte vi fikk av oppdragsgiver var for høyoppløst. Den hadde rundt 65 000 noder i meshen. For å lage en ny optimalisert mesh gjorde vi en “retopologi” av meshen. Dette innebar å tegne de individuelle



Figur 11: Babymodellen i 3ds Max, før og etter retopologi

flatene vi ønsket å ha oppå den gamle meshen. Ved å kun tegne halve modellen, kunne denne speiles igjennom sentralaksen for å spare tid.

Når man lager topologi i en mesh som skal brukes i en rigg er det viktig at man har jevn fordeling av flater. Helst bør alle flatene være firkanter. Grunnen til dette er blant annet at smooth-shading skal fungere riktig, og at modellen deformeres naturlig i bøyepunkter og ledd. Etter retopologien hadde meshen 12 000 noder. Dette var akseptabelt og ga god ytelse på VR headsettet.



Figur 12: Babymodellen i 3ds Max med og uten smooth-shading

### 4.3 Oppbygging av prosjektet i Unity

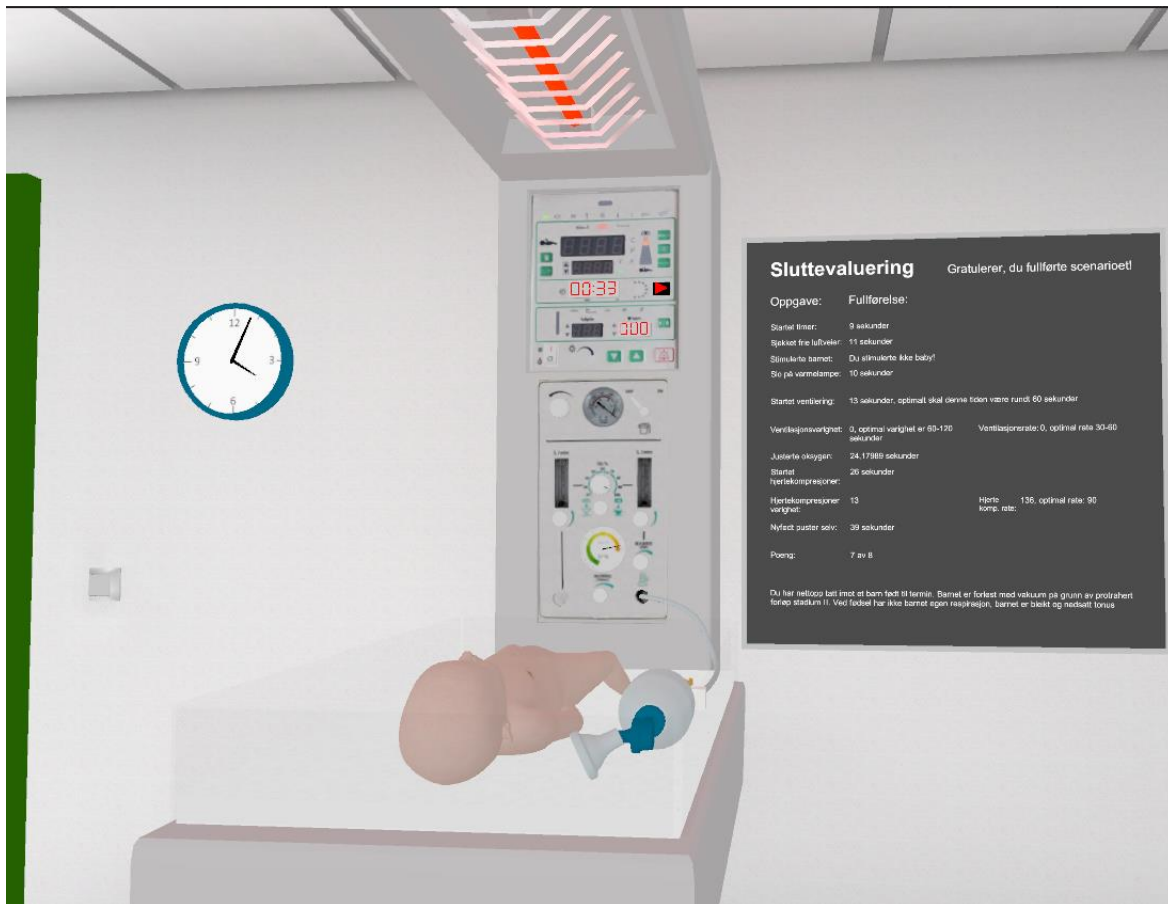
Unity prosjektet består av scener, scener kan forklares som spillbare områder eller omgivelser. Scenen består av et hierarki av GameObjects. Dette er base-objektet til alle andre objekter i Unity. Ethvert GameObject består i praksis bare av sitt eget lokale koordinatsystem. Et viktig element i Unity er scripts. Det er ved hjelp av scripts vi som utviklere bestemmer spillets gang, og hvordan ulike GameObjects skal oppføre seg. Scripts kan ikke kjøres om de ikke er festet eller kalt fra et GameObject. Dette gjør at man må ha en del “dummy” objekter i scenen for å kunne kjøre scripts.

### 4.4 Samhandlingen av kode i Unity

Scripts er C# filer som bruker Mono sin .Net Runtime. Mono er en open-kildekode-implementasjon av .NET rammeverket. Scripts må være lagt til som del-komponent av et GameObjects for å kjøre. En ulempe med dette er at det ikke er oversiktlig hvilke objekter som har scripts, og hvor kode kan ende opp med å kjøre fra.

Det er mulig å manipulere de forskjellige spillobjektene ved hjelp av scripts. For å få tilgang til objektene i selve scriptet må man lage offentlige variabler. Da kan man tildele spillobjekter til de ønskede variablene i Unity editoren.

## 4.5 Oppbygging av scenarioet



Figur 13: Fullført scenario i simulatoren

### 4.5.1 Kodemessig oppbygging i prosjektet

Unity sin arbeidsmetodikk er litt spesiell sammenlignet mot vanlig utvikling. Siden scripts er knyttet til fysiske objekter er det litt spesielt hvordan man sender informasjon rundt i koden. Informasjons-ekspert klasser ble istedenfor satt opp som “dummy” Gameobjekter som holdt informasjon. Objektene fikk offentlige felter for å holde referanser til andre objekter som er relevant. For eksempel “Scenario Objects” som holder referanser til baby, ventilator, og diverse andre objekter som er relevant til et scenario. Det er satt opp på denne måten slik at scenarioer ikke ender opp med samme antall offentlige felter selv, og for å gjøre det enklere å holde orden på referanser. Om en referanse bryter i “Scenario Objects” er det kun et sted å fikse den, istedenfor ti steder om vi hadde ti scenarioer.

Koden er hovedsakelig bygget opp med “kontroller” klasser som gjør små spesifikke oppgaver, eller til andre kontrollere som er relevant. For eksempel har “baby controller” referanse til “heart rate controller”, men også egne metoder for å håndtere hudfarge. Mye av koden går ut på å koble disse kontrollene sammen der det er nødvendig. Unity endrer ikke så veldig mye på hvordan enkapsulering av objekter fungerer, så det er mye det samme som vanlig arbeidsflyt. Hovedforskjellen her er at det føles mer “fysisk” i Unity, og mye av arbeidsflyten går gjennom å legge inn objekter via editoren og ikke via å kalle konstruktører.

Scenario er satt opp som en endelig tilstandsmaskin. Dette vil si at man alltid er i en “tilstand”, og man venter på en gitt hendelse for å gå til en annen tilstand. For eksempel er det en tilstand som heter “skru på stoppeklokke” som kun går videre til neste tilstand når stoppeklokken er skrudd på. Siden algoritmen for gjenoppliving er ganske lineær virket dette som en grei løsning. I noen tilfeller er det også mulig å ha “snarveier” fra en tilstand til en senere tilstand. Hvis brukeren glemmer å skru på stoppeklokken burde man kunne gå til en senere tilstand for å la scenarioet fungere som forventet, men gi “minuspoeng” for å ha glemt en viktig handling i spillet.

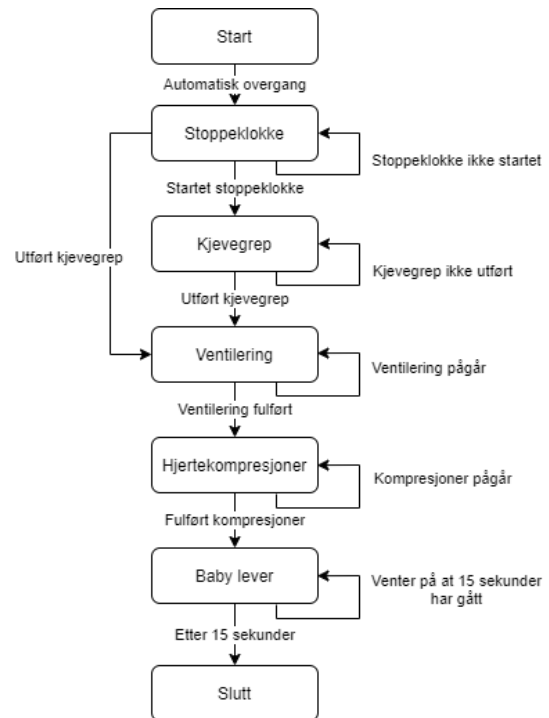
Mulighet til å ha flere scenarioer er lagt til med å ha predefinert start og slutt tilstander. Scenarioet vil alltid starte og stoppe automatisk i disse to tilstandene, alt imellom er definert via kode per scenario. Scenarioet har ellers full frihet til å gjøre det som er nødvendig. Scenarioer er derfor mer “globale”, disse startes og stoppes via en Scenario Manager. Scenario Manageren sin ansvar er å starte og stoppe scenarioer, samt holde rede på ekstra informasjon som hvor lenge de har kjørt. Dette gjør det enkelt for oss å ha et scenario som er fullførbart, men også et “fritt spillerom” scenario der brukeren kan bli vant til å bruke simulatoren.

Ventilering og hjertekompresjon frekvenser er regnet ut med å lagre siste X interaksjonene i en sirkulær liste. Antall plasser i denne listen er forskjellig i disse to tilfellene, men koden fungerer på samme prinsipp. Hver interaksjon er lagret som tidsstempel, dette gjør det enkelt å regne ut avstanden mellom dem. Hvor lenge brukeren har holdt på regnes ut med å ta tiden mellom siste aktivasjon minus den siste i listen som er relevant, altså uten en lang pause. Frekvensen kan da enkelt regnes ut med tiden som ble brukt og antall aktiveringer i denne perioden. Dette sjekker ikke om brukeren er konsistent mot forventet frekvens, men gjør en grei nok jobb over lengre perioder når man ikke tillater pauser.

## 4.5.2 Scenario

Ved oppstart av spillet vil studenten bli presentert med en meny der det er mulig å velge hvilke scenario man ønsker å spille. For hvert enkelt scenario står det litt informasjon om hva som foregår i det spesifikke scenarioet. For dette prosjektet har vi laget ett scenario, men i tillegg har vi lagt inn et åpent scenario for testing av simulatoren.

Ved å velge scenario og å trykke på start vil studenten bli satt inn i simulatoren og presentert med en baby som ikke puster, er blekt, har nedsatt tonus og en hjertefrekvens på 50 slag i minuttet. Herfra er det opp til studenten hva han/hun skal gjøre, men det er tenkt at studenten skal følge NRR sin gjenopplivnings algoritme.



Figur 14: Forenklet diagram av tilstandsmaskinen i scenarioet

Før studenten gjør noe annet skal han/hun starte stoppeklokken som er plassert på asfyksibordet. Deretter er det viktig å sikre frie luftveier for babyen. Dette blir gjort ved at studenten holder hånden over munnen på babyen og trykker på knappen som blir oppgitt i “tooltip”. Studenten skal slå på varmelampen. I virkeligheten er ofte varmelampen allerede slått på, men fordi studenten skal vise at han/hun har husket å slå den på er den avslått ved starten av spillet.

Det er valgfritt om studenten vil ta på pulsoksymeteret på barnet. Hvis studenten velger å gjøre dette vil hjerterytmen til babyen vises på puls displayet på asfyksibordet. Alternativt kan studenten holde hånden over hjertet til babyen, da vil kontrolleren riste i takt med hjerterytmen til babyen.

Etter at vurdering av babyen sin tilstand er ferdig, skal studenten begynne med ventilering. Dette skal bli gjort ved hjelp av ventilasjonsbagen. Før studenten fester ventilasjonsbagen til babyen er det viktig at det blir holdt kjevegrep. Her er det viktig at studenten utfører ventilering i mer enn 30 sekunder, med en rytme på ca. 30 til 60 innblåsing per minutt. Det er viktig at studenten presser ventilasjonsbagen godt nok sammen ellers vil ikke innblåsing være “godkjent”, og vil derfor ikke telle i de nødvendige 30 til 60 innblåsing per min.

Når studenten er fornøyd med ventileringen skal det bli tatt en ny vurdering av tilstanden til babyen. I denne vurderingen skal studenten observere at babyen har den samme tilstanden som ved starten av scenarioet. Dermed skal studenten fortsette med ventilering i 30-60 sekunder til. I simulatoren er det ikke et krav for at scenarioet skal bli gjennomført, men informasjon om hvor lenge man skal ventilere står i sluttrapporten. Deretter skal studenten sjekke hjerterefrekvensen på nytt, og ved denne observasjonen ser studenten at hjerterefrekvensen ikke har forandret seg.

Ettersom tilstanden til babyen ikke har forandret seg etter ventilering skal studenten starte med hjertekompresjoner, men før dette skal studenten øke mengden oksygen inn på masken. Under hjertekompresjoner skal barnet også ventileres, men dette kan studenten anta at blir gjort. Hjertekompresjoner blir gjort ved at studenten holder hendene sine over brystet til babyen og trykker på knappene som blir oppgitt i «tooltipet». Studenten må trykke på knappene for hver gang en kompresjon skal bli utført. Studenten skal tilstrebe en rytme på 90 kompresjoner per minutt, men spillet vil godta et veldig bredt spekter av rytme for å gjøre det enklere for studentene å ikke sette seg fast slik at de ikke kommer til sluttrapporten.

Når studenten har gjennomført godkjent HLR vil babyen sin tilstand forandre seg for å vise studenten at den er blitt frisk. Dette gjør babyen ved å ha en hjerterefrekvens som er over 100 slag i minuttet. Babyen vil begynne å gråte og den vil få synlig økt tonus.

Helt til slutt vil studenten bli presentert med en sluttrapport der de forskjellige stegene blir listet opp. For hvert steg vil det stå hvor lang tid det tok for studenten å bli ferdig med det aktuelle steget. For ventilasjon og HLR stegene vil det i tillegg bli oppgitt varighet og rate.

## 4.6 Bygge for Oculus Quest

Bygging mot Oculus Quest skjer som en applikasjon for Android i Unity, da headsettet kjører en modifisert Android versjon. Det er en del forarbeid som må til for å få mulighet til å kjøre

egen kode på headsettet. Headsettet må settes opp i Developer Mode, noe som krever “Oculus Developer” status på headsett brukeren. Dette tar en del tid å sette opp siden eieren av Oculus, Facebook, krever en god del personlig informasjon samt at de skjuler veldig mange innstillinger fra brukeren. For å skru på utviklermodus må man inn i brukerinnstillinger og skru det på. I tillegg må man skru på kjøring av usignerte apper.

Når alt er klart på brillene sin side er det kun nødvendig å sette Unity til å bygge mot Android, og godkjenne kommandoer fra Android Debug Bridge fra datamaskinen. <sup>[12]</sup> (Android, 2020) (ADB) ADB er et kommandolinje verktøy for å kommunisere med Android enheter. Det er ikke direkte brukt av oss, men det brukes blant annet av Unity for å lese ut logg og sende over appen til headsettet.

Etter dette er det bare å trykke på “Build and Run” i Unity mens brillene er koblet til med kabel. En av ulempene her mot andre VR headsett er at man ikke kan bruke “spill” knappen i Unity for å teste, som fører til mye venting når man lager små endringer. Det er mulig å jobbe rundt dette med å strøomme SteamVR <sup>[11]</sup> (Steam, 2020) fra Unity til headsettet, men dette bruker ikke samme innstillinger som å kjøre det rett på brillene. En annen ting som kan gjøres er å kjøre ADB over trådløst nett i stedet for kabel, men dette gir ikke raskere byggetid. Ulempen med dette er at det ikke fungerer bra med flere installasjoner av ADB, noe som man fort ender opp med da de fleste utviklingsverktøyene kommer med sin egen.

# 5 Evaluering

## 5.1 Evalueringsmetode

Under hele utviklingsperioden er det blitt gjort testing og evaluering av hver ny funksjonalitet som ble implementert. Det ble tidlig i prosjektet spesifisert hvilken funksjonalitet prosjekteier ønsket å ha i simulatoren. Evalueringen bestod i hovedsak av å teste de ulike funksjonene underveis i utviklingen, og evaluere om de var tilstrekkelige. Dette ble gjort ved starte simulatoren og teste den nye funksjonaliteten mot aktuelle brukstilfeller.

Grunnet Covid-19 situasjonen og at prosjektet var noe forsinket var det vanskelig for prosjekteier å teste og evaluere simulatoren underveis i utviklingen. Vi bestemte oss derfor for å bli ferdig med en prototype av simulatoren i god tid før prosjektet sitt sluttfrist. Meningen med dette var at da kunne prosjekteier teste og gi tilbakemeldinger. Hvis prosjekteier fant mye feil og ønsket mye forandringer hadde vi da god tid til å fikse opp i dette.

## 5.2 Oppdragsgivers underveisevaluering

Første prototype ble sendt til testing 15. mai 2020. Prototypen inneholdt første komplette scenario, med støtte for funksjonaliteten man trengte for å gjennomføre dette. Oppdragsgiver forsøkte så å gjennomføre scenarioet i simulatoren i henhold til deres fagkunnskap. Det ble raskt klart for dem at opplevelsen var svært lite brukervennlig. Bruksanvisningen inneholdt feil, og det var ikke tydelig hvordan interaksjoner skulle utføres. Som eksempel var det ikke tydelig om knapper skulle trykkes på, eller holdes inne. De hadde problemer knyttet til at det ikke var spesifisert hvilke knapper som gjorde hva, og at de måtte holde oversikt over dette selv. Dette følte de fjernet fokus fra den faktiske arbeidsoppgaven i simulatoren.

Oppdragsgiver opplevde også at simulatoren ikke var realistisk i henhold til modellen av asfyksibordet og ventilatorbaggen. Asfyksibordet manglet store deler av kontrollpanelet det har i virkeligheten. For oppdragsgiver var det et viktig moment at studentene skulle gjøre seg kjent med det utstyret de faktisk ville møte i virkeligheten. Ventilatorbaggen var for stor og manglet oksygenlange, i virkeligheten er det essensielt at oksygenlangen er koblet til.

Implementasjonen av scenarioet var også urealistisk, grunnet at det ikke var noen form for tilbakemelding fra barnet når man utførte ulike tiltak. F.eks. kunne de ventilere i henhold til algoritmen, og deretter utføre hjertekompressjoner i henhold til algoritmen uten å se noen forandring i barnets parametere.

Gruppen fikk mye veldig god konstruktiv tilbakemelding på hva som trengte forbedringer. Vi diskuterte prototypen i et møte med dem, og avtalte en ny testing med korreksjoner uken etterpå.

I neste versjon lagde vi en ny bruksanvisning [Se [Vedlegg 10.3](#)] som forklarte alle interaksjoner og knapper i detalj. Det ble også laget en plakat som forklarte knappe-layout, som ble hengt på veggen i simulatoren. Asfyksibordet sitt kontrollpanel ble implementert så virkelighetsnært som mulig, selv om all funksjonaliteten kontrollpanelet tilbyr ikke var implementert. Ventilasjonsbaggen ble justert i størrelse. Det ble lagt til en oksygenlange på



bag og asfyksibord som indikerer at de er koblet sammen. Det ble samtidig implementert flere former for tilbakemelding til brukeren:

- En positiv lyd som spilles av hvis brukeren gjør noe korrekt
- Pulsen på barnet forandrer seg mer i takt med brukerens interaksjoner
- Hudfargen på barnet forandrer seg i takt med brukernes interaksjoner
- Mulighet for å ta opp sluttevalueringsskjema ved å pause spillet, og se hva simulatoren har registrert har blitt gjort så langt.
- Mer informativ sluttevaluering, flere parametere blir vist til brukeren
- Reduserte krav for korrekt ventilering, og hjertekomprimering i forhold til tidligere. Brukerne får tilbakemelding på dårlig ventilering/komprimering i sluttrapport i stedet for at de ikke får gjenopplive barnet.

Oppdragsgiver var veldig positiv til den nye versjonen. “Utgaven er en mye bedre utgave enn det vi har blitt presentert for tidligere i prosessen. Mye av våre innspill er tatt hensyn til.”

Katrine Aasekjær, Ivar Rosenberg, oppdragsgivere i prosjekt (Evalueringsrapport, 27.05.2020)

Det var fortsatt noen justeringer som oppdragsgiver ønsket å få på plass. Disse måtte bli gjort i ettertid av at rapporten blir levert inn.

- Knappen for stoppeklokken var vanskelig å trykke på, måtte holde hånden inne i knappen før den kunne aktiveres.
- Stoppeklokken resetter seg ikke ved en omstart av scenarioet.
- Oksygenslangen var for stor dimensjon på, og burde plasseres på høyre side langs madrassen på asfyksibordet.
- Det var uklart hvordan stimulering skulle utføres i henhold til bruksanvisningen.
- Grepet om ventilasjonsbaggen ble plassert inne i håndleddet til VR spilleren, dette virket unaturlig.
- Kjevegrep kunne gli tilbake i baby modellen selv etter det var utført.
- Scenarioet følger ikke NRR algoritmen helt riktig. Studentene er ikke tvunget til å følge NRR algoritmen. Scenarioet oppleves som nært ønsket resultat, med kun få justeringer før det vil være tilfredsstillende å bruke i undervisning.

## 6 Resultater

Som en sluttevaluering utførte gruppen et videointervju med oppdragsgiver. Vi forsøkte her å diskutere oppgaven, resultatene, og eventuell måloppnåelse. Vi tok opptak av dette møtet og teksten under er en parafrasering av det som ble tatt opp.

### **Hvordan opplever du VR-miljøet med hensyn på realisme?**

Modellene i simulatoren oppleves som realistiske. Asfyksibordet er stort sett riktig dimensjonert, og gjør det mulig for studentene å gjøre seg kjent med den virkelige versjonen. Oksygenlange og ventilasjonsbag er litt ulik virkeligheten, men funksjonsmessig oppleves de realistisk. Babymodellen har litt merkelige dimensjoner i forhold til virkeligheten, men det ble ikke distraherende i forhold til arbeidsoppgaven. Rag-doll effekten skapte av og til øyeblikk hvor man ble løftet ut av illusjonen ved at denne plutselig skaper voldsomme bevegelser i barnet. Det ville vært ønskelig å ha mer realistisk bevegelse i barnet.

Lyden i simulatoren virker realistisk, men er kanskje ikke helt nødvendig. Katrine beskrev at man ikke legger merke til lydene rundt seg når man arbeider med resuscitering i virkeligheten. Hun la merke til bakgrunnslyden i simulatoren umiddelbart når hun startet simulatoren, men opplevde ikke at denne var i veien for arbeidsoppgaven. Ivar la ikke merke til lyden før Katrine kommenterte den, han opplevde den som til hjelp for å få studenten inn i riktig setting. Ventilatoren lager også en lyd når den komprimeres sammen. Lyden her var kanskje kraftigere enn i virkeligheten, men dette var en god indikator på at man ventilerer selv om man ikke ser på baggen.

Interaksjoner i simulatoren oppleves stort sett naturlig. Det er noen problemer knyttet til masken som skal festes på barnet da denne kan ende opp med å dingle frem og tilbake, eller suges inn i ansiktet til barnet, det virker distraherende fra arbeidsoppgaven i scenarioet. Sonene for å aktivere stimulering av barnet er vanskelig å treffe og overlapper med andre aktiveringssoner for hjertekompresjoner, disse har behov for justering.

### **Er det noe funksjonalitet dere savner?**

Katrine ønsker en mulighet for å kunne tørke barnet med et håndkle, som en videreføring av stimulering av barnet som er til stede i dag. Dette er et av de viktigste elementene av resuscitering, og ofte er det kun det som skal til for å hjelpe en nyfødt å puste.

Ivar ønsker at det skulle vært mulig at simulatoren stilte spørsmål til studentene underveis i scenarioet, for å tvinge studentene til å gjøre vurderinger i henhold til algoritmen og teknikker. Samt mulighet for at simulatoren tok time-outs hvis studentene satt seg fast eller gjorde feil. Hvor simulatoren ville forklart hva som burde gjøres, eller som ikke har blitt gjort.

### **Er det funksjonalitet du føler kunne vært løst på en bedre måte?**

Det er små justeringer som må til for å optimalisere brukeropplevelsen, men ellers er man fornøyd. Mer synlig at det går an å stimulere, fordel med å kunne plukke opp et håndkle.

### **Er det sannsynlig at simulatoren kan brukes til trening av jordmødre?**

Ikke i sin nåværende produktversjon. Men det er små justeringer som må gjøres for å gjøre dette godt nok til å brukes som et verktøy i undervisning. Selv om studentene kan ha nytte av simulatoren i dag er det å være føre-var og ha strenge kriterier til et nytt verktøy lurt.

Oppdragsgiver ønsker ikke at studentene plukker opp dårlige vaner som kan skape problemer i reelle situasjoner. Scenariot må være enda mer tro mot NRR retningslinjer og rekkefølgen i scenariot må gjøres mer faglig korrekt.

### **Hvordan opplever dere trygghet for brukeren i simulatoren?**

Simulatoren oppleves som et trygt miljø, det er usannsynlig at studenter vil få negative opplevelser i den. Studentene vil trolig få god mestringsfølelse etter å ha brukt den et par ganger.

### **Hvordan syns dere prosjektet har gått?**

Katrine ville satt pris på flere møter med oss. I prosessen har trolig begge parter sine behov for informasjon ikke blitt tilfredsstillt. Problemstillinger knyttet til simulatoren kunne kanskje vært unngått ved mer informasjonsdeling. Og opplevd fornøydhets og forståelse mellom partene kunne vært høyere ved hyppigere møter. Dette kunne trolig forbedret dynamikken mellom partene, og skapt mer tillit og åpenhet. Det ville vært ønskelig med tydelige forventningsavklaringer fra begge parter langs, slik at leverte resultater ville svart til alles forventninger.

Med tanke på Covid-19 pandemien ville det at gruppen hadde kontorplass hos oppdragsgiver trolig visket ut en del kommunikasjonsproblemer, men samtidig kunne dette vært forbedret av hyppigere møter.

Oppdragsgiver ønsker at vi utvikler prosjektet videre som en mastergrad evt. frilans arbeid senere.

# 7 Diskusjon

## 7.1 Utviklingsmetodikk, Github og Unity Collab

I prosjektet valgte vi å bruke Kanban som utviklingsmetodikk. I starten var planen å bruke Github Projects tavlen for å spore gjøremål og to-do's. I praksis ble ikke Github Projects brukt pga. gruppen ikke var veldig kjent med det, og ikke brukte tid på å sette seg inn i det. Når vi la merke til at metodikken ikke ble fulgt og Projects ikke ble brukt gikk vi raskt over til Trello boards som vi har brukt tidligere. Der fikk vi en god flyt på cards og det ble brukt aktivt i henhold til metodikken.

En annen grunn til at vi gikk bort fra Github er at vi sluttet å bruke det som repository. Unity prosjekter er ikke som andre kode prosjekter. Det er mange ressurser som kan ta veldig stor plass. Enkelte filer som scene filen inneholder referanser til alle objekter i scenen og relasjoner mellom dem. Dette førte til enorme problemer om to personer jobbet samtidig, siden scene filen måtte merges. Det er over 15 000 linjer i denne ene filen, og selv om den er lesbar av mennesker inneholder den veldig mye informasjon som er vanskelig å forstå og som er vanskelig å bestemme hvor skal plasseres. Gitt disse problemene med konstant merging gikk vi over til å bruke Unity sin Collab funksjonalitet for delt-arbeid på samme prosjekt.

Unity Collab løste som regel merging problemet vårt, siden den automatisk prøver å merge scene filer og andre spesielle filer relatert til Unity uten inngrep. Problemet er at verktøyet på sitt beste kan anses som i alfa versjon. Det henger kontinuerlig seg opp, uten noe tegn til å prosessere. Det endrer build-innstillinger som er vitale for at prosjektet skal fungere på Oculus Quest helt av seg selv. Det har ført til mye feilsøking og builds som ikke har fungert (hver build tar opp mot 20 minutter). Den kontinuerlige hengingen av UI-tråden gjør at Unity blir ubrukelig mens man må vente. Gjentatte problemer med dette førte til at gruppen raskt ble demotivert.

## 7.2 Begrensninger knyttet til lav ytelse på Oculus Quest

I den originale prosjektbeskrivelsen ble det foreslått å utvikle til HTC Vive headsettet. Dette er et headsett som kun fungerer som skjerm og kontroller for brukeren, og må drives fra en separat PC. Etter møte med oppdragsgiver i starten av prosjektet var det klart at de ønsket å bruke Oculus Quest headset. Da ble det enklere for studentene deres å bruke simulatoren, og de ikke trengte å låses fast til et bestemt rom og utstyr for å bruke den.

Gruppen estimerer subjektivt at vi har brukt mer enn tre uker på å tilpasse ressurser og programmering for å få simulatoren til å kjøre med ønsket bildefrekvens på Oculus Quest. Dette hadde vi sannsynligvis ikke hatt behov for i det hele tatt om simulatoren kjørte på en stasjonær datamaskin ment for VR bruk.

Det har også blitt svært redusert realisme i forhold til grafiske elementer i simulatoren grunnet den lave ytelsen i Oculus Quest. Simulatoren kjører nå uten noen form for direkte belysning, ingen overflater med refleksjon, sparsommelig bruk av store teksturer og ingen avanserte grafiske shadere. I utviklingsprosessen endte vi opp med å fjerne all geometri som ikke var

nødvendig, samt bruke samme materialer på flere geometrier for å redusere antallet tegnekall som bør begrenses til mindre enn 100 per frame for Oculus Quest.

Ved å bruke et alternativt headset ville vi sannsynligvis hatt langt over 10 ganger større ytelsesbudsjett. Dette kan vi se i forhold til hvilken bildefrekvens vi får på våre egne datamaskiner, bildefrekvens på Oculus Quest og bildefrekvens vi sannsynligvis ville hatt på en oppdatert og moderne stasjonær maskin tiltenkt VR spilling.

#### **Eksempel:**

Kjøring på en syv år gammel fire kjerners Intel i7 prosessor, 12GB RAM, med et Nvidia GTX1060 3GB skjermkort ga en bildefrekvens på rundt 1000 bilder per sekund. Oculus Quest kjøring på samme versjon av simulatoren ga bildefrekvens mindre enn 72 bilder per sekund. Gruppen estimerer at en ny moderne datamaskin ment for å kjøre VR spill sannsynligvis ville hatt bildefrekvenser rundt to-tre ganger høyere enn eksempel maskinen.

Prosjektet har hatt en løpetid på rundt 12 uker, hvor hovedfokus har vært to ulike ting; 1: Prosjektrapport og relaterte obligatoriske oppgaver i forhold til bachelorfaget. 2: Utvikling av simulatoren. Gruppemedlemmene sin kompetanse på området spillutvikling har vært lavt og enda lavere i forhold til utvikling til mobile VR-plattformer. Dette har skapt stort press på budsjett relatert til tid for gruppen, og ytelse mot funksjonalitet i simulatoren. Ved bruk av en annen VR-plattform ville det trolig løst opp veldig mye tid som kunne blitt brukt på oppdragsgivers funksjonelle krav.

I risikovurderingen vi har gjort før og under prosjektet har vi estimert risikoen for “Koden / assets er for ineffektive til å kjøre bra på Quest headsettet” til 4. Dette er trolig svært undervurdert. Konsekvensen det har hatt for prosjektet har vært veldig stor. Grunnen til at denne risikoen har blitt undervurdert er trolig sammensatt av at gruppen ikke hadde nok erfaring med spillutvikling, ytelsen til Quest og naivitet relatert til hvor vanskelig slik utvikling er.

## **7.3 utfordringer knyttet til Covid-19 pandemien**

Prosesen i prosjektet har blitt veldig annerledes fra hva skole, oppdragsgiver og gruppen hadde sett for seg. Dette har trolig ført med seg en del problemer som har gjort prosessen i prosjektet tregere.

I starten av prosjektet foreslo oppdragsgiver at vi skulle delta i en gjenopplivningsdemonstrasjon med dukke for å bedre forstå prosessen som vi skulle implementere i VR. Dette ble ikke mulig å gjennomføre etter at skolen ble stengt ned. Vår forståelse av prosessen var mye basert på diskusjoner med oppdragsgiver og scenario dokumentet de leverte. Det har vært et par misforståelser angående hva ulike uttrykk betyr og hvordan utstyr fungerer i virkeligheten. Med demonstrasjoner og tettere kontakt med oppdragsgiver ville vi nok unngått noen misforståelser og raskere fått fokus på den viktigste funksjonaliteten som måtte implementeres.

Den reduserte kontakten med oppdragsgiver førte også til at det ble vanskeligere å få til ekstern testing fra deres side. Vi følte ikke vi hadde applikasjon som kunne testes eksternt før veldig sent i prosjektfasen. Den første versjonen vi delte var den 28. april. Vi hadde et lite

klart produkt før dette, som ikke var veldig relevant å teste eksternt. Det manglet essensielle funksjoner og hadde for det meste funksjonalitet kun halvveis implementert. Dette forsinket nok forbedringer og tilbakemeldinger vi hadde trengt å få så tidlig som mulig.

I prosjektprosessen har vi som gruppe ikke møtt hverandre siden skolen ble stengt. Dette av hensyn til hverandres familier og i henhold til råd for å redusere smitte. Samarbeidet vi har hatt digitalt tidligere har vært godt, og arbeidsmåten har ikke vært ny for oss. Likevel har gruppen merket at det har vært lavere effektivitet enn normalt i arbeidet vårt. Ved usikkerhet og nybrottsarbeid har vi tidligere delt informasjon og oppklart misforståelser svært raskt, men dette har tatt mye lengre tid når vi har jobbet kun med digital kontakt. Trolig ville prosessen og sluttresultatet vært ganske annerledes om arbeidsmetodene våre ikke ble så forandret.

## **7.4 Innkjøp av 3d-ressurser, sammenlignet med å lage de selv**

Det var nødvendigvis ikke mest effektivt av gruppen å lage modeller selv med tanke på sluttresultat og læring innenfor dataingeniørfaget. Andre grupper og tidligere bachelorgrupper har fått dekket innkjøp av 3d-ressurser og modeller til bruk i prosjektene sine opp mot NOK 10.000. (Ifølge Kristoffer Nome i deres oppgave «Room of horror», Student v/HVL Data, chat over Facebook Messenger den 08.05-2020) I dette prosjektet ble det aldri drøftet budsjett for innkjøp eller andre innkjøp enn de oppdragsgiver allerede hadde gjort selv. Gruppen var heller ikke klar over at det hadde vært praksis med at oppdragsgiver setter av penger til innkjøp i andre prosjekt.

Å lage 3d-ressurser er ikke nødvendigvis veldig relatert til dataingeniørfaget, men heller mer rettet mot en kunstnerisk fagretning. I industrien i dag er det hovedsakelig fagutdannede grafikere som lager 3d-ressurser i prosjektsammenheng. Med tanke på at vi i dette prosjektet brukte opp mot  $\frac{1}{3}$  (subjektivt estimat) av tiden på å lage ressurser har det hatt en betydelig del av prosjektet. Sett i lys av funksjonalitet vi ikke fikk tid til å realisere og eventuelle forbedringer, kunne innkjøp av modeller vært hensiktsmessig.

Læringen innenfor å lage 3d-ressurser trenger en ikke se bort fra. Det er ikke sikkert at modellene vi kunne kjøpt ville vært funksjonelle for oss. Modellen av den nyfødte som ble kjøpt av oppdragsgiver måtte bearbeides for å kunne brukes. Det ble laget ny rigg og helt ny geometri.

Det er spesielle krav til optimalisering av ressurser når man arbeider mot mobile skjermkort, som f.eks. redusert flateantall, redusert antall bein med innflytelse per node i riggen, og begrensninger ved tekstur og materiale. Dermed er det noe usikkert hvilken innvirkning innkjøp av flere modeller ville ha hatt på prosjektet, men det er sikkert at vi kunne spart en del tid på enkelte mer vanlige modeller i rommet som dører, vinduer, vegger, tak, lysbryter etc.

## **7.5 Etikk i prototypen**

Som et verktøy kunne simulatoren ikke brukes i undervisning med den sist leverte prototypen. Dette på grunn av at den ikke er faglig korrekt nok, og at det er fare for at dette kan skape

uønskede vaner hos studentene som bruker den. Det er riktignok kun små justeringer som skal til før oppdragsgiver føler den ville vært et godt verktøy.

Med tanke på trygghet i simulatoren virker det som om den oppnår dette godt. Det er ingen tidsfrist for å bli ferdig med arbeidsoppgavene i scenarioet og oppdragsgiver opplever det som en god løsning. De mente det ikke vil være fare for at studenter kan bli redd eller negativt påvirket av den.

## 7.6 utfordringer med Unity VR og Oculus VR pakke

Under utvikling oppsto det mange utfordringer knyttet mot VR. Som nevnt tidligere brukte gruppen Steam VR for å raskt kunne teste builds, uten lange byggeperioder. Steam VR bruker Open VR grensesnittet for å snakke mot Unity, men dette ble fjernet i en nyere versjon. Støtte og oppdateringer mot Open VR sluttet i Unity 2018.x, og fjernet helt i Unity 2019.5.

Prosjektet ble satt opp med 2019.3. De nyere versjonene av Unity bruker et nytt system for å håndtere både AR og VR, men det er ingen oppdatering for støtte mot Open VR. Uten å vite dette oppdaterte vi til 2019.5, da mye søking på nettet sa at det var bedre ytelse og kortere byggetider. Dette viste seg å være en ekstremt dårlig ide, og tok mye frustrasjon og feilsøking å finne ut hva som faktisk gikk gale. Det var her Unity begynte å oppføre seg ekstremt merkelig. Valget for å bruke Open VR var av og til synlig i menyer, men det hadde ingen effekt å huke det av. Om man åpnet eller byttet faner i innstillingene så ville valget huke seg av uten at noen rørte det. Det var også flere problemer med denne versjonen for gruppen og prosjektet ble nedgradert til den tidligere 2019.3 som hadde vist seg å fungere.

Oculus har en ferdiglaget pakke for å enkelt kunne lage VR spill med Unity. I teorien skal man bare laste denne ned fra Unity Asset Store og importere. I realiteten er mye av koden og eksemplene her ikke robust nok implementert eller har dårlig ytelse. Pakken er presentert som et utviklerverktøy, men store deler av den mangler riktig dokumentasjon og vedlikehold. Mange av problemene var fikset enten ved å skrive om store deler av pakken selv, eller finne løsninger på Unity forumet som var over et år gamle. På lavere nivåer viste denne pakken seg å være grei å jobbe med, men førsteinntrykket på "eksemplene" som fulgte med var ikke bra. For eksempel er det en mulighet å si hvor hånden skal gripe et objekt når du plukker den opp. Koden for å regne ut forskyving for å visuelt gripe på riktig sted var regnet ut feil. Korrekt implementasjon ble funnet på Unity forumet. Denne var laget for en mye eldre versjon, som tyder på at problemet ikke har blitt fikset siden da.

Mye av VR dokumentasjonen til både Unity og Oculus viste seg å være feil, ikke utfylt nok eller baserer seg på bruk av annen teknologi. Løsningen på mange av disse problemene var dessverre å håpe at andre brukere hadde samme problemene og laget en løsning.

## 8 Konklusjon og videre arbeid

### 8.1 Måloppnåelse og konklusjon

Målet med prosjektet var å lage en VR-simulator som skulle inneholde et scenario for gjenopplivning av en nyfødt. Formålet med simulatoren er å gi jordmorstudentene et alternativ til dukke trening for å øve på NRR sin gjenopplivning av nyfødte.

Ut ifra de initelle kravene og evalueringsresultatet har prosjektet stort sett oppnådd målene sine, men prosjektet trenger enda mer arbeid for å kunne brukes til det originale tenkte formålet. Å levere et ferdig og klar- til bruk produkt ville vært utenfor rammene i prosjektet. Som en utforskning av problemet og produksjon av en prototype har prosjektet vært suksessfullt.

Prosjektet har hatt en uvanlig prosess grunnet Covid-19 pandemien og har dermed avviket fra ønsket prosess. Tidsbruk og kommunikasjon mellom oppdragsgiver og gruppen, samt innad i gruppen kunne vært bedre, men alle parter har en felles enighet om dette og forståelse for den endrede prosessen.

Det har vært en spennende og erfaringsrik prosess på et prosjekt som sannsynligvis har god samfunnsnytte. Trolig er prosjektet en utmerket plattform å bygge videre på for en enda bedre og mer faglig korrekt simulator.

### 8.2 Utelatt funksjonalitet

Tørking og stimulering av barnet er ett av de første og viktigste punktene i NRR algoritmen. VR systemer i dag (2020) tilbyr som regel ikke noen form for sensorisk tilbakemelding til brukeren annet enn syn og lyd. Hvis brukeren ikke kan kjenne vekt, motstand eller berøring fra systemet er det begrenset hvilke handlinger de kan foreta seg og fortsatt opprettholde en naturlig realisme. I simulatoren er det svært begrenset realisme når det gjelder å løfte barnet eller å berøre det. "Ragdoll"-fysikken oppfører seg ikke spesielt bra hvis brukeren gjør brå bevegelser. Vi har implementert en forenklet form for stimulering hvor studentene må holde kontrollene ved siden av brystkassen til barnet og stryke opp og ned, mens en knapp holdes inne.

Ventilator og pulsoksymeteret er i realiteten koblet til asfyksibordet. Å lage en oksygenlange for ventilatoren som ser realistisk ut er ikke enkelt, og er krever veldig mye ytelse dersom den skal oppføre seg riktig. Det finnes allerede eksisterende pakker i Unity Asset Store [\[17\]](#) som gjør det enkelt, dette ble ikke aktuelt da gruppen ikke hadde penger tilgjengelig og heller ikke ønsket å bruke egne penger på dette. Prosjektet hadde allerede ytelse problemer, og det er ikke helt sikkert om disse slangene hadde kjørt bra med den svakere maskinvaren i Oculus Quest.

I simulatoren blir de fleste parametere knyttet til barnets helse og kropp ikke simulert korrekt eller i det hele tatt. Måten det er programmert er statisk, det settes parametere og ved visse interaksjoner vil de settes til en ny verdi. En bedre og mer generell løsning kunne vært å faktisk lage en modell som simulerer et barn med kroppstemperatur, puls, oksygenprosent i blodet, respirasjonsrate, hudfarge, bevegelse, reflekser osv. Alle disse parameterne kunne hatt



realistisk påvirkning på hverandre sømløst. Gitt at studentene stimulerte barnet på en spesifikk måte, ville dette skapt en kaskadeeffekt med realistiske reaksjoner fra barnet. Grunnen til at vi valgte den statiske implementasjonen er at vi har begrenset med kunnskap om hvordan menneskekroppen fungerer, dermed ville det vært vanskelig å lage modellen korrekt. Trolig ville vi ikke hatt tid til å lage en god implementasjon av dette.

### **8.3 Videre arbeid**

Fra prosjektstart var oppdragsgiver klar på at dette prosjektet muligens var større --enn et bachelorprosjekt. Hovedsakelig gikk prosjektet ut på å lage en prototype som andre kan jobbe videre på. Dette målet er blitt oppnådd. Oppdragsgiver er fornøyd og er positiv til å arbeide videre med spillet.

Mye av utviklingen har gått til å få til funksjonalitet i simulasjonen. Ettersom mye av funksjonaliteten allerede er på plass, gjenstår det ikke like mange utfordringer ved å kunne implementere flere scenarioer slik at studenter kan øve på flere ulike situasjoner. Det må regnes med noe refaktorering av koden med tanke på kodestruktur. Dette er grunnet kort tid mot slutten hvor funksjonalitet ble prioritert over kodekvalitet. I tillegg kan det være hensiktsmessig å endre implementasjoner av noen kontrollere for å gjøre de mer generelle.

# 9 Referanser

## 9.1 Kildeliste

1. Norsk Resuscitasjonsråd (2015) *Retningslinjer 2015 Resuscitering av nyfødte*. Tilgjengelig fra: <https://nrr.org/images/pdf/Nyfodte.pdf> .
2. Unity (2020) *Unity for all*. Tilgjengelig fra: <https://unity.com>.
3. Oculus (2020) *Oculus Quest*. Tilgjengelig fra: <https://www.oculus.com/quest>.
4. Store norske leksikon (2018) *apgarscore* Tilgjengelig fra: <https://sml.snl.no/apgarscore>.
5. Unity (2020) *About Unity Test Framework*. Tilgjengelig fra: <https://docs.unity3d.com/Packages/com.unity.test-framework@1.1/manual/index.htm>.
6. NUnit (2020) *What is NUnit*. Tilgjengelig fra: <https://nunit.org/>.
7. Autodesk Knowledge Network (2020) *Programvare for 3D-modellering og gjengivelse for designvisualisering, spill og animasjon*. Tilgjengelig fra: <https://www.autodesk.no/products/3ds-max/subscribe?plc=3DSMAX&term=1-YEAR&support=ADVANCED&quantity=1>.
8. Autodesk Knowledge Network (2020) *Polygonal Modeling*. Tilgjengelig fra: <https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/GUID-7941F97A-36E8-47FE-95D1-71412A3B3017-htm.html>.
9. Autodesk Knowledge Network (2020) *Mapping UVs*. Tilgjengelig fra: <https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/GUID-FDCD0C68-2496-4405-A785-3AA93E9A3B25-htm.html>.
10. Concept Art Empire (2020) *What is 3D Rigging For Animation & Character Design?*. Tilgjengelig fra: <https://conceptartempire.com/what-is-rigging/>.
11. Steam (2020) *SteamVR*. Tilgjengelig fra: <https://store.steampowered.com/steamvr>.
12. Android (2020) *Android Debug Bridge (adb)*. Tilgjengelig fra: <https://developer.android.com/studio/command-line/adb>.
13. Høgskulen på Vestlandet (2020) *Institutt for helse- og omsorgsvitenskap*. Tilgjengelig fra: <https://www.hvl.no/om/organisering/fhs/iho/>.
14. Agile Alliance (2020) *Kanban*. Tilgjengelig fra: <https://www.agilealliance.org/glossary/kanban/>.
15. <https://nrr.org/images/nedlasting/pdf/nyfodt3.png> NRR plakat
16. Boldingh A.M, Solevåg A.L, Nakstad B. (2018) *Outcomes following neonatal cardiopulmonary resuscitation*. Tilgjengelig fra: <https://tidsskriftet.no/en/2018/05/oversiktsartikkel/outcomes-following-neonatal-cardiopulmonary-resuscitation>. (Hentet 10. April 2020)
17. Unity (2020) *Unity Asset Store*. Tilgjengelig fra: <https://assetstore.unity.com/>.

18. Unity (2020) *Skinned Mesh Render*. Tilgjengelig fra:  
<https://docs.unity3d.com/Manual/class-SkinnedMeshRenderer.html> Unity Manual,  
Skinned Mesh Renderer [Hentet 01.06.2020]

## 9.2 Figurliste

Figur 1: Medisinsk asfyksibord, Oculus Quest .....	14
Figur 2: Klokke på veggen og stoppeklokke i simulatoren .....	21
Figur 3: Varmelampe i simulatoren .....	22
Figur 4: Kontrollpanel asfyksibord i simulatoren.....	22
Figur 5: Nyfødt barn i simulatoren .....	23
Figur 6: Barnemodellen i 3ds Max med rig .....	23
Figur 7: Laerdalsbag illustrasjon .....	24
Figur 8: Laerdalsbag i simulatoren.....	24
Figur 9: UV utbretting av en kube .....	25
Figur 10: Rommet i simulatoren i en tidlig versjon .....	26
Figur 11: Babymodellen i 3ds Max, før og etter retopologi .....	26
Figur 12: Babymodellen i 3ds Max med og uten smooth shading .....	27
Figur 13: Fullført scenario i simulatoren.....	28

Figur 7 hentet den 02.06.2020 fra:  
[https://lvh.no/naar\\_det\\_haster/praktiske\\_ferdigheter/oksygenbehandling/oksygenbehandling](https://lvh.no/naar_det_haster/praktiske_ferdigheter/oksygenbehandling/oksygenbehandling)

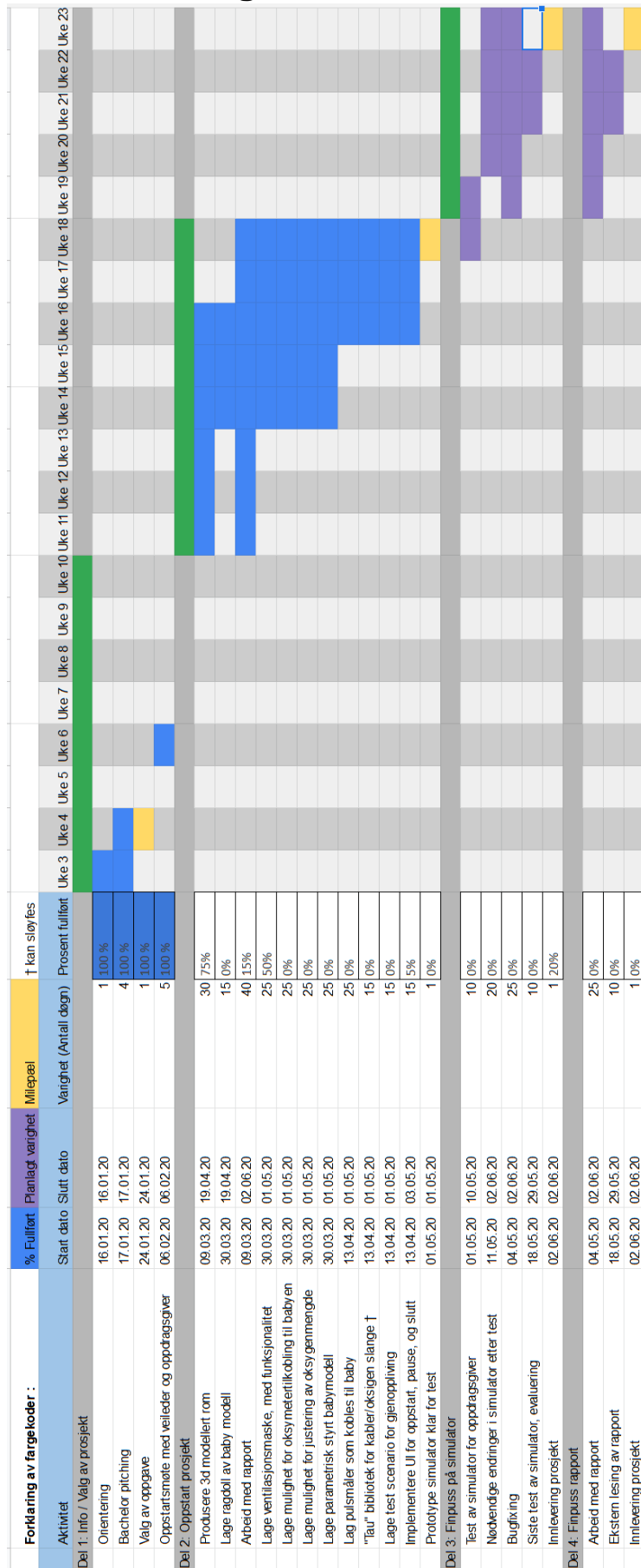
Figur 9 hentet den 02.06.2020 fra:  
[https://commons.wikimedia.org/wiki/File:Cube\\_Representative\\_UV\\_Unwrapping.png](https://commons.wikimedia.org/wiki/File:Cube_Representative_UV_Unwrapping.png)

# 10 Vedlegg

## 10.1 Risikoliste

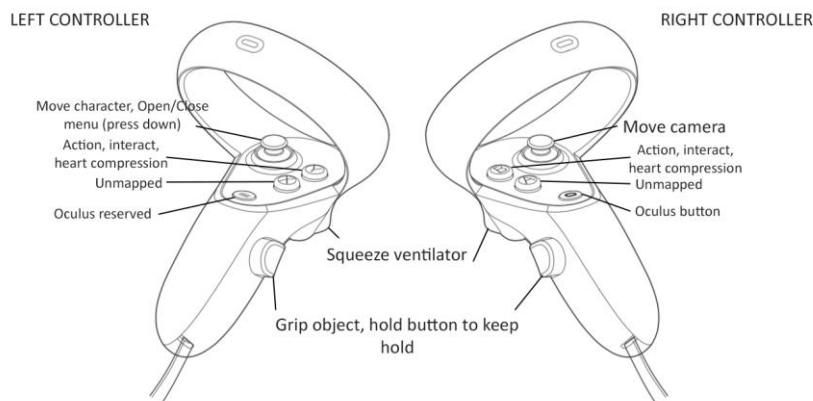
1. Skole og arbeidsplasser blir stengt
2. Unity er betraktelig vanskeligere å arbeide med en forutsatt
3. Vanskelig å få lagd nødvendige modeller og materialer til simulatoren
4. Koden / assets er for ineffektive til å kjøre bra på Oculus Quest headsettet
5. Animasjon kan være veldig vanskelig
6. Kode kvalitet for gjenbrukbarhet og lett utvidelse
7. Mangler nødvendige 3d modeller
8. Alle mister internett
9. Noen/Alle blir syke

## 10.2 Gantt diagram



## 10.3 Brukermanual

**Bruksanvisning:** VR-simulator for trening til korrekt gjenopplivning av nyfødte



**Disse tingene må gjøres for at scenarioet skal bli godkjent og ferdig. Disse elementene må gjøres i rekkefølge.**

1. Start stoppeklokke
2. Ta kjevegrep
3. Ventil minimum 30 ganger med rate mellom 30 og 60 per minutt, det kan ikke være opphold på mer enn 10 sekunder på ventilasjoner
4. Gjør hjertekompresjoner 90 ganger, med rate mellom 80-100, det kan ikke være opphold på mer enn 10 sekunder på kompresjonene. Man kan anta at barnet blir ventilert tilstrekkelig mens dette gjøres.

**Kan gjøres, men er ikke krevd for å fullføre scenarioet.**

- Start varmelampe, kan gjøres når som helst.
- Feste pulsoksymeter på fotbladet til barnet, kan gjøres når som helst.
- Juster tilført oksygen i maske til 100%, kan gjøres når som helst.

Når scenarioet er ferdig, vil du få opp sluttrapport ved siden av asfyksibordet. Hvis du sitter fast kan du ta opp menyen for å få opp sluttrapporten underveis i spillet, der kan du se hva spillet har registrert at du har gjort. (Det kan være bugs relatert til dette.)

### **Bruksanvisning:**

Scenarioet startes automatisk, men kan også startes fra menyen.

Menyen kan åpnes ved å trykke ned den venstre joysticken. En laserpeker vil være synlig når bruken sikter mot menyen.

Scenario kan velges med rullegardinmenyen, enten ved å peke med laserpekeren eller med venstre joystick.

Du kan bevege karakteren med venstre joystick og kameraet med høyre joystick, slik at brukeren ikke trenger å gå langt i virkeligheten.

Som oftest interagerer brukeren med den øverste knappen på den korresponderende kontrolleren, men dette kan variere ut ifra hvilken handling som gjøres. Det er tooltips

tilgjengelig ved siden av kontrollen når en handling kan utføres, som opplyser om hvilken knapp som er relevant for aksjonen. Det er også en plakat i rommet til høyre som forklarer knappene.

Objekter kan plukkes opp med håndtriggeren, tooltips vil være synlig når noe kan plukkes opp. Ikke alle objekter og knapper i simulatoren kan gjøres interaksjoner med, så langt er det ikke en tydelig måte å se dette, men ta hensyn til bruksanvisningen.

#### **Hvordan bruke de ulike funksjonene:**

- Gripe, flytte barnet, sjekke tonus: Før kontrolleren til lemmet du vil gripe, trykk gripe knappen og hold denne så lenge du ønsker å gripe lemmet. Du kan flytte kontrolleren og slippe for å sjekke refleks og tonus i lemmet. Ikke alle deler av barnet kan gripes eller flyttes. Sjekk tonus ved å løfte armer, eller bein.
- Stimulere barnet: Før kontrollene på siden av brystet til barnet, hold knappen som tooltip sier (Action knappen), mens denne holdes inne, stryk opp og ned langs barnet minimum 3 ganger.
- Starte stoppeklokke: Ta hånden på knappen ved stoppeklokken, trykk på knappen som tooltip beskriver, (Hånd trigger knapp, gripe objekt knappen)
- Ta kjevegrep: Ta hånden i på babyen sitt ansikt, trykk på knappen tooltip beskriver. Du trenger ikke holde kjevegripen kontinuerlig. (Action knappen på kontrolleren du holder med, Y eller B.)
- Slå på varmelampe: Ta hånden på knappen ved knappen for varmelampe, trykk på knappen som tooltip beskriver, (Hånd trigger knapp, gripe objekt knappen)
- Juster oksygen % i maske: Ta hånden på skruknappen for oksygen, trykk på knappen som tooltip beskriver og vri til ønsket prosentmengde. Displayet for oksygenprosent vil vise innstilt prosentmengde også. (Hånd trigger knapp, gripe objekt knappen)
- Feste pulsoksymeter: Før hånden til pulsoksymeteret, grip dette med gripeknappen, før dette mens du holder det bort til en av fotbladene til barnet, bruk Action knappen (Y eller B) for å feste det til foten. Du vil nå få opp pulsen til barnet på asfyksbordet.
- Ventil med maske: Ta hånden på ventilatorbaggen, grip den med gripeknappen, denne må holdes inne for å ikke slippe den. Før ventilatormasken nært babyen sitt ansikt, trykk på knappen som tooltip beskriver for å feste masken til babyen sitt ansikt. (Action knappen på kontrolleren du holder med, Y eller B)

For å komprimere baggen trykker du på index trigger knappen, ventilatoren komprimeres i henhold til knappens bevegelse. Sørg for å se at brystet til barnet løfter seg når du komprimerer.

Hvis brystet på barnet ikke løfter seg, er masken ikke festet. Masken kan rives av hvis du drar ventilatoren bort fra barnet.

- Gjøre hjertekompresjoner: Før begge hendene til brystkassen til barnet, les tooltip, trykk Action knappen på begge kontrollene for å komprimere brystet til barnet. Disse må rytmisk trykkes og slippes for å komprimere korrekt. Se at brystet senkes raskt når man komprimerer.

## 10.4 Ordliste

- **Albedo**

“Albedo, mål for refleksjonsevne. Albedo er et uttrykk for hvilken evne flater har til å reflektere lys, og blir definert som den brøkdelen av parallelt innfallende lys som blir spredt diffust. En albedo på 0,10 vil si at 10 % av innfallende lys blir kastet tilbake.” - <https://snl.no/albedo>
- **Apgar-score:**

“Apgarscore er et system for rask vurdering av allmenntilstanden hos nyfødte. 0, 1 eller 2 poeng gis for hjertefunksjon, respirasjon, muskeltonus, reflekser og hudfarge, og summeres til en score som maksimalt kan være 10 poeng. Vurderingen gjøres normalt to ganger; én ett minutt og én fem minutter etter fødselen.” - <https://sml.snl.no/apgarscore>
- **Dummy-objekt**

Et objekt som ikke vises i spillet, men brukes for å ha kjørende komponenter.
- **Frame**

En ny bildetegning på displayet.
- **Mesh**

En sammenhengende overflate av polygoner
- **Muskeltonus:**

“Muskeltonus er den normale (ubevisste) spenningen i en hvilende muskel; kan være spenning både i skjelettmuskulatur og glatt muskulatur, for eksempel i åreveggene.” - <https://sml.snl.no/muskeltonus>
- **Node**

Et punkt med en plassering i 3 dimensjoner. Brukes for å definere linjer, en linje er mellom to punkter. Vertex på engelsk.
- **Polygon:**

En to-sidig flate definert av minimum 3 noder, opp til n-noder. Vanligvis fire eller trekantet.
- **Resuscitasjon:**

“Resuscitering er et medisinsk begrep som i hovedsak brukes for å betegne gjenopplivingsforsøk med hjerte-lunge-redning av barn og voksne med hjertestans.” - <https://sml.snl.no/resuscitering>
- **Skinning**

Prosessen med å “feste” et mesh som et hudlag over et skjellet (en rigg). Gjøres ved å vekte innflytelsen av et beins plasing og rotasjon på hver enkelt node i meshen. Dette gjøres for å abstrahere animasjon bort fra å måtte animere nodene i seg selv, til at man animerer skjelettets bevegelse, og meshen og nodene følger dette.
- **Tooltip**

En tekstboks med ekstra informasjon til brukeren som forklarer en handling eller interaksjon
- **UV mapping**

En plassering av polygonene (flatene) i en mesh i et plan av to akser, U og V. Disse går numerisk fra 0 til 1. Plasseringen gjør det mulig å beregne nøyaktig hvilken farge en piksel skal ha på skjermen, ved å beregne hvor den interpolerte teksturen skal vises på flaten til meshen, og hvilken farge punktet pikselen skal vise har i teksturen.



## 10.5 Scenario foreslått av Katrine Aasekjær

### Nyfødtresuscitering – algoritme

Du har nettopp tatt imot et barn født til termin. Barnet er forløst med vakuumpå grunn av protrahert forløp stadium II. Ved fødsel har ikke barnet egen respirasjon, barnet er bleikt og nedsatt tonus

Hva skal du gjøre?

1. Sjekk nødvendig utstyr (gjærne gjennomført på forhånd)
2. start tiden (viktig at de må starte tiden for dette er avgjørende for resten av algoritmen og når du skal gjøre hva)
3. Sørg for frie luftveier (kjevegrep), tørk og stimuler barnet, hold barnet varmt (det skal i prinsippet ikke ta mer enn 10 sekunder å få en oversikt over situasjonen)

Her kan det være aktuelt å tenke at varmelampe må slås på – eventuelt forsikre om at den er på, skifte ut våte håndklær

4. Tiltak skal vurderes. Hva er barnets respirasjon – puster/puster ikke barnet? Gisper det? Hva er barnets hjerterefrekvens? Er den over eller under 100 slag/minutt. Kommuniser barnets hjerterefrekvens. Hvordan er barnets tonus – er barnet slapt/hypotont/bevegelse? Hvilken hudfarge har barnet? Rosa/blek/grå/hvit/grønn av mekonium (barnets avføring)

Svar på observasjoner i punkt 3:

Barnet har ingen egenrespirasjon (puster ikke), det er slapt/hypotont, hudfargen er blek og hjerterefrekvensen er 50 (langsom)

Tiltak som skal iverksettes

Fortsatt sikre frie luftveier (kjevegrep). Ventilasjon med maske/bag – det skal da være 30 til 60 innblåsing per minutt (pågår i 60 sekunder). Her må thorax (brystet til barnet) heve seg – dette skal observeres ved at brystet beveger seg – indikator på at barnet får i seg luft. Om brystet ikke hever seg må studenten undersøke om masken er plassert korrekt over munn og nese, for deretter og sikre korrekt kjevegrep. Dersom det er flere til stede i situasjonen – vurder om det skal kobles til et pulsoksymeter – dette kan si noe om barnets oksygenmetning (dersom vi kobler til O<sub>2</sub> metning kan vi vurdere å legge inn mengde oksygen som skal strømme på masken – vanligvis starter vi med romluft, for deretter å øke prosentvis O<sub>2</sub> dersom vi ser at barnet ikke responderer på tiltak som iverksettes – dette kan vi diskutere, så skal jeg undersøke hvor mye O<sub>2</sub> som tilføres av gangen og hvor raskt dette skal gjøres)

Ventilering av barnet skal pågå i 60 sekunder

Ny vurdering (ved denne vurderingen er hjerterefrekvensen 50, fortsatt ingen egenrespirasjon, blek i huden, ingen tonus)

5. Dersom hjerterefrekvensen etter 60 sekunder fortsatt er under 100 iverksettes følgende tiltak
  - Forsikring av frie luftveier. Kontroller hodeposisjon, sjekk for maskelekkasje

- Øk eventuelt inspirasjonstrykket (det trykket du bruker for å blåse luft ned i lungene)

6. Ventiler i ytterligere 30-60 sekunder – deretter sjekkes hjerterefrekvens på nytt.

Hjerterefrekvensen er fortsatt under 60 – og de skal da starte med hjerterekompresjoner (HRL) 3:1. Her skal man tilstrebe 90 kompresjoner og 30 innblåsing per minutt. Økt O<sub>2</sub> konsentrasjonen på maske til 100%

7. Hjerterefrekvens skal nå revurderes hvert 30-60 sekund

- Hjerterefrekvensen er nå over 100 – kompresjoner avsluttes,
- Barnet starter med egenrespirasjon – grynter/presser/gråter – opprettholder gjerne O<sub>2</sub> på maske ( dette kan vi diskutere)

## 10.6 Evaluering av første prototype fra oppdragsgiver 20.05.2020

### Evaluering av VR-resuscitering av nyfødte

Hei

Vi har nå tatt en grundig gjennomgang og har testet det grundig begge to, både hver for oss og sammen. Vi har prøvd å gå så godt inn i detalj som mulig og håper dere finner tilbakemeldingene våre nyttige til innspurten av prosjektet.

Det er bra at de fleste parameterne i algoritmen for gjenopplivning finnes i scenarioet. Tilbakemeldingen nedenfor vil være innspill som krever justeringer for at det blir mer realistisk og brukervennlig. Vi er klar over at vi må kompromisse, men nå er det for stor avstand mellom hva som er reelt og hva som skjer i VR.

### Innspill:

Menyen og instruksjonene til hvordan bruke VR rommet er ikke intuitiv for en bruker som ikke er kjent med VR. Det står en del feil i instruksjonen på hvordan man skal bruke knappene. Det står ingen steder i instruksjonen at knappene skal holdes inne for å opprettholde funksjonen som skal gjennomføres. Dette gjør at man som uerfaren i VR mister for eksempel et kjevegrep når dette er gjennomført. Dere må ha større fokus på hvem brukeren er.

Instruksjonen til hvordan man skal bruke de ulike knappene på kontrollerne må forklares mer inngående i starten av instruksjonen inne i VR miljøet. Dette gjelder for hvordan man tar et kjevegrep, hvordan man sjekker tonus, hvordan man kontrollerer for hjerteaksjon. Om ikke dette forklares, kan det se ut som at spillet ikke har disse funksjonene. Husk at brukerne av spillet med stor sannsynlighet har liten eller ingen erfaring med VR fra tidligere. Hudfarge er vi usikker på om dere har fått med.

Scenariotittel må endres/fjernes

Start og stopp av stoppeklokke er vanskelig. Her står det feil i bruksanvisning og etikett hvilken knapp man skal bruke

Det står i bruksanvisning hvordan man skal skru på O2, men det er veldig unaturlig da det ikke er koblet noen ting til O2 apparatet (eksempel slange)

Når man kommer inn i rommet med barnet på asfyxibordet ligger Laerdalsbag og SaO2 på et bord ved siden av. Hvorfor det? Videre har ikke Laerdalsbagen koblet en O2 slange til seg. Hvorfor ikke? For å kunne ventilere et barn må det være koblet til oksygen. Bag vil ligge på asfyxibordet. Husk at bag ikke er dimensjonert slik som dere har laget den (den er for stor). Jeg vet vi snakket om å ha et bord til utstyr, men da mente jeg det utstyret som ligger i selve bordet. Bag ligger alltid på bordet.

Skuffen i bordet er åpen – hvorfor? Om den skal være åpen bør det da ligge utstyr i denne.

Asfyxibordet er strippet ned og mangler veldig mange av de funksjonene som et reelt asfyxibord skal inneholde. Hvorfor det? Tanken er jo at det skal være så autentisk som mulig, og da er det viktig å gjøre seg kjent med utstyret og det som det inneholder. Startknapp er veldig stor, den er jo ikke det i det virkelige liv.

Når man holder kjevegrep og skal få plassert bag korrekt over munn og svelg, er dette veldig vanskelig å gjennomføre. Vanskelig på en slik måte at man som spiller flytter fokus fra hva man skal gjøre for å redde babyen og over på hvordan funker det tekniske i dette spillet. Dette henger sammen med det som jeg sa innledningsvis – det står ikke forklart og da vet

man ikke når man kommer dit i spillet. Husk at man ikke kan se at man gjennomfører kompresjoner – det kan være lurt at man har en indikator på at man gjør ting korrekt. Både kompresjoner, maske og kjevegrep. Vi snakket om at det skulle kunne komme innspill i spillet, men nå er jo alt fastsatt og kan dermed ikke korrigeres. Dette gjelder for kjevegrep og det gjelder for hjerteaksjon. Det kommer aldri en endring i hjerteaksjonen så langt som vi klarer å gjennomføre scenarioet. Det betyr at denne babyen ikke responderer på noen av de tingene vi iverksetter.

Du spiller et scenario i blinde, men du får ingen innspill gjennom spillet. Dere sier jo at vi får det, men vi ser ikke dette for vi klarer ikke å fullføre scenarioet. Det er neo som vi gjør feil, men vi klarer ikke å finne ut hva

Det er lite i spillet som tilsier at man skal tenke på ting som stimulering og tørking av barnet. Vi diskuterte jo mulighetene for dette på sist møte, og at man kan gjennomføre dette ved en form for nedtrekks-gardin med valg om at man vil gjøre i løpet av de første sekundene man treffer barnet. Ved at disse tingene mangler helt blir det veldig lite autentisk

OBS at det er lett å komme borti knappene som styrer bevegelse (joystick), dette gjør at du plutselig befinner deg på et annet sted i rommet. Ting hopper. Dette kan løses ved å plassere brukeren rett foran bordet, og fjerne det ekstra-bordet, og da er det ikke behov for å vandre rundt i rommet

Barnet har et veldig unaturlig bevegelsesmønster – vi har tidligere diskutert filledukkeeffekten. Denne bør bare være på når man sjekker tonus, eller når man skal stimulere barnet.

I den nye versjonen ser det ut som det er en modell utenfor rommet, synlig gjennom et hull i taklampene – ta dette bort?

### **Prosessevaluering:**

I prosjektperioden har vi som oppdragsgivere i liten grad blitt involvert i utviklingen og fremdriften i prosjektet. Vi er klar over at koronapandemien satt en del begrensinger, men vi tror dere bør, for fremtidige prosjekter, være oppmerksom på at det kan være hensiktsmessig å arbeide tettere opp mot oppdragsgiver. Vi vil gjerne ha et møte for å se hvilke justeringer/endringer som er mulig å gjennomføre før en innlevering av prosjektet.

### **Oppsummert:**

Et bra utkast til videre arbeid. Men det fremstår som mye mer forenklet enn det vi hadde trodd og håpet på. Det er ikke helt brukervennlig.

Ivar og Katrine

## 10.7 Evaluering fra oppdragsgiver av andre prototype den 27.05.2020

Testing av VR resuscitering av nyfødte den 27.05.2020

Utgaven er en mye bedre utgave enn det vi har blitt presentert for tidligere i prosessen. Mye av våre innspill er tatt hensyn til.

Asfyxibordet er satt opp korrekt og virker dimensjonert i forhold til det som er bestillingen.

Start av stoppeklokke – noe forvirrende fordi man må putte fingeren inne i selve knappen, det holder ikke å bare touche (kanskje vurdere en større kollisjonsboks?)

Bruksanvisningen er noe feil i forhold til algoritmen fra NRR – det første man skal gjøre er å få oversikt ved å sjekke hjerterefrekvens, respirasjon, stimulere og sikre frie luftveier. Deretter gjør man kjevegrep. Dette kan løses ved å lage flere scenarier – type lett, vanskelig, realistisk. Kan be spilleren om å svare på hva som er korrekt å gjøre. Dette burde vi ha sett tidligere i testingen av spillet og ser at det ikke er helt logisk for utvikler.

Stimulering – dette står ikke i bruksanvisning. Hvordan dette skal utføres må være litt mer tydelig, men veldig flott at dette er mulig å gjøre. Litt vanskelig å treffe den sonen der det er mulig å gjennomføre stimuleringen

O2 slangen er korrekt plassert, men er for stor dimensjonert. Denne kan godt ligge parallelt med barnet på bordet. Kobling mellom slange og bag er korrekt. Tenk på at de fleste er høyrehendte – kunne slange og bag bli lagt slik at det ligger langs barnets høyre side? «bug» i bag – når man griper bag blir denne liggende på spillerens håndledd - du får lett bagge, men du blir litt forvirret i starten fordi baggen ligger på håndleddet

Kjevegrep – dette er enkelt å plassere, men det «glir» noe tilbake etter hvert. Det innebærer at barnet legger haken mer og mer ned mot brystet igjen (kan dette justeres?). Enkelt å ta opp bag, men her er det noe problematisk å få fiksert munnstykket slik at det ligger over nese og munn. Når man får til å feste masken er det enkelt å se at brystkassen hever seg. Det er ikke samspill mellom innblåsing og hjerterefrekvens, men hudfargen endres etter 30 sekunders innblåsing. Tenk at sirkulering av blod (hjerterefrekvens) har betydning for hudfarge. Dette burde vi ha presisert tidligere, kan løses ved at hudfargen ikke endrer seg så raskt eller ved at hudfargen endres i takt med at hjerterefrekvensen går opp

Vi tror at utfordringene knyttet til kjevegrep og maskeplassering kan løses ved å slå av filledukke funksjonen eventuelt justere denne.

Mange av funksjonene i spillet er bra og kan være gode for en videreutvikling av spillet. Noe av algoritmen til NRR følges ikke – det er veldig bra at man kan justere på O2, men se på algoritmen hva den sier slik at det blir samsvar mellom tilbakemelding etter endt spill og det som NRR sier. Dersom det ikke er samsvar mellom NRR og spillets fasit vil jo studenten som kan algoritmen aldri få 100 % score. Se tilbake på den filen som vi ga til dere 10.mars. Her har dere en detaljert beskrivelse av hvordan dere kan bygge opp et scenario i spillet. Vi ser at oppskriften dere fikk av oss ikke er fulgt, men samtidig ser vi at dere er veldig nærme. Noe som betyr at det ikke nødvendigvis er veldig mye som skal til for at det blir større samsvar mellom bestilling og produkt. Dersom disse tingene løses har man et produkt som kan brukes i undervisning, noe som er utrolig bra.

Ved komprimering er det samspill mellom kompresjoner og respons i hjerteaksjonen. Litt vanskelig å se om det er tatt hensyn til algoritmen fra NRR med tanke på 3 kompresjoner og en innblåsing som et intervall. Dette diskuterte vi, og at en virtuell kollega kan overta en av funksjonene – dette kan løses ved en tekst som kan informere om hva som skal skje (tenk at noe av det første som står i NRR sin algoritme er å tilkalle hjelp)

Når scenariet skal re-startes, så restarter ikke den interne klokken i programmet. Klokken begynner da å telle fra det forrige scenarioets slutt.

**Oppsummert:**

Mye bra, og et langt bedre produkt enn tidligere presentert og testet av oss. Noen justering vil heve spillets funksjonalitet og brukeropplevelse. Vi som bestillere er fornøyde, og har fått et produkt som vi kan arbeide videre med.

Bergen 27.05.2020

Katrine Aasekjær og Ivar Rosenberg

## 10.8 Original prosjektbeskrivelse før valg av prosjekt

### P1 -Spillutvikling –Resuscitering (gjenoppliving) av nyfødte (Master i jordmorfag, HVL)

#### Bakgrunn for prosjektet

Resuscitering av nyfødte er en situasjon som studentene sjeldent eller aldri får erfare i klinisk praksis. Til tross for dette er det forventet at en nyutdannet jordmor har både kompetanse og ferdigheter knyttet til resuscitering av nyfødte barn. En slik kritisk situasjon kan oppstå uventet og det er derfor nødvendig at jordmødre og annet helsepersonell er godt forberedt og har kompetanse i bruk av de nødvendige prosedyrer som kreves i en slik situasjon.

Gjennom SIM-Arena og de ressursene vi har der kan studentene simulere på denne situasjonen, og gjennom simulering trene på den aktuelle algoritmen for gjenoppliving av nyfødte barn. Utfordringen knyttet til simuleringstreningen er at studenten sjelden utfordres på å ta en beslutning i tillegg til å øve på de tekniske ferdighetene.

Vi planlegger derfor å lage et VR-spill basert på algoritmen fra Norsk Resusciteringsråd, med et variert antall av pasientscenarier. Når stress og tidspress er en faktor er det viktig å ha tilstrekkelig trening i prosedyrer og fremgangsmåter, slik at man tar den rette beslutningen knyttet til relevant tiltak -et slikt spill vil kunne bidra i å sikre studentenes kompetanse og trygghet, samt muligheten til å trene uavhengig av lærer, tid og sted.

#### Arbeidsoppgaver

Utvikle beskrevet læringsspill i Unity. Primært for VR (HTC Vive, men muligheter for andre plattformer bør tas hensyn til.

Studenten får presentert en nyfødt som i en eller annen grad har behov for livreddende førstehjelp. Informasjonen er begrenset til respirasjon, hjerterefrekvens, tonus,refleks og hudfarge (apgar score).

Studenten ser babyen liggende foran seg på skjermen og hun/han må ved hjelp av audiovisuell input tolke behov for livreddende tiltak, og ut fra den informasjonen vurdere hvilke tiltak som skal iverksettes. Studenten kan velge mellom tiltak som stimulering, frie luftveier, sug, innblåsing/ventilering, hjertekompresjoner, tilførsel av iv væske, blod osv. Studenten må utføre hjerte lungeredning der han/hun blant annet skal sørge for frie luftveier, sikre hodeleie og iverksette innblåsing eller kompresjoner i riktig takt og med korrekt antall repetisjoner.

Tilstanden til pasienten påvirkes av studentens valg og målet er å redde pasienten. Om studenten skulle gjøre valg som vil dramatisk forverre situasjonen vil spillet ende i at en virtuell kollega/lege gripe inn og redde situasjonen.

Alle valg vil bli fortløpende registrert og i debriefing etter endt øvelse vil disse bli vurdert opp mot en fasit. Studenten kan også bli bedt om å vurdere egen innsats i et refleksjonsnotat på slutten av spillet.

Eksempel på ønsket funksjonalitet: Hjerterekompresjoner gjøres f.eks. ved å trykke på riktig plass på brystkassen, i korrekt tempo og riktige antall kompresjoner. Tilsvarende teknikk kan også anvendes for innblåsing.

Vi har også tanker om hvordan dette kan utvides og tilpasses pasienter i alle aldre og tenker at dette fint også kan utvides til et masterprosjekt.

### **Kontaktpersoner**

Katrine Aasekjær –førstelektor ved institutt for helse og omsorgsvitenskap – Master i jordmorfag. Underviser i grunnleggende jordmorfag, med fokus på studentaktive læringsformer og bruk av digitale ressurser i undervisningen. Har ansvar for simuleringstrening ved utdanningen, og er opptatt av at studentene i tillegg til å lære seg kliniske ferdigheter også utvikler evnen til å ta en beslutning.

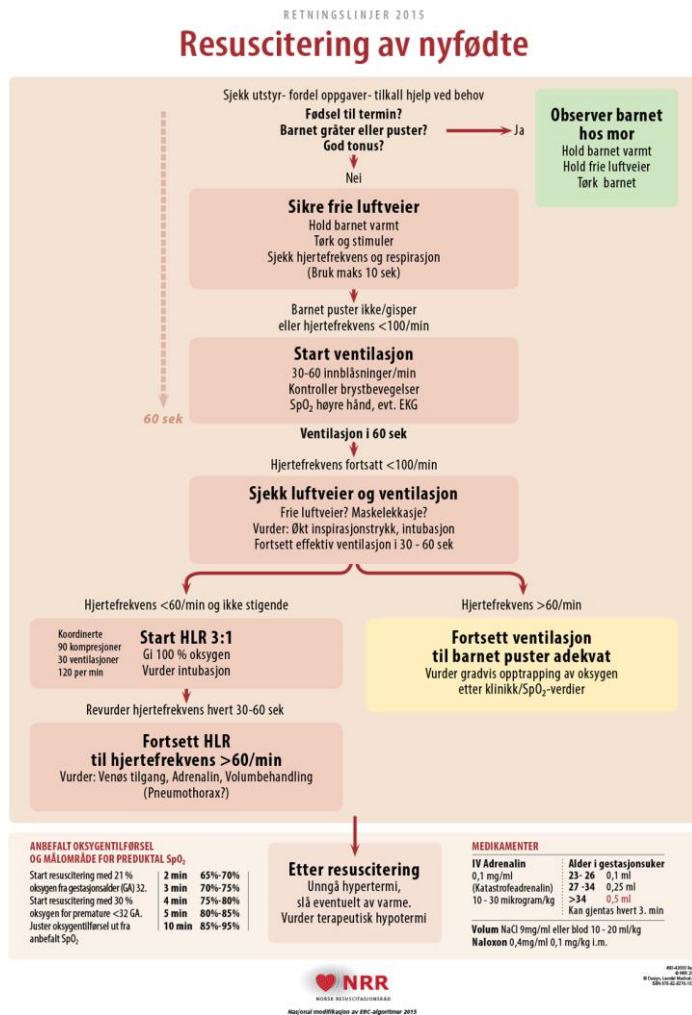
Kan kontaktes på [kaaa@hvl.no](mailto:kaaa@hvl.no)

Ivar Rosenberg -prosjektleder e-læring, Fakultet for helse og sosialvitenskap siden 2015. Har 10 års erfaring med e-læring og utvikling og bruk av interaktiv 3d i oljebransjen.

Kan kontaktes på [ivar.rosenberg@hvl.no](mailto:ivar.rosenberg@hvl.no) Mobil: 93617784



## 10.9 NRR algoritmen i detalj



Figur NRR algoritme for nyfødte.

Dette er en gjennomgang av algoritmen skrevet av gruppen, og inneholder derfor vår tolkning av den.

NRR algoritmen starter med en vurdering hvor man skal sjekke følgende:

- Sjekke utstyr, fordel oppgaver, tilkall hjelp om nødvendig
  - Utstyret man trenger for gjenopplivning skal være til stede, og satt opp riktig. Man må også starte en stoppeklokke for å kunne holde styr på tiden siden start.
- Fødsel til termin?
- Gråter barnet eller puster?
- God tonus?
  - Med dette menes naturlig spenning i muskler, i motsetning til et slapt livløst barn

Hvis man svarer ja på alle spørsmålene trenger barnet ikke gjenopplivning.

Ved nei på ett eller flere spørsmål skal følgende gjøres:

- Sikre frie luftveier
  - Sjekke at det ikke ligger blokkeringer i luftveien, sørge for at luftveiene ikke er kollapset.

- Hold barnet varmt
  - Tørke barnet, sørge for at det ikke mister kroppsvarme eller får lav kroppstemperatur
- Tørk og stimuler barnet
  - Tørke barnet, styrke over huden og stimulere det med hendene eller et håndkle for å forsøke å starte egenrespirasjon hos barnet
- Sjekke hjerterefrekvens og respirasjon

Disse stegene skal maksimalt ta 10 sekunder. Hvis barnet ikke puster, eller har hjerterefrekvens under 100 slag per minutt skal følgende gjøres:

- Start ventilasjon
  - Dette er ekstern tvunget respirasjon med Laerdalsbag. Sørge for å utføre med 30-60 innblåsninger per minutt. Kontrollere at brystkassen hever seg ved innblåsning.
- Sette på pulsoksymeter, eller bruke EKG
  - Pulsoksymeter måler oksygenprosent i blodet, og hjerterytme.

Fra man har startet algoritmen til man starter ventilering skal det helst ta maksimalt 60 sekunder. Man skal ventilere i 60 sekunder, og hvis hjerterefrekvensen fortsatt er under 60 slag per minutt gjør man følgende:

- Sjekke luftveier, sjekke ventilasjon
  - Forsikre seg om at luftveier er frie, og sjekke om masken til ventilasjonsbaggen forseglar mot ansiktet riktig
- Vurder økt inspirasjonstrykk, intubasjon
  - Inspirasjonstrykket justeres ved hvor hardt man trykker sammen baggen i hånden. Intubasjon vil si å føre en slange ned til lufterøret i halsen for å forbiassere eventuelle blokkeringer i luftveier ovenfor.

Man skal fortsette med effektiv ventilasjon i 30-60 sekunder, og så gjøre et veivalg i forhold til om barnet har puls over 60 eller under 60.

Over 60:

- Fortsett ventilasjon til barnet puster adekvat
- Vurder gradvis opptrapping av tilført oksygen i bag i henhold til oksygenmetning i blodet til barnet

Under 60:

- Start HLR
  - 3:1 forhold kompresjoner:ventilasjoner
  - 120 per minutt
- Gi 100% oksygen
- Vurder intubasjon

Man skal nå revurdere hjerterefrekvens hvert 30-60 sekund. Og fortsette med HLR til puls er over 60 per minutt. Man kan vurdere å gi adrenalin, og andre mer tekniske og avanserte teknikker. Dette er siste steget hvis barnet ikke viser bedring i tilstand.

Ved forbedret tilstand skal man unngå at barnet blir for varmt, vurdere å slå av varmelampe, eller vurdere å bruke terapeutisk hypotermi (nedkjøling av barnet).