

GoSaveThem

Finalized Project Report

Grupp 14

Carl Johan Freme
Nils Lindberg Odhner
Eric Lind
Ävelin Pantigoso
Emilia Fridsäll
Stina Långström
Soheil Saadati
Rasmus Bergendal

Handledare
Björn Thuresson
Företag
CrashCourse
Företagskontakt
Lukas

Innehållsförteckning

1. Introduktion	3
1.1 Rapportens innehåll	3
2. Omvärldsanalys	5
3. Produktkrav	7
3.1 Slutgiltiga produktkrav	7
3.2 Målgrupp	13
4. Produktbeskrivning	14
4.1 Sammanfattning av slutprodukten	14
4.2 Användargränssnitt	15
4.3 Produktens funktionalitet	27
4.3.1 Tutorial	27
4.3.2 Egenskaper "slumpas ut på skadade"	27
4.3.3 Tidtagning	28
4.3.5 Poängräkning	28
4.3.6 Triage-aktiviteter	30
4.3.7 Activity Diagram	32
4.4 Teknisk produktspecifikation	33
4.4.1 Game objects	33
4.4.1.1 Prefabs	33
4.4.1.2 Scener	34
4.4.2 Scripts	35
4.4.3 Införskaffade Assets	38
4.5 Framtida utveckling	39
5. Verifiering och validering	40
5.1 Traceability matrix	40
5.2 Produkttestning på användare	40
5.2.1 Sammanfattning av testresultat	41
6. Utvecklingens tillvägagångssätt	42
6.1 Arbetsmetod	42

6.2 Riskanalys	42
6.3 Arbetet över tid	43
6.4 Reflektion över projektet	43
7. Appendix	44
7.1 Verifiering och validering	44
7.1.1 Traceability matrix	44
7.1.2 Test-matris	46
7.1.3 Dependency matrix	47
7.2 Produkttestning på användare	49
7.3 Riskanalys	53
7.4 Arbetet över tid	54

1. Introduktion

Detta dokument behandlar ett skolarbete inom kursen Mjukvarukonstruktion, DD1393 där elever grupperas in i team och erbjuds möjlighet att komma i kontakt med företag och deras projektuppdrag. Ett av dessa företag heter *CrashCourse* och innehålls av Lukas Berg, vilka har bidragit med ett projektet som heter Gamification of Lifesaving Skills. Projektets idé bygger på att utveckla ett spel som insättar den arbetsmetod som ambulanspersonal tillämpar på en olycksplats. Det centrala i spelet är därför att följa ett triage, vilket är den typ av mall som ambulanspersonal arbetar utefter. Syftet med produkten är att ge allmänheten en lärorik introduktion till yrket som ambulanssjuksköterska men också att ge möjlighet till de som redan verkar inom branschen att repetera sina kunskaper. En viktig del av projektet är också att skapa ett spel som ska vara underhållande och att scenerna ska upplevas som komiska och ha en lätsam ton.

Projektet startades upp i december 2018 och löper fram till och med maj 2019, då en spelbar prototyp som skapats med hjälp av spelmotorn Unity lanseras och överlämnas till företaget.

1.1 Rapportens innehåll

Rapporten är uppdelad i sex kapitel. Kapitel 2 innehåller en omvärldsanalys om tidigare produkter inom samma område. Efter detta, i kapitel 3 nämns de krav som lett till den slutgiltiga produkten, här beskrivs också utförligt den målgrupp som produkten främst riktar sig mot. Kapitel 4 innehåller en mycket detaljerad bild av produkten, först genom en innehållsrik överblick av användargränssnittet och efter detta olika funktioner i spelet. Slutligen i kapitel 4 finns en beskrivning av kod och andra tekniska hjälpmedel som kan vara bra att känna till för den som är mer intresserad eller ska bygga vidare på produkten. Kapitel 5 beskriver den testning som gjorts. Slutligen, i kapitel 6 kan man följa arbetets flöde och läsa mer om på vilket sätt vi arbetat under projektets gång. Utförliga tabeller om risker med arbetet samt dess tidsplanering finnes i appendix sektion 7.3 respektive 7.4. Tidigare sektioner i

appendix innehåller tabeller när det kommer till testning men också svar från de testpersoner som intervjuats.

2. Omvärldsanalys

Det finns idag redan ett antal spel på App Store som behandlar nödsituationer. Ett av dessa spel heter Rescue Busters och behandlar situationer när det kommer till eldsvåda och första hjälpen. Spelaren styr en avatar som i ett antal minispel får möjlighet att hantera hjärt-lungräddning och Heimlich-manöver på en skadad person. Spelaren får också komma i kontakt med skadade som lider av chock, blödning, allergisk reaktion och medvetslöshet bland annat.

En likhet med produkten som vi ska producera och Rescue Busters är syftet att vara ett spel som är både lärorikt men samtidigt innehåller ett lekfullt uttryck. I Rescue Busters ges det bland annat ”roliga” kommentarer under spelets gång och spelaren får också möjlighet att klä avataren i en utstyrsel efter eget önskemål. Funktionen att som spelare själv få välja avatar och dess utseende är något som vi övervägt att tillämpa i den egna produkten. Detta har valts bort på grund av att införandet av en sådan funktion i en första version inte anses tillräckligt viktig i förhållande till den arbetstid det skulle kräva.

Uppbyggnaden av Rescue Busters är lik vår produkt på det sätt att spelet innehåller en kort scen där någon form av livräddande åtgärd ska ske innan nya banor är möjliga att låsa upp. Eftersom spelet inte har någon tidspress och på det sättet heller inte upplevs så dramatiskt, så är en målsättning att den mer förutbestämda tidsramen som ska appliceras på vår produkt skulle ge en mer intensiv och på så vis roligare spelupplevelse.

Det existerar också en mängd av lärorika appar inom sjukvård. Dessa applikationer bygger på att lära ut bland annat hjärt- och lungräddning genom information, filmer och illustrerade bilder. Problemet med dessa applikationer skulle kunna vara att det är svårare att fånga intresset hos främst yngre att ladda ner och bruka dem. På detta sätt skulle vår produkt med sin spelfunktion kunna locka en målgrupp som inte annars har ett intresse av att endast inhämta information utan också söker någon form av underhållning.

Ambulanspersonal förekommer också i existerande applikationer, där de främsta av dessa har en tendens att fokusera på att köra så fort som möjligt för att kunna rädda så många liv som möjligt. Applikationer som innehåller upplysning om den sjukvård som sker på en olycksplats med hjälp av att agera

utifrån ett triage har inte hittills kunnat hittas. Detta gör att vår produkt skulle kunna ha en förutsättning att tillföra kunskap som i dagsläget är mycket svår att finna i de applikationer som presenteras på App Store.

3. Produktkrav

3.1 Slutgiltliga produktkrav

Spelet ska ge möjlighet för ambulanspersonal att få användning av erfarenheter och kunskap inom sitt yrke i ett roligt spel där lärdom uppmuntras. Utöver att blåljuspersonal kan repetera sina redan innehavda kunskaper, ska även allmänheten kunna utnyttja spelet som en rolig plattform för att få insikt och grundläggande kunskaper i ambulanspersonalens arbete. En central del i att utveckla spelet är att finna en blandning mellan orden lärorikt och roligt.

Krav ID	Namn	Beskrivning	Prioritet
P1	iOS	Plattformen som spelet ska användas på är mobiler med iOS-system.	Hög
P2	Unity	Utvecklingen sker genom Unity-spelmotorn.	Hög
P3	Assets och paket	Miljön i spelet är utformat med resurser i form av Polygon-paketet och assets gjorda av Synty Studios.	Hög
P4	Triage	Spelet ska använda sig av <i>triage</i> som grund för spelmekaniken, se figur 1 nedan.	Väldigt hög
P5	Svårighetsgrad	Svårighetsgrader ska avspeglas i olika scener av olika nivåer.	Låg

P6	Poängsystem	Poängsystem utformas efter tid och antal räddade offer.	Medel
P7	Språk	Språket i spelet är engelska.	Medel
P8	Hazards	Spelaren ska kunna dö under spelets gång.	Låg
P9	Ljud	Bakgrundsljud och pratljud ska finnas med	Låg
P10	Prestanda	Spelet ska vara uppbyggt på ett sätt som minimerar prestandakraven	Hög
P11	Slump	De skadade och dess åkommor ska slumpsas ut	Hög
P12	Räddningsledare	Räddningsledare ska ge information om olycksplatsen	Medel
P13	Färdigställning	Produkten ska vara klar 22:a maj 2019.	Hög

Tabell 1 - Produktkrav

P1 - iOS

Enligt företagets önskemål ska spelet utvecklas till en iOS-enhet. Anledningen är främst för att en lansering i för iOS-enheter ger en stor spridning av produkten inom den relevanta målgruppen. Då Unity har enkla sätt att exportera produkter till önskad plattform är detta förhållandevis enkelt att uppfylla.

P2 - Unity

Att utveckla spelet i Unity är väldigt viktigt i och med dess enkelhet att använda. Detta gör att spelet kan utvecklas vidare på ett smidigt sätt i framtiden. Vår företagskontakt föreslog att vi skulle använda Unity tillsammans med dess verktyg *Unity Hub*. Vi övervägde för- och nackdelarna

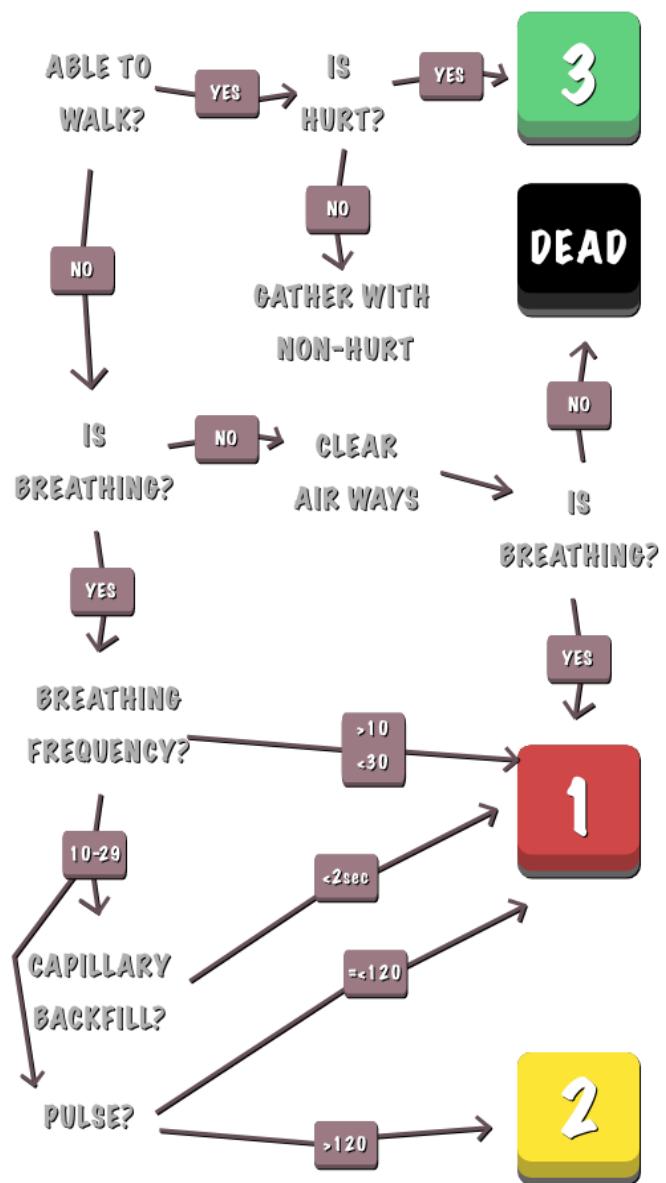
med att nyttja Unity Hub, men kom fram till att vi inte vill lägga till ytterligare en inlärningskurva till vår process. Resultatet blev att vi använde GitHub, vilket vi alla är bekanta med. Att nyttja Unity som spelmotor har dock alltid varit självklart. Däremot övervägde vi under en kort period att använda Unity tillsammans med *React Native* efter diskussion med vår handledare, men efter förtydliganden insåg vi att det inte tillför något värde till projektet.

P3 - Assets och paket

Strikta krav på hur spelet ska se ut har funnits. Detta gör att spelet blir enhetligt och enkelt att designa, inga egna animation behövs utan färdiga animationer kan avnändas.

P4 - Triage

Grundmekanismen i spelet är att göra kontroller på skadade personer som följer de aktuella triage-reglerna som finns. Detta är väldigt viktigt att det både görs bra och korrekt. I och med att hela spelet är byggt på triaget så har detta varit den största prioriteten under spelets utveckling. De kontroller som gjorts har varit puls, andning, kapillär återfyllnad, stoppad blödning och frigöring av luftvägar.



Figur 1 - Spelets triage

P5 - Svårighetsgrad

När spelet utvecklas ska svårighetsgraden bli högre för varje bana. Detta kan innebär fler skadade att göra livsuppehållande behandling på, kortare tidsfrist eller många hazards som är svåra att undvika. I och med att valet har gjorts att enbart göra en spelbar bana så har svårighetsgraden ej prioriterats. Detta är något som är enkelt att göra i framtida banor i och med att spelet är enkelt att utveckla vidare. Detta har gjorts genom att bland annat att göra så allmänna funktioner som möjligt så att de ska kunna tillämpas i flera situationer och på så vis uppmuntra modularitet.

P6 - Poängsystem

I en verlig situation är det väldigt viktigt att man arbetar snabbt men också att man gör korrekt. Gör man stora fel vid en patient, till exempel glömma att stoppa en blödning eller ge väldigt fel prioritet, ska detta synas i poängavdrag. Spelet ska uppmuntra korrekthet och snabbhet, vilket görs via poängsystemet.

P7 - Språk

Viktigt att spelets alla funktioner är på engelska. Detta eftersom ett sådant spel är mer gångbart i världen, vilket i sin tur gör att fler människor kan spela det.

P8 - Hazards

Spelet ska vara en kombination av lärorikt och roligt. För att åstadkomma detta har valet om att införa hazards gjort, vilket innebär att spelaren kan få "game over" om man är för oförsiktig. I den implementerade banan sker detta om man försöker gå över ett hål med gasledningar under och i framtida banor kan hazarden visualiseras genom att gå för nära en brinnande bil eller gå ut på en trafikerad gata och därav bli överkörd.

P9 - Ljud

Ljud ger en djupare spelupplevelse. I och med att de flesta ej har på ljud när man spelar ett mobilspel har ljudet ej prioriterats. Detta innebär att ljud finns som bakgrund undet spelet och ett prat-ljud när konversation sker. Ljud kan med fördel införas vid mätning av puls och andningsfrekvens, men detta är ingenting som har prioriteras.

P10 - Prestanda

I och med att enheten som ska användas till detta spel är en mobil så är prestandan begränsad. På grund av detta har val gjorts om att bland annat använda lågupplösta polygoner, partikelsystem används i så låg grad som möjligt och menybilden är gjord i orörlig 2D. Det är även viktigt att väntetiden för att ladda olika scener är låg och att spelet ej har några fördröjningar.

P11 - Slump

Spelets mål är att vara enkel att förstå men svårt att bemästra. I och med detta är det viktigt att man ej kan memorera en bana för att få så bra poäng som möjligt. Detta gör att det är viktigt att de skadade placeras ut på ett slumpmässig sätt och att deras parametrar också slumpas (dvs. puls, andning etc). Vi vill utmana spelaren till att lära sig och förstå triagen.

P12 - Räddningsledare

Företagskontakt önskade en räddningsledare på plats som ger information om olycksplatsens status. Detta för att spelet ska vara så verklighetstroget som möjligt då detta görs av ambulanspersonal i det ”riktiga” arbetslivet. Spelaren återkopplar också till räddningsledare när denna är klar med sitt uppdrag och på så vis avslutas spelet. Denna typ av återkoppling är också med i spelet för att göra det verklighetstroget i kontakten med räddningsledare.

P13 - Färdigställning.

Produkten ska vara klar och levereras i tid enligt överenskommelse.

En mer systematisk och detaljerad sammanfattning över kraven och dess beroende finns under avsnitt 6.1 Verifiering och validering.

3.2 Målgrupp

Målgruppen för produkten har diskuterats fram av företaget i samarbete med teamet, och målgruppen har fastställts till både blåljusperssonal och allmänheten. Anledningen till varför blåljusperssonal är en del av målgruppen är för att ett av spelets syften är att låta ambulansperssonal repetera sina färdigheter under en nödsituation i en underhållande miljö. Det tåls att nämna att spelet ej är till för att utgöra en uttömnande och komplett lärmiljö för blåljusperssonal. Spelet är snarare ämnat för att ge ett alternativt och lätsamt sätt att repetera viktiga principer och därmed ge dem en större trygghet när de möter dessa situationer under arbetstid.

I och med att blåljusperssonal är en förhållandevis liten målgrupp så har denna breddats till att även inkludera allmänheten. Detta eftersom det av uppdragsgivaren är önskvärt att spelet ska kunna bära sin egen kostnad. Spelet är intressant för allmänheten eftersom det ger en vanlig människa en inblick och förhoppningsvis lärdom av vad som sker och vad som ska göras på en olycksplats. På grund av spelets lekfulla uppbyggnad kommer det rikta sig främst mot ungdomar eller unga vuxna. En möjlig målgrupp som diskuterats är amerikanska skolungdomar i high school, eftersom elever där lär sig om triage under skoltid kan de tillämpa och förbättra sina kunskaper genom att spela spelet.

4. Produktbeskrivning

4.1 Sammanfattning av slutprodukten

Den levererade produkten är en *proof-of-concept*. Produkten uppfyller alla de ovanstående krav som ses i ”tabell 1 - Slutgiltliga produktkrav”, i sektion 3.1. Arbetet har skett genom att arbeta med fokus på kvalitet framför kvantitet, vilket innebär att den levererade produkten har begränsats till att innehålla en bana med en bilolycka som innehåller full spellogik tillsammans med en tutorial i samma scenario. Detta för att ha en demo som är lätt att bygga vidare på. Tanken är att huvudsakliga funktionaliteter i spelet finns som prefabs vilket gör att nya banor kan skapas enklare, eftersom prefabs kan användas som ”drag-and-drop”. Spelet innehållar en startmeny där man får en överblick över de banor som spelats. Till denna startmeny ska det vara simpelt att addera fler banor och nya banor ska läsas upp utefter poäng.

Banan innehåller utplacerade skadade ”människor” av olika typer, vilket gör att alla spelets huvudfunktioner används. Detta inkluderar att egenskaper slumpats ut på de skadade, till exempel blödning, livlös, saknar andning, hög respektive låg puls, igentäpta luftvägar, dålig kapillär återfyllnad och hög respektive låg andningsfrekvens. Spelaren får möjlighet att undersöka den skadade och kan sedan med hjälp av triage-regler sätta en prioritet på denna. Spelarens huvudsakliga uppgift är att hitta alla skadade och sätta en prioritet på dessa så fort som möjligt innan spelaren går till räddningsledaren och spelet avslutas. Banan innehåller också en hazard i form av gasläcka som gör att spelaren själv kan avlida vid oaktsamt beteende. Poängräkningen är fullständigt implementerad och poäng beräknas beroende på hur rätt spelaren lyckas prioritera de skadade, där triaget är facit.

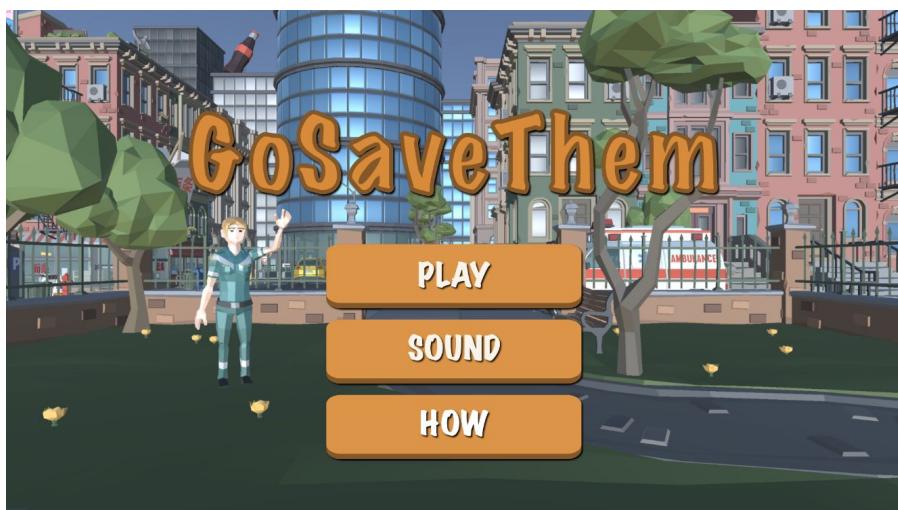
4.2 Användargränssnitt

I följande avsnitt presenteras alla de scenarier som en spelare kan hamna i under spelets gång utav en screenshot tillsammans med tillhörande beskrivning. Bilderna presenteras i kronologisk ordning för att ge en överblick över hur spelet spelas och vilka val som är möjliga att göra i respektive spelvy. Utseende och utformningen av användargränssnittet skapades med avseende på två faktorer. Den första är att spelet ska innehålla ett lekfullt uttryck, vilket nämns i avsnittet "målgrupp". Den andra är att spelet ska vara utformat med hjälp av Polygon City Pack vilket är ett av de krav som fanns på produkten.

Huvudmeny

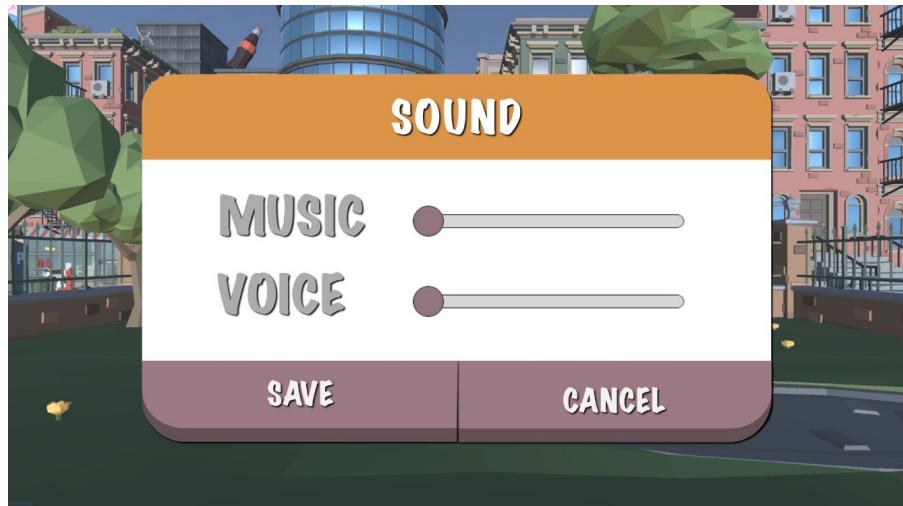
Screenshot 1. - Huvudmeny

Huvudmeny vid spelets början. Går att ställa in ljud, kolla på triaget under "How" och sedan gå vidare mot att börja spela.



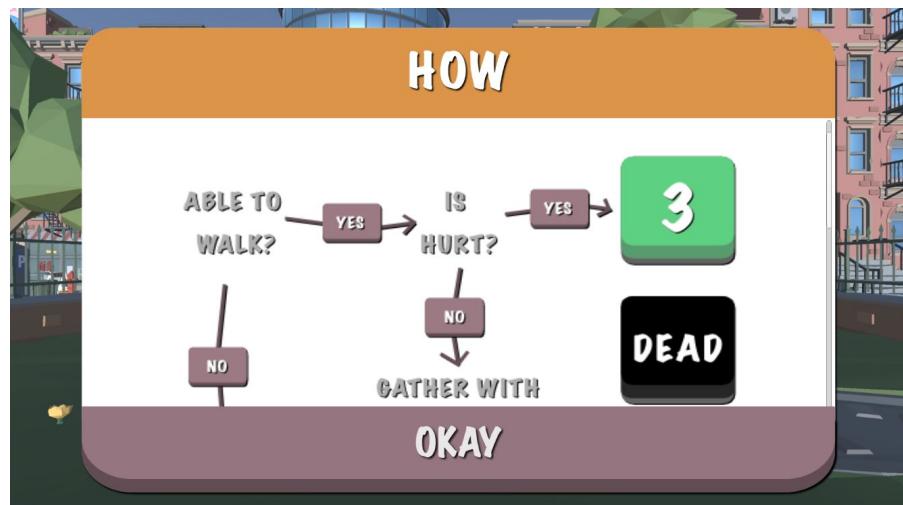
Screenshot 2. - Sound-vy (huvudmeny)

Här kan ljudet aktiveras eller avaktiveras för både röster och musik.



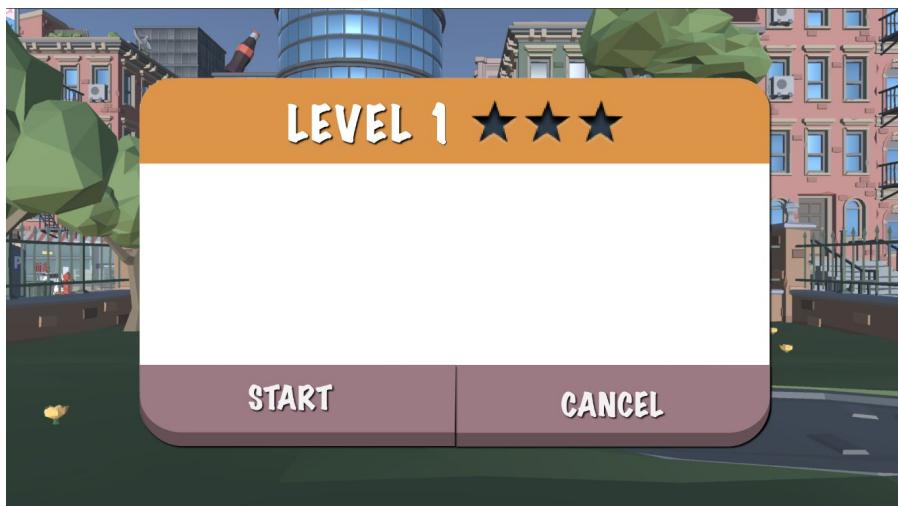
Screenshot 3. - Triage-vy (huvudmeny)

Denna vy gör det möjligt för spelaren att titta på triaget och förbättra sina kunskaper innan det att spelet startas.



Screenshot 4 och 5. - "Välja spel" - meny

Efter att ha tryckt på play kommer användaren till denna skärm och här väljer man vilket uppdrag man vill utföra. För att låsa upp nya banor måste spelaren samla på sig stjärnor, ifall spelaren inte har tillräckligt många stjärnor får denna gå tillbaka och förfina sin teknik. Detta är något som kommer vara mer applicerbart i en framtida version när spelet har flera banor. Just nu är level 1 en tutorial och level 2 den första spelbara banan.



Spelvy

Screenshot 6. - Ursprunglig spelvy

Avataren följs med en fixerad kamera som går att zooma in eller ut mellan två olika nivåer, detta sker med ett dubbeklick. Längst upp i skärmens högra övre hörn syns en pausknapp. I det nedre vänstra hörnet har vi en joystick med vilken avataren styrs. På höger sida finns tre knappar som vecklas ut (1). Med hjälp av dessa knappar kan vi senare utföra handlingar när avataren kommer till en räddningsledare eller skadad person. Det finns möjlighet för spelaren att titta på triaget även under spelets gång, dock kommer speltiden att fortsätta räknas ner medans triaget är öppet (2). I övre vänstra hörnet syns den speltid som spelaren har kvar.



Screenshot 7. - Pausmeny

Under hela spelets gång finns möjlighet att pausa spelet och då komma till följande meny där det är möjligt att ställa in ljud, starta om spelet eller avsluta. När spelet avslutas på detta sätt sker ingen poängräkning.



Räddningsledare

Screenshot 8. - Vy hos räddningsledare

När spelaren anländer till räddningsledare finns det möjlighet att interagera med denna på två olika sätt. Ett utav dem är att prata och den andra är att avbryta spelet och på detta sätt ”meddela att man är klar”.



Screenshot 9. - Talk

Hos räddningsledaren finns möjlighet att få information om platsen, och om eventuella Hazards, vilket är en fara som kan förekomma på platsen. Faran består här av en gasläcka som spelaren kan råka gå in i och på det sättet dö.



Screenshot 10. - Avsluta spel

Spelet avslutas. Spelaren får feedback och poäng på hur spelomgången gick.



Skadad person

Screenshot 11. - Vy vid skadad person

Om spelaren kommer tillräckligt nära en skadad kommer denna att omringas av en vit ring för att markera att spelaren kan interagera med den skadade. Då aktiveras också knapparna i det nedre högra hörnet. När detta sker kan spelaren utföra diverse undersökningar på den skadade. Man kan kontrollera andning (3), puls (4) och kapillär återfyllnad (5), och också stoppa blödning (6) och frigöra luftvägar (7). Det går att se om den skadade blöder eller inte, men annars kan inte tillståndet hos den skadade synas utanpå.



Med hjälp av att utföra ovanstående nämnda undersökningar, ska spelaren, beroende på sin kunskap av triaget, kunna uppfatta sig en bild av vilken prioritet som bör sättas på den skadade. Med andra ord, kan spelaren uppskatta hur dåligt skick den skadade befinner sig i eller om hen till och med är livlös. Inga undersökningar måste göras för att kunna ge en prioritet, men det är en förutsättning för att klara spelet. För att ge prioritet trycker spelaren på prioritets-knappen.

Screenshot 12. - Puls

Väljer spelaren att kontrollera den skadades puls kommer denna röda bild på skärmen att pulsera i takt med den puls som den skadade har. Det är spelarens uppgift att räkna pulsen.



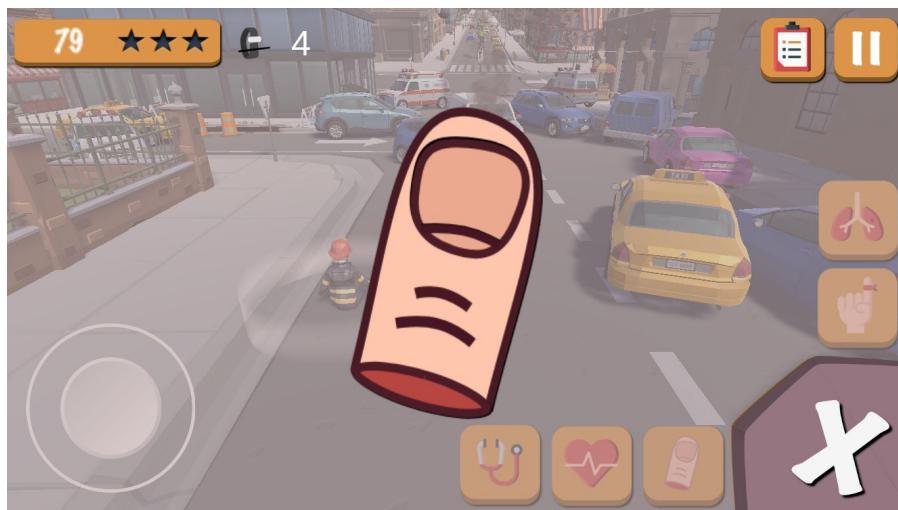
Screenshot 13. - Andning

Väljer spelaren att kontrollera den skadades puls kommer denna vita imma att återkomma på skärmen i takt med den andningsfrekvens som den skadade har. Det är spelarens uppgift att uppskatta andningsfrekvensen.



Screenshot 14. - Kapillär återfyllnad

Väljer spelaren att kontrollera den skadades kapillära återfyllnad kommer denna tumme att visas på skärmen. Spelarens uppgift är att hålla in tummen tills dess att den är vit, och sedan uppskatta den tid det tar för den skadade att återfå sin normala hudfärg.



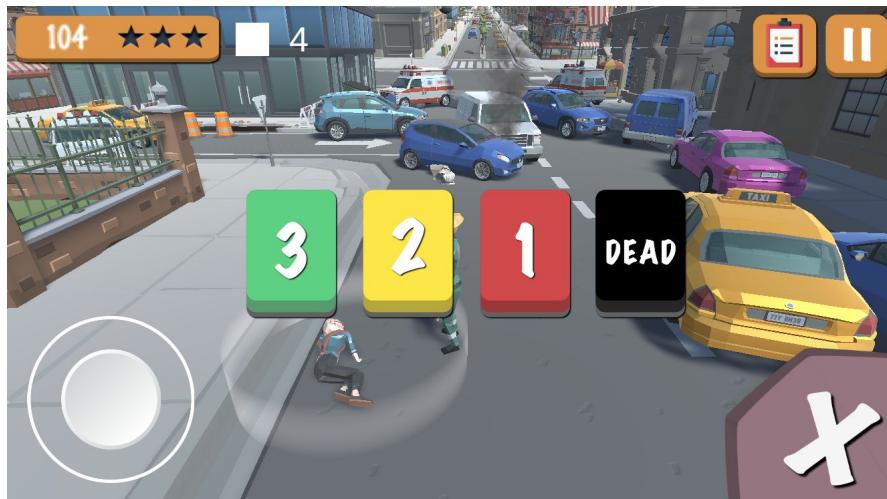
Screenshot 15. - Stoppa blödning med Troquet & frigöra luftvägar

Väljer spelaren att stoppa blödning kommer denna att sluta. Det är ett krav att stoppa blödning på alla de skadade som blöder för att ha möjlighet att få högsta poäng. Om spelaren istället väljer att frigöra luftvägar kommer det vara möjligt för den skadade att återfå en andning. Båda dessa åtgärder iscensätts av en animering där avataren går ner på knä vid den skadade.



Screenshot 16. - Prioritetskort

Med hjälp av denna vy kan spelaren välja en prioritet som den skadade får.



Screenshot 17. - Prioriterat skadad

Efter att spelaren har valt en prioritet kan man se att den skadade har blivit tilldelad denna prioritet genom den lilla symbolen över den skadades huvud.

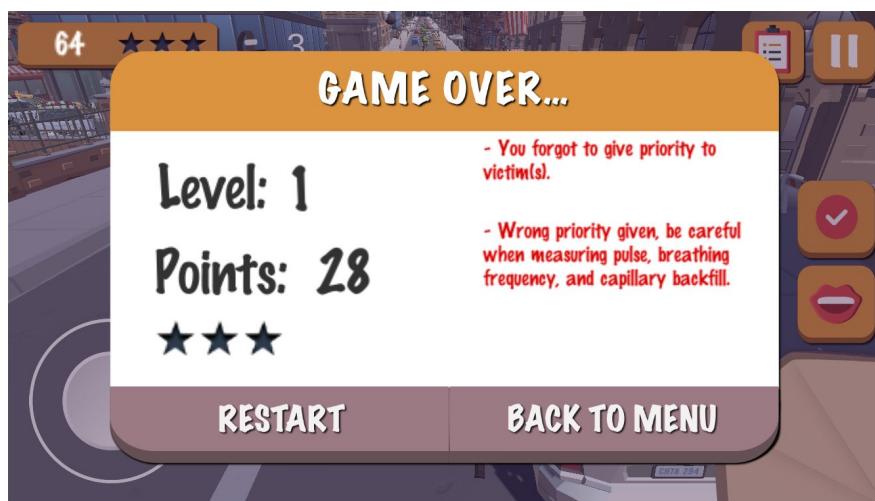


Game Over

Game over kan ske på tre olika sätt. Det första sättet sker genom att spelaren återvänder till räddningsledare och meddelar att hen är klar. Alternativt dör spelaren dör på grund av faran (gasläckan) eller att speltiden tar slut.

Screenshot 18. - Game Over med återkoppling

När spelaren avslutar spelet hos räddningsledaren får spelaren sina poäng tillsammans med feedback för spelomgången. Denna feedback består av maximalt tre påståenden som kan förbättras för att få högre poäng.



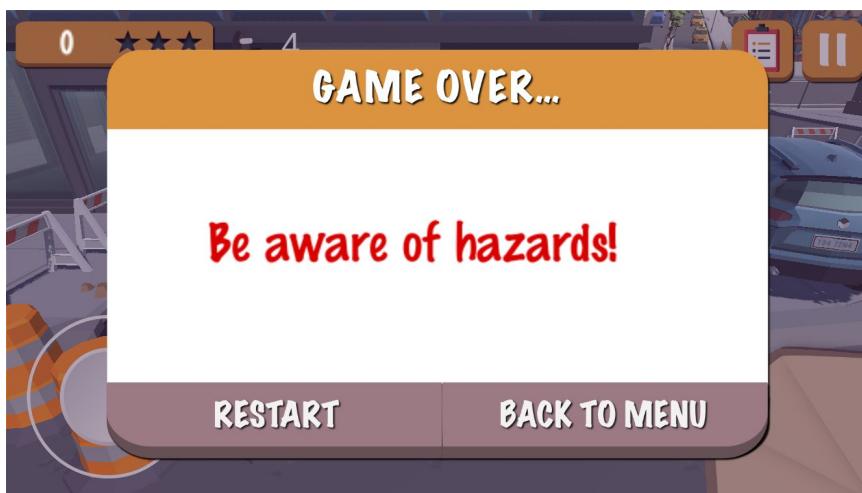
Screenshot 19. - Hazard

Om spelaren går in i ett område med en hazard dör hen. Detta visualiseras i denna bana genom en explosion i en gasledning.



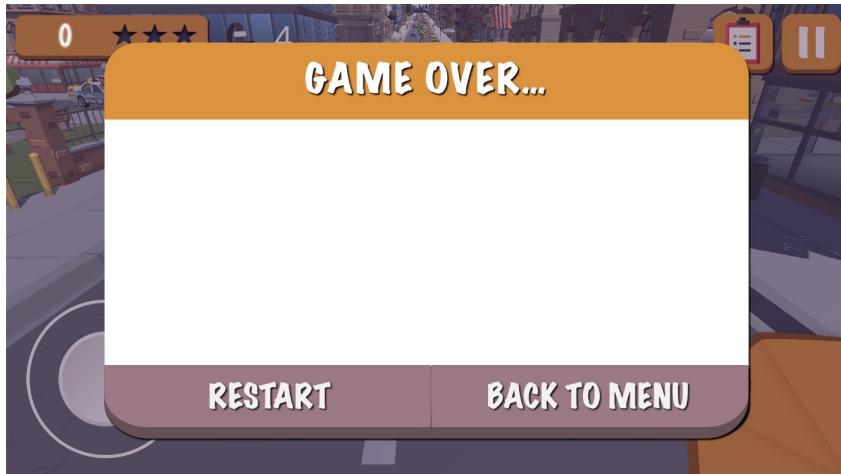
Screenshot 20. - Game Over efter hazard.

Om spelaren dör av en hazard i spelet får denna inga poäng och texten "Be aware of hazards!" visas för spelaren.



Screenshot 21. - Game Over när tiden tagit slut

Om spelaren inte självmant har avslutat spelet när tiden tagit slut avbryts spelet och en skärm med Game Over visas. Inga poäng ges.



4.3 Produktens funktionalitet

I sektion 4.2 ges en övergripande bild av hur spelet spelas och hur dess interface är uppbyggt. Vissa funktionaliteter och implementeringar beskrivs här mer detaljerat.

4.3.1 Tutorial

Den första banan i spelet är en tutorial som går igenom spelets logik, de knappar som visas på skärmen och spelarens huvudsakliga uppgift. Valet att skapa en tutorial gjordes i och med att spelet har en hög inlärningströskel med många olika funktionaliteter. Tutorialen går igenom spelets hela flöde, från att prata med räddningsledaren, gå till ett offer och utföra livsuppehållande handlingar till att avsluta spelet vid räddningsledaren.

4.3.2 Egenskaper "slumpas ut på skadade"

Spelet ska vara lätt att lära men svårt att bemästra. På grund av detta är det viktigt att banorna ej går att memorera för att få så bra betyg som möjligt. För

att lösa detta så slumpas alla skadade och dess placeringar ut för varje spelomgång. Alla parametrar för varje skadad byter också värde efter varje spelomgång. Detta sker dock ej helt slumpmässigt i och med att värdena på alla parametrar är beroende på hur skadad personen är. Det är exempelvis inte möjligt att ett offer som är livlöst har god andningsfrekvens.

4.3.3 Tidtagning

För att få en ytterligare svårighet och motivera en stress i spelet som liknar stressen en ambulanssjuksköterska känner i verkligheten har en tidtagning implementerats. När tiden tar slut så avslutas spelet automatiskt. Tiden påverkar även poängsättningen för att uppmuntra spelaren att öva upp en snabbhet i sin bedömning.

4.3.4 Feedback

I samband med att poängen presenteras för spelaren ges även en återkoppling på spelomgången. Detta består av maximalt tre uppmaningar att förbättra något i spelet. De uppmaningar som visas är rangordnade i olika prioriteringar där de största felet har högst prioritet. Detta är om man har missat någon skadad, om man ej har stoppat blödning på någon som blöder eller om man ej har frigjort luftvägar på någon som behövt det. Annan återkoppling man kan få är om man utfört någon handling i onödan eller om man gett fel prioritet på grund av misstag vid pulskontroll eller liknande. Feedbacken finns för att spelaren ska bli motiverad att göra om och göra bättre. Detta för att lära sig så mycket som möjligt.

4.3.5 Poängräkning

Tanken är att spelet på sikt ska innehålla flera banor, och då också en lägsta nivå av poäng som krävs på varje bana för att ta sig vidare till banan efter. Poängberäkningen sker utifrån några olika parametrar:

- Tid
- Antal rätt markerade skadade
- Antal missade skadade

Med hjälp av dessa utformas ett poängsystem som återspeglar spelarens prestationer. Poängen är uppdelade i tre etapper som presenteras med hjälp av stjärnor. En spelare kan få mellan 0 till 3 stjärnor.

X = Spelarens andel av totalt antal möjliga poäng

Icke-godkänd:		X<=25%
Godkänd:		25% > X >= 50%
Väl godkänd:		50% > X >= 75%
Mycket väl godkänd:		X >= 75%

Ekvation - beskrivning:

- Spelaren börjar med 0 poäng.
- För varje sekund kvar av tiden får spelaren **X=2** poäng.
- För varje korrekt triage-markering får spelaren **Y (= 100X)** poäng.
- För varje missad markering dras **Z (=75X)** poäng av från spelarens insamlade poäng. Förhållandet till **Y** bör vara 3:4.
- För varje felaktig markering och icke stoppad blödning dras **Q (=25X*n)** poäng av från spelarens insamlade poäng.
Förhållandet mellan **Y** och **Q** bör vara 1:4. **n** är ett heltal mellan 1 till 3. Värdet på **n** beror på hur felaktig markeringen är. För varje "steg" av felaktighet ökar **n** med 1. Om någon ska ha ett svart kort (4/4) och istället får ett grönt (1/4) är **n** = 3.

Ekvation:

$$\begin{aligned}
 & 0p + (\text{antal sekunder tillgodo} * X) + (\text{antal korrekta markeringar} * 100X) \\
 & - (\text{antal missade markeringar} * 75X) - (\text{antal icke stoppad blödning} * 25X) \\
 & - (\text{antal felaktig markering} * 25X) = \text{Resultatet}
 \end{aligned}$$

4.3.6 Triage-aktiviteter

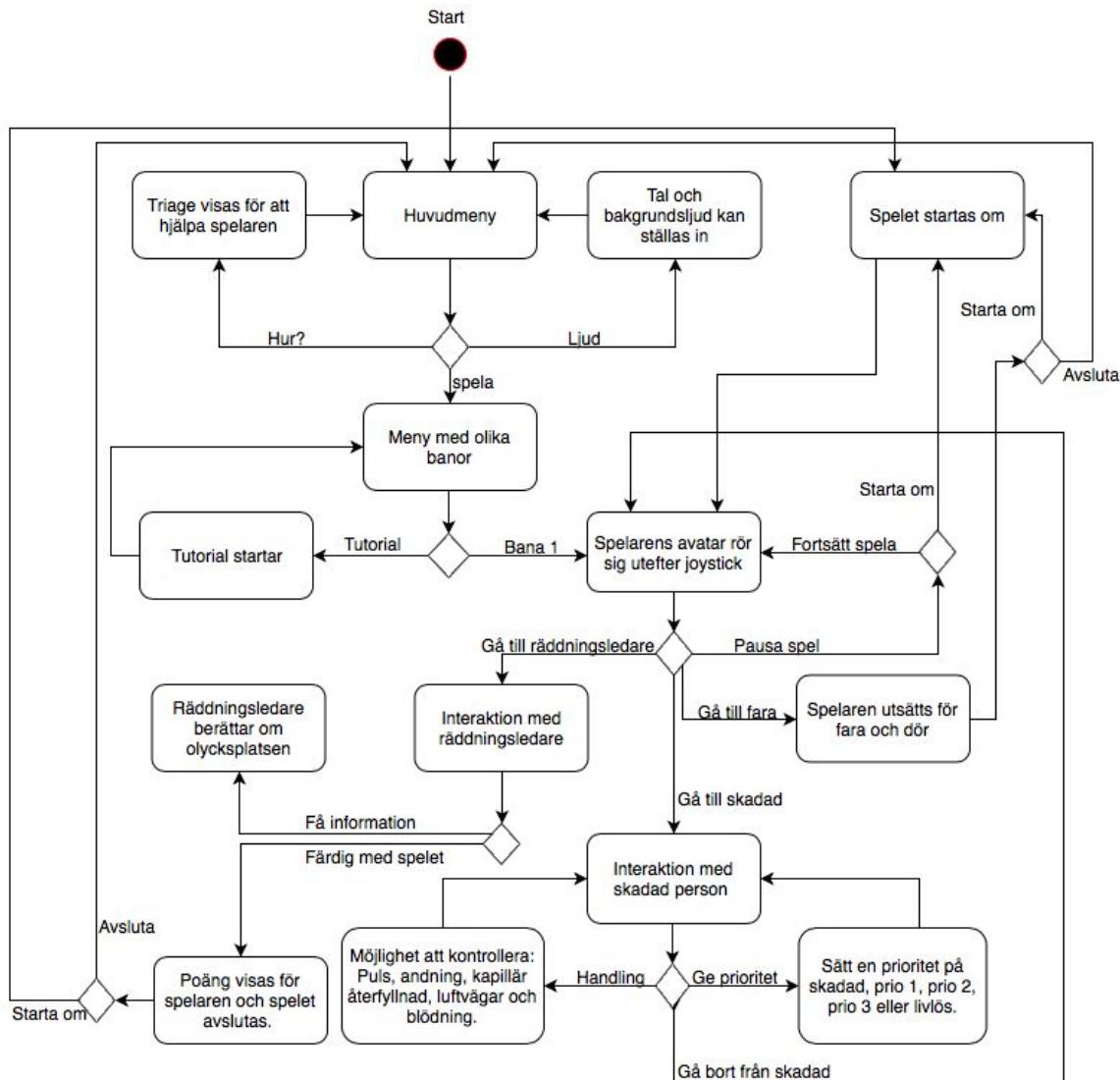
För att göra rätt bedömning av en skadad person finns det olika aktiviteter som spelaren kan utföra.

Aktivitet	Implementering
Puls	Vid pulskontroll läses den skadades puls-parameter av och registreras. Denna information ger ett värde på hur lång tid det är mellan varje pulsslag. Pulsslagen visualiseras sedan för spelaren på skärmen genom att en röd bild som föreställer blod pulserar på skärmen i den takt pulsen är. Denna bild pulseras så länge spelaren håller in pulsknappen.
Andningsfrekvens	Vid andningsfrekvenskontroll läses den skadades anings-parameter av och registreras. Denna information ger ett värde på hur lång tid det är mellan varje andetag. Andetagen visualiseras sedan för spelaren på skärmen genom att en dimma framträder i den takt den skadade andas. Denna bild visas så länge spelaren håller in andningsknappen.
Kapillär återfyllnad	När kapillär återfyllnad kontrolleras kommer det upp en bild av en tumme på skärmen. När spelaren håller in på tummens nagel ändras färgen från beige till vit under 2 sekunder. När spelaren sedan släpper nageln återgår färgen på tummen i den hastighet som finns under den skadades parameter.
Frigöra luftvägar	Varje skadad har en parameter som är true eller false beroende på om luftvägarna är blockerade eller inte. Om spelaren väljer att frigöra luftvägar så kommer

	denna variabel att ändras. Sedan killas det om variabeln "dead" är true, är den inte det så kommer det slumpas ut en ny andningsfrekvens på den skadade. Är den skadade istället död hjälper det inte att frigöra luftvägarna, det leder då inte till att en andningsfrekvens slumpas ut, utan denna är fortfarande obefintlig.
Stoppa blödning	När spelaren väljer att stoppa blödning kommer en parameter för blödning hos den skadade att ändras från true till false. Detta innebär i sin tur att den visuella blödningen också upphör.

4.3.7 Activity Diagram

I figur 2 nedan visas spellogik med hjälp av ett "activity diagram". Diagrammet ger oss en tydlig bild över spelets delar och hur de är beroende av varandra.



Figur 2 - Activity Diagram

4.4 Teknisk produktspecifikation

I denna sektion ges det en detaljerad beskrivning av tekniska filer, deras användning och deras relation. Detta syftar på Assets som är skapade eller modifierade av projektgruppen. Filer som är köpta via Unitys Assets store har angiven documentation inom respektive assets mappar, en kortare beskrivning av dessa finns i 4.4.3 införskaffade assets. Alla filer beskrivna inom den sektion ligger under **GoSaveThem -> Assets -> Own**, förutom scener som ligger under **GoSaveThem -> Assets -> Scenes**.

4.4.1 Game objects

4.4.1.1 Prefabs

Prefab	Beskrivning
Initiate	Innehåller spawn funktioner, för att kunna generera Victim object. Deras position inom scenen bestäms av spawnpoints som ligger som ett gameobject under Initiate/SpawnPoints.
Hazard	Ett osynligt 3D-objekt med "Box Collider". Triggas när spelaren kolliderar med boxen och spelaren dör (Game Over)
HUD	Canvas som innehåller alla knappar, pop-ups, joystick samt tid- och poängmätaren.
Tutorial HUD	Innehåller alla objekt, knappar och pop-ups som visas för spelaren. Binder samman dessa m.h.a. scripts och tillsammans skapar dessa en spellogik i tutorial sammanhang.
Player	Spelarens object rör sig efter touch controller och animationscontroller sätter bestämmer hur object meshen animeras.
Rescue Leader	Räddningsledare som förekommer i början av varje level. Spelaren kan prata med ledaren samt avsluta leveln.
Victim	De skadade personerna som ligger utspridda över leveln. Dessa har parametrar som representerar vitala värden

	(puls, andningsfrekvens, kapillär återfyllnad samt blödning/andningsstopp). I prefabben finns en ring som lyses upp när den skadade är markerad som "aktiv" (Det är på den aktiva skadade som spelar utför sina undersökningar).
Tutorial Victim	Motsvarar ett standard victim. Skillnaden är att denna är specifikt skapad för att fungera i tutorial sammanhang och . Separerar de två victim typerna för att motverka att ändringar i standard victim påverkar tutorial och vice versa.
Particle Systems • Blood • Oil	partikelsystem som genererar partiklar föreställande blod/olja.
Music	Sätter igång musik-filen som loopar under spelets gång i alla scener.

4.4.1.2 Scener

- **Main Menu**

Spelet startas från denna scen. Scenen innehåller en start-vy som inkluderar menyknapparna Play, Sound och How. Denna scen finns för att orientera spelaren mellan de olika nivåerna i spelet.

- **Level 1 (Tutorial scene)**

För att spelaren snabbt ska förstå grundmekaniken i spelet och få information om alla spelets knappar är den första banan i spelet en handledningsbana. Banan består av instruktioner vad spelaren ska göra i varje steg och förklarar hur man orienterar sig i banan.

- **Level 2**

Detta är scenen där huvudspelet finns. I denna scen finns hela spelets grundkoncept implementerat, vilket inkluderar alla triage-kontroller man kan göra, konversation med räddningsledaren, hazards och ett poängsystem baserat på spelarens prestation. Det är utifrån denna scen resten av spelets nivåer kommer utgå från i och med att denna scen innehåller alla funktioner spelet ska ha.

4.4.2 Scripts

GameObject	Script	Beskrivning
Player	Movement.cs	Tar input från joystick:en och rör spelaren utefter denna. Har även metoder för spelarens olika animationer samt en metod för när spelaren dör.
Main Camera	PlayerCameraController.cs	Hanterar kamerans rörelse så att spelaren alltid är i mitten av skärmen.
Main Camera	zoomTap.cs	Vid dubbelklick på skärmen byter kameran mellan två lägen; inzoomat eller utzoomat.
HUD/White	Backfill.cs	Vid tryck på tummens nagel blir denna vit med konstant hastighet. När spelaren sedan släpper återgår nageln till ursprunglig färg med en hastighet som hämtats från den "aktiva" skadade.
HUD/Pulse	TakePulse.cs	Medan knappen för pulstagning hålls intryckt pulserar en bild med blodstänk. Pulseringen hanteras med time.deltaTime och sker med den frekvens som hämtats från den "aktiva" skadade.
HUD/Breathing	BreathingEffect.cs	Medan knappen för andningsfrekvens hålls intryckt pulserar en bild med imma på skärmen. Detta hanteras på samma sätt som TakePulse.
HUD	BtnReact.cs	Innehåller metoder för hantering av olika knapptryck. Väljer vilka knappar som ska visas samt vilka som ska döljas.
HUD/ThumbBlur	CloseThumb.cs	Döljer tummen när spelaren trycker någonstans utanför tummen.
HUD/CountDowntxt	CountDown.cs	Räknar ner nivåns tid, från den satta tidsgränsen och neråt. Tiden skrivs sedan ut i en text component inom objektet scriptet ligger i, när tiden tar slut pausas leveln och en game over meny dyker upp.

GoSaveThem
Gamification of Lifesaving Skills

RescueLeader/ Radius	RescueMeny.cs	Vid närmande av räddningsledaren visas, ui knappar upp för räddningsledarens funktioner. En knapp är för att påbörja en konversation och den andra är till för att avsluta nivån och få sina poäng.
TutorialHUD	TutorialBtnReact.cs	Detta script motsvarar ordinarie BtnReact.cs med viss modifiering för att fungera i samband med tutorial samt för att separera dessa så att påbyggnad av BtnReact.cs inte påverkar tutorialn där spel logiken är begränsad.
Victim/radius	UIAppear.cs	Vid närmande av en skadad visas de olika undersöknings alternativen. Den skadade blir markerad som "aktiv" och detta visas av att en ring börjar lysa kring denna.
Victim	Parameters.cs	Victim initieras med en slumpmässigt vald prioritet (grön, gul, röd eller svart). Utefter den valda prioriteten slumpas den skadades attribut (puls, andningsfrekvens, kapillär återfyllnad samt huruvida en blödning förekommer)
Victim/PriorityCard	ShowPriority.cs	Hanterar renderingen av "kristallen" som flyter över varje skadad. När spelaren trycker på ett prioritetskort börjar "kristallen" lysa i den färg som valts.
StartLevel	LevelSettings.cs	Innehåller parametrar för en viss Level. Dessa parametrar avgör antalet skadade personer, och hur många av varje prioritet som ska existera.
Initiate/Spawn Points	Spawn.cs	Använder level settings för att generera och sätta upp parameters för skadade (victim object).
Hazard	HazardScript.cs	Vid kollision med spelaren spelas en animation där spelaren dör. Därefter visas "Game Over"
Victim	victimController.cs	Sköter animationer hos de skadade (sittande, liggande osv.).
Player	Feedback.cs	Genererar feedback utifrån hur spelaren

		presterade under spelets gång. Feedbacken presenteras för spelaren när banan är avslutad.
Ligger ej i object	LevelControl.cs	Hanterar övergångar mellan scener.
main_menu scenen: canvas	menuActions.cs	Används för att skifta mellan huvudmenyns olika faser genom att visa och dölja olika menyLAGER.
Player	PointCount.cs	Räknar ut poäng ut efter poängekvationen som beskrivs i 4.3.5 Poängräkning. Poängen, samt antal stjärnor, visas sedan upp i en meny efter avklarad nivå.
TutorialHUD/TutorialView	Tutorial.cs	<p>I detta script skapas spellogiken för tutorial. I scriptet finns ett antal funktioner som öppnar och stänger pop-ups i en specifik ordning, vars syfte är att vägleda spelaren genom spelets möjligheter i en särskild ordning. Det finns en funktion som återställer pop-ups när man är inom räckhåll för en victim så att dessa pop-ups dyker upp varje gång man är inom ett victims räckhåll (<i>EnteredVictimArea()</i>). På samma sätt finns ett script med motsvarande funktionalitet för Rescue Leader (<i>ShowCommunicate()</i>). Utöver dessa finns ytterligare en funktion som förhindrar spelaren från att bli färdig såvida denne inte placerat ett triagekort på ett victim (<i>FinishTutorial()</i>). De funktioner som baseras på spelarens räckhåll körs i en update-metod som alltid ser över vilka villkor som uppfylls (<i>Update()</i>).</p> <p>Spellogiken styrs framförallt av följande flaggor (<i>booleans</i>): NotShownCommunicate, NotShownVictim, IsXClickable, IsTripleMidClickable, IsVictimMarked, IsTutorialFinished.</p> <p>I enlighet med namnen visar dessa nuvarande status på spelet. T.ex. avgör IsXClickable om stängknappen på spelarens actions är aktiverad eller inaktiverad. IsTutorialFinished</p>

		kontrollerar ifall spelaren genomför de handlingar som färdigställer en tutorial eller ej. Innebördens av dessa flaggor syns tydligt i scriptet.
--	--	--

4.4.3 Införskaffade Assets

Asset paket	skapare	Beskrivning
PolygonCity	Synty	Paket innehållande meshes för karaktär, bilar och stadsmiljö. Karaktär används för spelarens avatarr, skadade och räddnings ledare. Alla karaktär-meshes har ett humanoid skelett vilket gör det svårt att skapa egna animationer med verktygen som medkommer med Unity. Meshes för bilar och stads-miljö så som vägar och hus används när man skapar en ny miljö för en scen. Tips är att titta i demoscenen som medkommer i detta paket för att få inspiration. Det kan även hjälpa att ta bilder av demoscenen för att slippa modellera hela staden, denna lösning används i båda scenerna skapade i detta projekt.
PolygonCityCharacters	Synty	Extra characters meshes.
PolygonPrototype	Synty	Innehåller meshes för att kunna skapa en prototyp av ett spel. Vissa FX objekt från detta paket har används för att förstärka spelet.
RPG Character Animation Pack	Explosive	Innehåller ett helt system för RPG animationer med en fullt utbyggd animationscontroller. Endast animationer från detta paket används i projektet. Dessa animationer är humanoid anpassade vilket förenklade arbetet.
TextMesh Pro	Unity	Används för all typ av text i spelet, eftersom den ger ett mycket bättre resultat än Unitys inbyggda text.
Viritual Joystick	Fenerax	Används för touch Joysticken som spelet

Pack	Studios	använder sig av för att man ska kunna kontrollera karaktären.
------	---------	---

4.5 Framtida utveckling

Vår sluttgiltiga produkt kommer som sagt att innehålla en tutorial tillsammans med en fullt spelbar bana. Det finns mycket att bygga vidare på, och en av de sakerna som diskuterats och som skulle kunna göras i första hand är en high score-lista i huvudmenyn. Tanken med denna är också att spelarens egna poäng ska kunna sparas på fil och att man på så vis kan återkomma till spelet med samma poäng som man hade föregående spelomgång. Detta är i dagsläget inte möjligt.

En sak som vore önskvärt i framtiden är också att införa ”friska” personer på olycksplatsen som spelaren ska hänvisa till en uppsamlingsplats efter att ha haft en konversation med dessa och frågat hur de mår. Denna funktion är något som skulle kunna tillföra en mer verklighetstrogen spelupplevelse eftersom uppsamlingsplatser för de som inte är i behov av sjukvård är något som tillämpas i verkligheten.

En annan självklar sak som spelet är uppbyggt för är att implementera flera banor. Dessa kommer kunnas bygga på samma typ av spelmekanik, men med fler skadade, mer tidspress och fler faror som spelaren kan råka ut för under spelets gång. För att utföra all typ av påbyggnad, läs med fördel kapitel 4.4 för att få en insyn in den tekniska produktbeskrivningen.

5. Verifiering och validering

5.1 Traceability matrix

De tidigare nämnda kraven har på en mer detaljerad nivå införts i en matris tillsammans med de test som behöver göras för att säkerställa att kraven uppfylls. De produktspecifika kraven har brutits ned till delkrav för att systematiskt kunna kartlägga dem och följa upp med tester. Produktkraven från 3.1 har alltså delats upp enligt **appendix 7.1.1**.

Test-matrisen innehåller en beskrivning av de test som ska utföras tillsammans med dess id och vilka krav som varje test testar. Denna redovisas i **appendix 7.1.2**.

Dependency Matrix visar vilka krav-id som är beroende av varandra enligt **appendix 7.1.3**. Detta innebär att för att ett krav ska kunna uppfyllas så måste eventuella andra krav redan vara uppfyllda.

5.2 Produkttestning på användare

Spelet har också testats på personer, som innan spelet fått se triaget och sedan fått en snabb förklaring att spelet går ut på att vara en ambulanssjukskötarska som ska prioritera personer. De har sedan både fått gå igenom tutorialen och spela banan. Efteråt ställdes ett antal frågor till testpersonerna:

Fråga 1: Förstår du vad som ska göras i spelet?

Fråga 2: Hur upplever du tutorialen?

Fråga 3: Är det något du inte förstår/något som är oklart?

Fråga 4: Känner du att du uppfattat essensen/syftet/huvudmekanismen av spelet?

Fråga 5: Var det svårt? (1-5 rank)

Fråga 6: Var det kul? (1-5 rank)

Fråga 7: Är kontrollerna i spelet lätta att använda?

5.2.1 Sammanfattning av testresultat

De flesta testpersoner upplever att de förstår vad som ska göras i spelet, det framgår dock att instruktionerna skulle kunna vara tydligare. I dagsläget är återkopplingen på att frigöra luftvägar att spelarens avatar sätter sig ner vid den skadade, en tanke var att göra en tydligare återkoppling på detta. När vi iakttagit testpersonerna är det tydligt att majoriteten inte förstår vilka knappar som ska hållas intryckta och vilka som endast kräver ett tryck. Detta är något som har förtydligats i tutorial.

Nästan alla upplevde att spelet var svårt. Eftersom att man manuellt måste tolka testvärdena blir de svåra att begripa, vilket leder till att flera sätter fel prioritet på de skadade. Flera önskade assistans vid tolkningen av mätvärdarna men detta är något vi ej kommer implementera i enlighet med företagets kravställning.

Vidare tycker de flesta att syftet blir någorlunda tydligt med hjälp av tutorial, ändå önskas ytterligare förklaringar på detaljnivå (till exempel hur man tolkar testvärdens). Detta är något som implementeras i tutorialen.

I stort upplevdes spelet roligt och lärorikt eftersom att det var ett nytänkande koncept. Iochmed den höga inlärningskurvan upplevde de dock att spelet kanske var riktat mot personer med kunskap inom sjukvård.

Företagskontakten hade som plan i början att utveckla spelet i två olika versioner beroende på om det var någon med kunskap inom sjukvård eller ej som spelade. Detta var ingenting vi hade tid för att utveckla, men i och med dessa åsikter från testpersonerna är detta mycket möjligt något som kan vara värt att lägga resurser på i framtiden.

6. Utvecklingens tillvägagångssätt

6.1 Arbetsmetod

Vi har saknat en förutbestämd och detaljerad kravbild från vår företagskontakt. Istället har vi fått en bred idé där vi sedan själva fått lägga mycket tid på att fastställa detaljer och spellogik. Med hjälp av uppföljande möten med vår företagskontakt där vi bollat idéer fram och tillbaka har vi under utvecklingens gång färdigställt krav och ett tydligare mål för produkten.

Arbetsmetoden har varit den agila arbetsmetoden Scrum, vilket innebär att man inleder olika långa sprintar för hur långt tid en viss uppgift inom projektet ska ta. Regelbundna möten används för att stämma av vad som gjorts och inte, samt vad som ska göras framöver under nästa bestämda tidsperiod. Beslutet att arbeta på detta sätt var ganska självklart, eftersom det kändes som en beprövad och erkänd arbetsmetod. Det är ett effektivt sätt att strukturera upp arbetet och få en överblick över vad som gjorts och inte. Längden på sprintar valde vi att anpassa efter arbetsområdets storlek, men generellt sett ganska korta sådana eftersom tätare möten, oftast redan efter någon vecka ger mer överblick och kontinuerligt arbete.

För ytterligare struktur har arbetet också fördelats på olika områden, där teamets gruppmedlemmar varit uppdelade och mer eller mindre ansvariga i olika grupper. Områdena som arbetet delats in i är användarvänlighet, spellogik, dokumentation, animering och testning. Anledningen till en indelning var delvis för att effektivisera arbetet och se till att ingen del faller genom stolarna.

6. 2 Riskanalys

Riskerna kan påverka projektet på det sätt att de kan leda till ett mer tidskrävande arbete eller på annat sätt hämma det arbete vi utför. På grund av detta har de analyserats före och under projektets gång och har till största del tagits fram för att alla i teamet ska vara medvetna om dem. Riskerna har

analyserats med avseende på risknivå och riskens konsekvenser för projektet. Nivåerna är 1-5 varav 1 är lägst risk och 5 är högst. Dessa risker samt deras rangordning finns att studera i **appendix 7.3**.

6.3 Arbetet över tid

Projektets process och de utförda arbetsmomenten har redovisats i **appendix 7.4** för att redogöra för tidsåtgången och utvecklingen. Det är en sammanställning på var vi har börjat och hur vi stegat oss fram till en slutprodukt.

6.4 Reflektion över projektet

I början av projektet var bilden av produkten väldigt oklar. I efterhand går det att inse att mycket av tiden lades på att strukturera upp och bestämma hur spelet skulle vara uppbyggt. Då ingen tidigare jobbat särskilt mycket med Unity var det också svårt att veta vad som var genomförbart och inte under den tidsramen vi hade. Det man skulle kunna säga om detta är att man kanske hade kunnat ”komma längre” med produkten om fler riktlinjer och krav på spelet varit färdigt i förväg. Samtidigt har det varit intressant och lärorikt att vara med och forma spelet mycket på vårt eget sätt med förhållandevis mycket fria händer.

Något som varit svårt var att veta hur man skulle skapa en produkt som både passar in under kategorierna roligt och lärorikt. Mycket tid har lagts på att diskutera kring vad som gör ett spel roligt, där vi alla haft olika åsikter och erfarenheter. Det var mer uppenbart vad som behövde göras för att spelet skulle uppfattas lärorikt, det vill säga att grunda spelet på det triage som företaget givit oss.

Slutligen tåls det att nämna att både projektet och kursen i sin helhet varit lärorik och gett oss alla erfarenheter att ta med oss i fortsatta studier och inte minst i framtida arbetsliv.

7. Appendix

7.1 Verifiering och validering

7.1.1 Traceability matrix

Krav	Krav_Id	Test_Id	Test-resultat	Problem
AVATAREN				
Avatar rör sig (springer och går) mha joystick	1	1	Ok	
Avataren ska ej kunna integrera med statiska objekt	22	6	Ok	
Avataren ska kunna integrera med dynamiska objekt	23	6	Ok	
När avatar kommer till skadad ska meny visas	9	4	Ok	
Går avataren i fara ska den dö, spelet avslutas	20	5	Ok	
Animationer fungerar som de ska	32	1, 3	Ok	
SKADAD				
Någon skadad kan lida av yttre blödning	4	3	Ok	
Någon skadad kan ha felaktig puls	5	3	Ok	
Någon skadad kan ha dålig kapillär återfyllnad	6	3	Ok	
Någon skadad kan ha felaktig andning	7	3	Ok	
Det finns skadade som både ligger och sitter på olycksplatsen	8	8	Ok	
Prioritet till skadad ska kunna väljas	31	3	Ok	
Alla skadade har en puls, andning, kapillär återfyllnad	27	3	Ok	
När puls kontrolleras visualiseras detta genom en pulserande bild	12	11	Ok	

När andning kontrolleras visualiseras detta genom en ”pulserande” bild	13	12	Ok	
Kapillär återfyllnad ska iscensättas genom en knapp som efter nedtryckning återfår sin färg	28	3, 16	Ok	
RÄDDNINGSLEDARE				
Räddningsledaren ger avataren information om olyckan	33	17	Ok	
Spelet går enbart att avslutas för att få poäng vid räddningsledaren	34	17	Ok	
OMGIVNING				
Miljön består av statiska objekt såsom bilar, bakgrund	14	8	Ok	
Miljön består av dynamiska objekt så som en fara, räddningsledare och skadade	15	5, 8	Ok	
MENYER				
Pausknapp ska finnas	30	15	Ok	
I huvudmenyn ska knappar finnas: starta spel, triageinformation och ljudinställningar.	24	13	Ok	
I pausmeny ska knappar finnas: avsluta, fortsätt spela, information om triage	25	13	Ok	
Menyn för skadad person existerar. Menyknappar ska bestå av andning, puls, kapillär återfyllnad, frigöra luftvägar, stoppa blödning, ge prioritet.	10	4	Ok	
Alla knappar i meny vid skadad fungerar och knapptryck registreras	16	4	Ok	
POÄNGRÄKNING				
Delpoäng ges för varje korrekt prioriterad skadad	18	9	Ok	
Tiden under spelomgången räknas	17	9	Ok	
Total poäng beräknas genom att tid och	19	9	Ok	

delpoäng för prioritering räknas samman				
Vald prioritering sparas	29	9	Ok	
ÖVRIGT				
Spelet fungerar på IOS enhet	2	2	Ej klar	Inte gått live på AppStore ännu
Spelet ska kunna avslutas på tre olika sätt. Genom att gå till räddningsledaren och avsluta, dö i en hazard eller att tiden tar slut.	3	7	Ok	
Ljud ska spelas som bakgrundsmusik	26	14	Ok	
Ljud ska kunna stängas av	11	14	Ok	

7.1.2 Test-matris

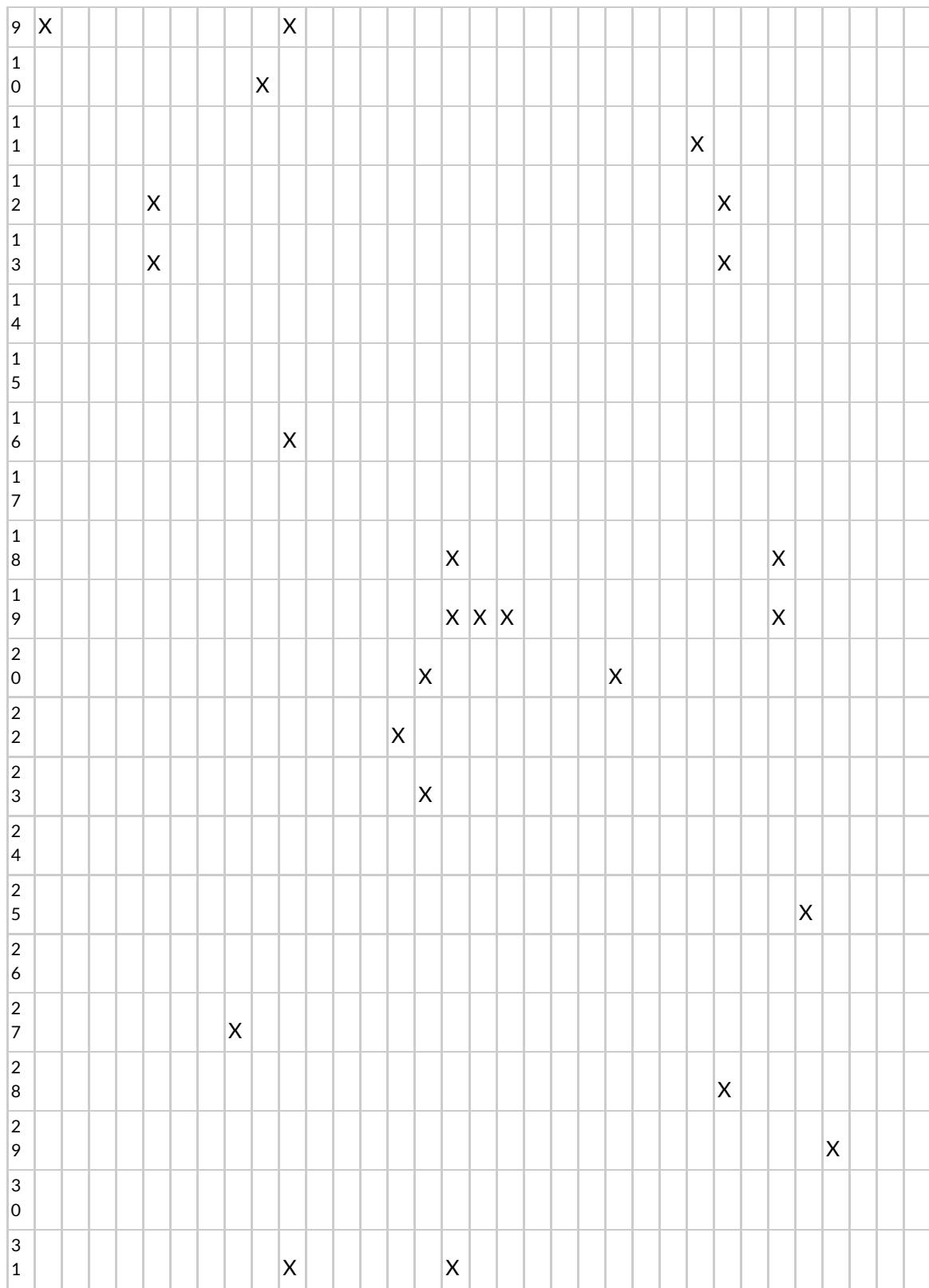
Test_Id	Krav_Id	Beskrivning
1	1, 32	Rörelse kan ske problemfritt i alla riktningar, gå och springa
2	2	Spelet körs korrekt på iOS-enhet
3	4, 5, 6, 7, 27, 28, 31, 32	Gå fram till alla skadade, identifiera åkomma och ge en prioritet.
4	9, 16, 10, 31	Meny med knappar vid skadad visas och fungerar när avataren är tillräckligt nära den skadade. Knappttryck registreras för korrekt poängräkning.
5	15, 20	Fara existerar (gasläcka) och gör att avataren dör om den "går in" i faran.
6	22, 23	Avataren integrerar korrekt med objekt, dynamiska objekt - meny kommer fram. Statiska objekt - ska inte kunna gå in i/genom dessa.
7	3	Avsluta spelet på 4 olika sätt, vid räddningsledare, genom att tiden tar slut, dö vid fara eller pausa spelet och avsluta därifrån.

8	8, 14	Gå runt i spelet och få en överblick över spelmiljön
9	17, 18, 19, 29	Poängräkning är korrekt
11	12	Kontrollera puls för skadad. Görs genom att trycka på pulsknappen i spelmenyn.
12	13	Kontrollera andning för skadad. Görs genom att trycka på andningsknappen i spelmenyn.
13	24, 25	Prova alla menyknappar i både startmeny och den meny som kommer fram vid paus.
14	26, 11	Ljud spelas oavsett handling och ska kunna stängas av.
15	30	Spelet ska kunna pausas. Testas genom att pausa under flera olika situationer i spelet.
16	28	Kontrollera kapillär återfyllnad för skadad. Görs genom att trycka på fingret en viss tid och sedan avläsa återfyllnaden.
17	33, 34	Kontrollera att räddningsledaren gör som den ska. Dvs. gå fram till räddningsledaren och prata med den samt avsluta spelet.

7.1.3 Dependency matrix

I	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3	3	3					
d	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	2	3	4	5	6	7	8	9	0	1	2	3	4			
1																																				
2																																				
3																			X																	
4																			X																	
5																			X																	
6																			X																	
7																			X																	
8																				X																

GoSaveThem
Gamification of Lifesaving Skills



7.2 Produkttestning på användare

Fråga 1: Förstår du vad som ska göras i spelet?

Svar 1:

Det är bra instruktioner men det är svårt att komma ihåg allting. Man blir lite för förvirrad när man väl kör på tid.

Svar 2: Ja typ. Man lär sig efter tiden hur man ska följa triaget, svårt att komma ihåg vad man ska göra. Bättre feedback när man frigör luftvägar, fattar inte vad som kommer hänta.

Svar 3: Ja! Som utbildningssyfte är det bra.

Svar 4: Övergripande, inte i detaljnivå.

Svar 5: Inte i detalj.

Svar 6: Syårt att förstå menyer

Svar 7: Ehhh, inte på egen hand. Tutorial hjälpte.

Fråga 2: Hur upplever du tutorialen?

Fråga 2.1: Tydligt vad nästa steg är? Flow?

Svar 1: Den var tydlig och säger vad man ska göra.

Svar 2: Ja typ. Man lär sig efter tiden hur man ska följa triaget, svårt att komma ihåg vad man ska göra. Bättre feedback när man frigör luftvägar, fattar inte vad som kommer hänta.

Svar 3: Bra. Den var tydlig, man förstod vad som skulle göras. Den var inte för lång för att man skulle tappa intresse och hoppa över allt. Det var dock stressande att tiden fortsatte ticka ner när man läste informationstexterna. Kan göra att man skippar det och tänker att man lär sig allt senare.

Svar 4: Inte helt färdigutvecklad, i övrigt bra.

Svar 5: Tutorial hade väldigt mycket text. Hade föredragit om de visades visuellt snarare än text. Det hade gett en bättre koppling till spelets scenarion. Jag trodde jag förstod innebörden efter tutorial, men det var inte helt tydligt vad jag gjorde på level 2.

Svar 6: Vi prova och få reda på hur exakt man gör mätningar.

Svar 7: Informativ, bra korta texter som förklarade vad grejer gjorde. Lite oklart vad vissa grejer var.

Fråga 3: Är det något du inte förstår/något som är oklart?

Svar 1: Svårt när man tryckte på knapparna vad som händer. Förklara i tutorialen vad som syns när man trycker på knapparna, det var svårt att förstå.

Svar 2: Tiden bör pausas när man kör tutorial. Man vill se vad som händer när man testar en knapp, ett riktigt test av varje funktion. Svårt att ta sig till räddningsledaren, highlighta den eller liknande.

Svar 3: Varför trycker man på tumnageln? Det framgår ej. Detta hade kunnat stå i tutorial.

Svar 4: Hur man använder de olika knapparna. Tydliggöra att man ska läsa av puls manuellt.

Svar 5: Instruktionerna på hur man behandlar en person blev aldrig riktigt tydligt för mig. Jag kunde se att någon blödde, men utöver det var det svårt att förstå.

Svar 6: Puls är svår att läsa av högre puls gick knappt att läsa av.

Svar 7: Bra när någon blöder, tydligt med blodet. Det som var oklart var när man var klar. Man kanske vill veta hur många det är som man ska rädda. Svårt

att få en uppfattning om banan, minikarta hade varit att föredra så att man kunde hitta alla victims och sen gå tillbaka till räddningsledare.

Fråga 4: Känner du att du uppfattat essensen/syftet/huvudmekanismen av spelet?

Svar 1: Ja, det är för att man ska lära sig prioritera för att kunna rädda så många som möjligt.

Svar 2: Ja, det skulle jag säga. Speciellt när ni förklarat lite. Man får testa sig fram flera gånger för att få bättre resultat. Information i början av vad alla olika delar av spelet är och vad som är tanken skulle vara bra. I tutorialen får man gärna berätta lite om syftet av spelet!

Svar 3: Ja! Detta har troligtvis att göra med att grundmekaniken blev förklarad tidigare, men tutorialen är tydlig för att förstå att man ska springa fram till kroppar och undersöka dem etc.

Svar 4: Ja!

Svar 5: Ja, målet är väl bara att rädda folk och avgöra om de är friska eller inte.

Svar 6: Ja förstår helhet av spelet men feedback borde ge en koppling till specifika skadade.

Svar 7: Ja. Förstår ganska snabbt att spelet handlar om att rädda människor pga spelet namn. Svårt att veta om det är för utbildningssyfte eller bara kul.

Fråga 5: Var det svårt? (1-5 rank)

Svar 1: Det var lätt att styra och man ser alla offer lätt. Det var svårt att veta vad som händer när man trycker på knapparna. Det skulle kunna var att det blir grönt när pulsen är bra etc. Svårt att räkna själv. Rank: 3.

Svar 2: Ja, svårt för förstagångsspelare (4), inte omöjligt. Styra spelet var inte svårt, att förstå syftet med spelet var lite svårt. Vissa knappar var inte jättetydliga hur man ska använda dem, främst de knappar man ska hålla in.

Svar 3: Svårt om man utgår från resultat (4). Gör man inte det så är det inte alls svårt (2), det var lätt att förstå hur man ska styra, använda knapparna etc. Lite svårt dock att sätta prioritet då det var lite oklart vart man ska trycka.

Svar 4: Ja! 4.

Svar 5: 2, eftersom det finns en timer.

Svar 6: 5, Ja

Svar 7: Stressigt.

Fråga 6: Var det kul? (1-5 rank)

Svar 1: Kul att prova på något nytt, 3.

Svar 2: Jo men det var det. 3.

Svar 3: Njae, 2. Allt är relativt.

Svar 4: 2.

Svar 5: Det var intressant, men inget jag skulle fortsätta spela. 2.

Svar 6: 4, Utmanande

Svar 7: Det var kul, kanske inte hade spelat det hela tiden men kul att testa. 3.5

Fråga 7: Är kontrollerna i spelet lätta att använda?

Svar 1: Ja!

Svar 2: Ja... de man förstår vilka det är. Hur man ska röra sig runt är bra.
Tryckknapparna är inga problem, hålla-in-knapparna är svårare.

Svar 3: Ja! Hade önskat något som gör att man kan byta riktning på kameran,
hade gjort det lättare att se den skadade bakom bilen.

Svar 4: Joystick ja! Tydlig layout. Men interaction med knapparna lite
inkonsekvent med ibland krävs intryckt och ibland bara trycka.

Svar 5: Kontrollerna är väldigt enkla. Det är snarare gameplay som är svår att
tyda.

Svar 6: Ja men menyer kan förbättras.

Svar 7: Ja det var dom. Ganska lätt. Rent intuitivt svårt att förstå "behandlings-knapparna". Bra styrförmåga, svänger inte vänster när jag svänger höger osv.

7.3 Riskanalys

Risk	Förklaring	Risknivå	Riskens konsekvenser
Produkten ska fungera på IOS system	IOS kan endast köras på iphone. Smartphone formatet innebär begränsad prestanda på grund av begränsningarna i hårdvaran. För att utveckla mot IOS och släppa produkten krävs en kostnad för ett Apple Developer Program konto.	3	5
Produktens målgrupp är stor	Med en för stor målgrupp riskerar vi att produkten inte blir anpassade för någon användare.	2	2
Stor arbetsgrupp	Många gruppmedlemmar gör den allmäna kommunikation svårare. Vid möten med alla gruppmedlemmar kan närvaren bli bristfällig vilket kan leda till stora problem att ligga i fas med projektet.	3	3
Dokumentation	Kontinuitet och struktur i dokumentationen är betydande för tydlighet.	1	2

Rapportskrivning	Viktiga punkter kan glömmas bort i den slutgiltiga rapporten.	1	3
Git och kodverifiering	<ul style="list-style-type: none"> • Begränsad kunskap och brant inlärningskurva kan skapa konflikter mellan versioner i branches och lokalt. • Funktionella krockar mellan kod skrivna av två olika personer. • Inga issues skapas då det saknas erfarenhet av att arbeta med stora projekt. 	3	5
Kund	Kunden ändrar sin version av produkten sent. I och med att kunden ej är så insatt i de tekniska aspekterna kan en lite förändring för honom göra stor skillnad för uppbyggnaden av produkten.	1	4
Saknad av kunskap	Eftersom vi är nya på att använda Unity och arbeta i ett så här stort projekt kan brist på kunskap ha stor betydelse. Det är även lätt att ta på sig för svåra funktionaliteter i och med att erfarenhet i att skapa spel saknas.	3	4

7.4 Arbetet över tid

Månad	Utfört arbete
-------	---------------

December	Project Scope skrevs och lämnades in till den 19 december.
	Arbetet planerades utefter en övergripande tidsplanering som lades upp och arbetsgrupper skapades för att strukturera det projektet kommande arbete.
	En mer utförlig kravspecifikation fastställs utefter de krav som givits från företaget, dessa krav lade sedan grunden till vad produkten skulle bli.
	Projektgruppen introducerades för spelmotorn Unity, då dencar relativt obekant för de flesta. Detta görs främst genom tutorials och är en grundläggande idé och egna mindre tester för att på ett smidigare sätt komma igång med arbetet.
	Vi beslutar att första scenen, dvs första spelbanan, ska innehålla en bilolycka då det är en situation som många kan relatera till. Beslutet grundar sig också till stor del på att material, i form av assets som skapar en verklighetstrogen stadsmiljö och tillhörande trafikolycka, fanns tillgängliga genom företaget.
	Vi beslutar oss även för att använda GitHub som verifikationsverktyg då den hade använts innan av alla gruppmedlemmar, ett annat verifikationsverktyg som vi övervägde var UnityHub. Repot sattes upp privat och användes sedan under hela utvecklingens gång.
Januari	Arbetet delades först in i två scener, i den ena scenen byggs en stadsmiljö för bilolyckan och i den andra utvecklas spellogik genom att skapa en avatar som kan röra sig.
	Avataren kan röra sig i alla riktningar med hjälp av en joystick. Detta beslut grundar sig på att en joystick tar liten plats på skärmen och ger en god spelupplevelse. En tidig idé var att använda två joysticks, en för rörelse och en för att kolla omkring, men valdes bort på grund av platsen den tog på skärmen och att det inte är helt nödvändigt för spelupplevelsen.
	Meny-scenen utvecklades och scener kopplade samman genom knappar som lade grunden för spelflödet genom hela spelappen.

	<p>Spellogiken utvecklades på det sättet att en skadad har placerats ut på skärmen, när man närmar sig denna visas en meny i högra hörnet med knappar som kan utföra olika handlingar. Placering av menyn grundar sig på att den är lätt att nå och att joysticken ligger i det vänstra hörnet.</p> <p>Animering för spelaren införs, först som en kvinnlig sjukvårdare som kan springa runt på platsen. Animeringen var väldigt viktig för att få en bra spelupplevelse, detta var självklart för projektgruppen redan innan produkten utvecklats.</p> <p>Spelaren och banan lades ihop i samma scen. Detta för att kunna arbeta så agilt som möjligt och alltid ha ett fungerande spel som kan visas för både företaget och för handledaren i projektet, samt för oss själva för att se uppfyllda krav.</p> <p>Bilder till spelet börjar ta form som en första version. Bilderna till knappar och för att kunna sätta en prioritet på en skadad tas fram, men kommer att utvecklas senare i projektets gång.</p> <p>Planering av alla kontroller som görs på den skadade påbörjas. Mer fokus läggs på hur de bäst ska implementeras för att få en så verlig upplevelse som möjligt.</p>
Februari	Parametrar till den skadade implementeras i spelet.
Mars	<p>Funktionen för att slumpa parametrar och positioner för de skadade implementeras.</p> <p>Alla funktioner som styr undersökningarna och åtgärderna för den skadade färdigställs, detta gör att vi även kan implementera dessa grafiskt.</p> <p>Inzoomning kan ske på banan med hjälp av en dubbeltryckning på skärmen. Anledningen varför detta lades till var för att man får ett bättre speldjup då det var ett problem att gå åt vissa håll när kameran var fixerad. In- och utzoomning gör att spelaren kan både se närmare på detaljer, men även en större vy av omgivningen.</p> <p>En startmeny läggs in i starten på spelet och som utvecklats från den tidigare versionen. Denna version har mer fokus på att, genom grafiken, visa spelaren dennes alternativ.</p>

	<p>En räddningsledare läggs in i spelet. De funktioner som denna är en punkt där spelaren kan få mer information, men även en avslutningspunkt. Funktionen baseras på hur verkliga situationer hanteras.</p>
	<p>Fler animationer läggs in i spelet. Detta inkluderar att spelaren sätter sig på knä för att hjälpa den skadade, att räddningsledaren vinkar för att få uppmärksamhet i spelets start och att spelaren kan variera mellan att springa och gå beroende på joysticks rörelse.</p>
	<p>Feedback på spelomgången implementeras och ges till spelaren i spelets slut. Feedbacken består av maximalt tre kommentarer om vad som ska förbättras till nästkommande spelomgång, detta implementerades för att spelaren ska kunna lära sig något till nästa spelomgång. Vi upplevde det som omotiverande att inte veta vad man gjort för fel samtidigt som r vår företagskontakt ansåg att feedbacken inte skulle vara för utförlig eftersom feedback ej är något som förekommer i verkligheten.</p>
	<p>Alla bilder till spelet är klara och läggs in i spelet, dessa används för all grafisk .</p>
	<p>Eftersom projektet och också tiden börjar närma sig sitt slut, tas ett beslut om vad det exakt är som ska produceras. Detta fastställs till en fullt spelbar bana tillsammans med en kortare tutorial. Anledningen till en tutorial, som inte var ett självklart val från början var att det framgick, bland annat från handledaren, att det var förhållandevis svårt att förstå exakt vad man som spelare ska göra i spelet.</p>
April	<p>Tutorial-banan påbörjas. Denna bana går ut på att gå igenom spelets funktioner på ett enkelt och snabbt sätt. Detta för att en spelare ska kunna få en bra överblick av spelet funktioner flr att sedan sätta sig in i spelet utan födröjning.</p>
Maj	<p>Hazard implementeras. Denna hazard består av att spelaren kan falla ner i ett hål där den är en gasläcka. Grundmekaniken i hazarden är gjord så att om spelaren går på en viss area så dör den, det blir "game over". Lokala ändringar i animationer behöver göras för att hazarden ska bli verklighetstrogen. Men själva</p>

funktionaliteten fungerar på vilken hazard som helst.
Buggar kollas över och hanteras.
Tutorial-banan färdigställs så att det går att köra från början till slut.
Spelbanan färdigställs, både omgivningen och vissa funktioner ändras smärt för att fungera och bli färdigt.
Den slutgiltliga dokumentationen av projektet lämnas in.
Ljussättning läggs till båda banorna. Detta görs sist eftersom allt annat arbete blir svårt när det är gjort och då detta tar väldigt mycket tid. "Baking", som det kallas, ger ett statiskt ljus som inte förändras vilket gör att spelet blir optimerat.
Presentation och demo av projektet utförs.