

GRUPP5

Projektplan

DigiMergo Editor

Version 0.2

Martin Bodin

2014-02-17

Status

Status	Namn	Datum
Granskad	Martin Bodin	2014-02-17
Godkänd		

Projektidentitet

Grupp 5 TDDD77

2014 VT

Linköpings tekniska högskola, IDA

Namn	Ansvar	Telefon	Mail
Jon Dybeck	Projektledare (TL), Specialist (GIT)	013-112233	jondy276@student.liu.se
Fredrik Präntare	Kvalitetssamordnare (KVA)	073-0301911	frepr183@student.liu.se
Marcus Jonsson	Utvecklingsledare (UTV)	070-2457071	marjo519@student.liu.se
Mattias Lantz Cronqvist	Testledare (TST)	070-4758115	matcr043@student.liu.se
Anders Söderström	Specialist (Användbarhet) (SPS)	070-2405100	andso217@student.liu.se
Carl Einarson	Arkitekt (ARK)	070-7292310	carei692@student.liu.se
Oscar Nöjd	Analysansvarig (ANA)	073-3108707	oscno940@student.liu.se
Martin Bodin	Dokumentansvarig (DOK), Specialist (Trac)	070-7442604	marbo0182@student.liu.se

Mailinglista för gruppen: tddd77-group5@lists.lysator.liu.se

Kund: Jonas Ryding och Magnus Bång, 581 83 LINKÖPING

Kontaktperson hos kund: Jonas Ryding

Kursansvarig: Kristian Sandahl, kristian.sandahl@liu.se

Handledare: Jonas Lindgren, 013-142231, jonas.lindgren@liu.se

Innehållsförteckning

1	BESTÄLLARE	1
2	ÖVERSIKTLIG BESKRIVNING AV PROJEKTET.....	1
2.1	SYFTE OCH MÅL.....	1
2.2	LEVERANSER	1
2.3	BEGRÄNSNINGAR.....	1
3	FASPLAN	2
3.1	FÖRE PROJEKTSTART	2
3.2	UNDER PROJEKTET	2
3.3	EFTER PROJEKTET.....	2
3.4	SEMAT ALPHA-TILLSTÅND.....	2
4	DOKUMENTPLAN	2
5	UTVECKLINGSMETODIK	3
6	UTBILDNINGSPLAN	3
6.1	EGEN UTBILDNING	3
6.2	KUNDENS UTBILDNING.....	3
7	RAPPORTERINGSPLAN	3
8	MÖTESPLAN.....	3
9	MILSTOLPAR OCH BESLUTSPUNKTER	3
9.1	MILSTOLPAR.....	3
9.2	BESLUTSPUNKTER	4
10	AKTIVITETER	4
10.1	GENERELLA AKTIVITETER	4
10.2	AKTIVITETER FÖR KURSEN	5
10.3	DIGIMERGO RESULT VISUALIZER	5
10.4	DIGIMERGO SCENARIO EDITOR	5
10.5	DIGIMERGO EXERCISE MANAGER	6
10.6	DIGIMERGO SERVER	6

10.7	DIGIMERGO SCENARIO CLIENT.....	6
11	RISKANALYS	7
12	PRIORITERINGAR	8
13	PROJEKTAVSLUT	8
14	LÖSNINGSFÖRSLAG.....	8

Dokumenthistorik

Version	Datum	Utförda ändringar	Utförda av	Granskad
0.1	2014-02-12	Första utkast av projektplan	Alla	Martin Bodin
0.2	2014-02-17	Ändringar enligt handledares kommentarer	Alla	Martin Bodin

1 Beställare

Beställare av projektet är Jonas Ryding samt Magnus Bång, Institutionen för Datavetenskap på IDA, LiTH. Kontaktinformation till beställare finns under rubriken projektidentitet i inledningen av projektplanen. Betalning för projektet sker i form av högskolepoäng vid leverans av komplett och fungerande system.

2 Översiktlig beskrivning av projektet

I detta projektet kommer en editor att utvecklas för att skapa dynamiska scenarion som kan exekveras i Emergo Train System® under en katastrofövning. Det ska även byggas vidare på delar av det existerande systemet.

Emergo Train System® är ett interaktivt pedagogiskt simuleringssystem utvecklat vid Katastrofmedicinskt Centrum (KMC) vid universitetet i Linköping. För mer information om hur ETS används, se systemets officiella hemsida: www.emergotrain.com.

2.1 Syfte och mål

Projektet utförs i inlärningsyfte för samtliga medlemmar i projektet. Det slutgiltiga målet med projektet är att skapa värde för beställaren, i form av ett fungerande system med uppfyllda krav.

2.2 Leveranser

De leveranser som ingår i projektet listas nedan:

Datum	Levererat objekt	Mottagare	Leveranssätt
Veckovis	Tidrapport	Handledare	Mail
2014-02-17	Förstudie: Projektplan, kravspec, kvalitetsplan, arkitekturbeskrivning, statusrapport, testplan	Handledare	Mail
TBD	Komplett system, dokumentation	Beställare	Fysisk leverans
TBD	Efterstudie	Examinator	Mail

2.3 Begränsningar

Projektet är begränsat av arbetstiden på totalt 250 timmar per person och det material som görs tillgängligt av beställare.

3 Fasplan

Här beskrivs översiktligt vad som sker före, under och efter projektet.

3.1 Före projektstart

Före projektstarten ska en projektgrupp bildas och ansvarsområden delats ut. Det skall även produceras tids och aktivitetsplaner samt kravspecifikation. Lösningförslag ska läggas fram på hur systemet ska designas samt hur det ska struktureras.

3.2 Under projektet

Under projektet kommer en editor att skapas. Vi kommer ha ett lunchmöte varje måndag där eventuella problem kan tas upp och på så sätt förväntas alla hålla sig uppdaterade om projektets utveckling. Om beställaren begär en statusrapport ska det skickas in. Tidsrapporten ska uppdateras kontinuerligt. Projektgruppen kommer gemensamt arbeta mot målen.

3.3 Efter projektet

Efter projektet ska gruppen skriva en efterstudie. En demonstration av den färdiga produkten för beställare ska uppvisas. Beslut om att upplösa projektgruppen tas sedan.

3.4 SEMAT Alpha-tillstånd

Se separat dokument "SEMAT Alpha - Nuvarande och planerat".

4 Dokumentplan

Nedan följer en lista på de dokument som kommer att tas fram under projektet.

Dokument	Ansvarig	Syfte	Distribueras till	Färdigdatum
Kravspec.	ANA/TL	Definierar alla krav på systemet	Beställare	2014-02-17
Kvalitetsplan	KVA	Definierar hur projektets kvalité ska upprätthållas	Handledare/Examinator	TBD
Testplan	TST	Definierar hur testning ska gå till	Handledare/Examinator	TBD
Arkitekturplan	ARK	Beskriver systemets uppbyggnad och arkitektur	Handledare/Examinator	TBD
Projektplan	TL	Definierar hur projektet ska genomföras	Beställare/Handledare/Examinator	2014-02-17
Användarmanual	DOK/Alla	Bruksanvisning på hur systemet används	Beställare/Kund	TBD
Teknisk dok.	DOK/Alla	Beskriver hur systemet är uppbyggt	Beställare/Kund	TBD
Efterstudie	DOK/Alla	Belysa erfarenheter gruppen har erhållit i projektet	Examinator	TBD

5 Utvecklingsmetodik

För att arbeta så effektivt som möjligt kommer gruppen delas upp i mindre team som kommer utveckla olika delar av systemet simultant. Dessa team ansvarar för sin egen del av systemet och för att den blir klar i tid. Om ett team känner att de har problem eller vet att de inte kommer hinna klart med sin del i tid ska detta tas upp med resten av gruppen omedelbart. Gruppen beslutar gemensamt om större beslut som berör systemet i helhet, medans mindre utvecklingsbeslut kan tas av teamen direkt.

6 Utbildningsplan

Gruppmedlemmarna och kunden behöver utbildning under respektive efter projektet.

6.1 Egen utbildning

Gruppens medlemmar kommer att utbilda sig inom de moment där kompetens inte finns genom att t.ex. utföra laborationer och lära sig nya programmeringsspråk. Bland annat kommer en kortare introduktion i Git samt Trac hållas under början av projektet. Alla gruppmedlemmar förväntas studera C# om behov finns.

6.2 Kundens utbildning

Vid projektets avslut kommer kund att utbildas inom funktionalitet hos produkten genom en användarhandledning och en demonstration av produkten.

7 Rapporteringsplan

Under projekts gång skall en tidsrapport levereras varje vecka. Varje gruppmedlem ansvarar individuellt för att fylla i sin tidsrapport efter ett arbetstillfälle. Beställaren har möjlighet att beställa en statusrapport från gruppen i vilket fall en sådan sammanställs och skickas in till beställaren.

8 Mötesplan

Gruppen ska ha ett möte en gång i veckan då alla ska närvara. Detta möte ska hållas på måndagar kl 12-13. Då projektet kommer igång så kommer gruppen delas upp i mindre team som utvecklar olika delar i systemet. Dessa team kommer vid behov ha fler möten än det på måndagar.

9 Milstolpar och beslutspunkter

Projektet har ett antal definierade milstolpar och beslutspunkter. Milstolparna fungerar som en mall för att se i vilken fas projektet befinner sig i, och om det går som planerat. Beslutspunkterna har till syfte att utvärdera projektets nuvarande tillstånd, och avgöra om projektet ska fortsätta eller läggas ner.

9.1 Milstolpar

Tabellen nedan åskådliggör gruppens milstolpar. Flera utav dessa har ännu inget fast datum specificerat, mer än att de ska åstadkommas i en viss iteration.

Nr	Beskrivning	Datum
1	Förstudien färdig	2014-02-17
2	Editor-prototyp färdig	TBD
3	Manager-prototyp färdig	TBD
4	Visualizer-prototyp färdig	TBD
5	Iteration 1 färdig	2014-03-07
6	Server API färdigt	TBD
7	Editor med grundläggande funktionalitet färdig	TBD
8	Manager med grundläggande funktionalitet färdig	TBD
9	Visualizer med grundläggande funktionalitet färdig	TBD
10	Iteration 2 färdig	2014-04-11
11	Editor färdig enl. krav	TBD
12	Manager färdig enl. krav	TBD
13	Visualizer färdig enl. krav	TBD
14	Projektleverans	TBD

9.2 Beslutspunkter

Tabellen nedan beskriver de beslutspunkter som förväntas avklaras. TBD.

Nr	Beskrivning	Datum
0	Godkännande av projektdirektiv, beslut att starta förstudie	2011-02-10
1	Godkännande av kravspecifikation, beslut att starta förberedelsefasen	2011-02-19
2	Godkännande av projektplanering, beslut att starta utförandefasen	2011-02-25
3	Godkännande av designspecifikation, beslut att fortsätta utförandefasen	2011-03-10
4	Används ej	-
5	Godkännande av produktens funktionalitet, beslut att leverera	2011-03-31
6	Godkännande av leverans, beslut att upplösa projektgruppen	2011-04-30

10 Aktiviteter

Aktiviteter har ännu inte tidsbestämts, TBD.

10.1 Generella aktiviteter

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
1.1	Skapa testplan	Utarbeta testplan enl. IEEE 730		20
1.2	Teknisk dokumentation	Skriva teknisk dokumentation		100
1.3	Protokoll	Skriva protokoll		30
1.4	Efterstudie	Skriva Efterstudie		200
1.5	Designspecifikation	Skriva designspecifikation		50
1.6	Hålla gruppmöten			120
1.7	Hålla kundmöten			25
1.8	Hålla användarmöten			20
1.9	Utföra Efrarenhetsinfångst			20

1.10	Utföra Kvalitetsarbete	25
1.11	Utföra tester	25
1.12	Utföra systemintegration	20
1.13	Utföra Demonstration	20
1.14	Utbildning C#	15
1.15	Utbildning Git	15
1.16	Utbildning DigiMergo kodbas	25
1.17	Undersöka DigiMergo API	20

10.2 Aktiviteter för kursen

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
2.1	Seminarium sälja en idé			20
2.2	Seminarium affärsplan			20
2.3	Seminarium sälja en lösning			20
2.4	Skriva projektplan			25
2.5	Skriva kravspecifikation			25
2.6	Skriva kvalitetsplan			25
2.7	Skriva arkitekturbeskrivning			5

10.3 DigiMergo Result Visualizer

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
3.1	Undersöka Användbarhet / Design			15
3.2	Implementera Excelexportering	Exportering av resultat till excelfil		15
3.3	Implementera Grafvisualisering av data		3.1	30
3.4	Implementera Server API calls/callbacks			10
3.5	Implementera GUI		3.1	30

10.4 DigiMergo Scenario Editor

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
4.1	Undersöka Användbarhet / Design			30
4.2	Undersöka Scenario filformat			15
4.3	Undersöka Resource package filformat			15
4.4	Undersöka Kartformat			20
4.5	Implementera Körtidsalgoritm		4.4	20
4.6	Implementera Resource Package "hantering"		4.3	20

4.7	Implementera kart/scenario merge	4.1, 4.2, 4.3	40
4.8	Implementera Prototyp GUI	4.1	40
4.9	Implementera manusskapning	4.2	30
4.10	Implementera exportering av manus till pdf		20
4.11	Implementera Redigering	4.8	30

10.5 DigiMergo Excercise Manager

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
5.1	Undersöka Användbarhet / Design		5.2	50
5.2	Undersöka vad excercise manager bör kunna utföra			50
5.3	Implementera Prototyp GUI		5.1	30
5.4	Implementera GUI API klisterkod		5.3	25
5.5	Implementera Server API calls/callbacks		1.17	25

10.6 DigiMergo Server

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
6.1	Implementera Ny upstartsekvens			30
6.2	Undersöka Arkitektur för events			30
6.3	Implementera Events		6.2	30
6.4	Implementera Strictness			30
6.5	Implementera DigiMergo API		1.17	25
6.6	Implementera Log Parsning för statistik		3.4, 5.5	25
6.7	Implementera Complications & Casualties			20

10.7 DigiMergo Scenario Client

Nr	Aktivitet	Beskrivning	Beroende	Beräknad tid
7.1	Omfaktorera			30
7.2	Implementera "genom vindrutan" vy			20

11 Riskanalys

Här listas de risker vi har samt vad dess allvarlighet är för projektet. För en beskrivning på hur vi kommer hantera dessa risker se rubriken "Riskhantering" i Kvalitetsplanen.

- Någon medlem i gruppen blir sjuk.
Låg risk. Man är oftast inte sjuk i mer än en vecka, och om man är någorlunda kry så kan man även programmera hemifrån om det skulle behövas. Om någon blir mer allvarligt sjuk så man blir borta fler veckor kan det bli problem, men det är inte jättestor risk att det inträffar.
- Någon medlem i gruppen hoppar av.
Hög risk. Om någon hoppar av så tappar gruppen väldigt mycket arbetstid, samt att någon/några måste sätta sig in i det arbete denna person gjorde innan han hoppade av, vilket tar ännu mer arbetstid som man ej räknat med.
- Interna konflikter, tex att vissa personer inte kommer överens.
Låg risk. Om några inte kommer överens så får man hitta ett sätt att lösa det på, vi är trots allt vuxna människor som måste kunna respektera varandra.
- Kommunikationsproblem inom gruppen.
Medel risk. Detta kan orsaka en del problem/missförstånd som kan leda till tappad arbetstid eller ge andra oönskade effekter. Så länge det upptäcks i tid är detta ingen större risk.
- Privata problem, tex familjemedlem blir sjuk.
Medel risk. Väldigt varierande risk beroende på hur allvarligt det är. Är det väldigt allvarligt så kan det leda till att en medlem måste vara borta från projektet en längre tid eller helt enkelt inte orkar fortsätta och hoppar av. Är det mindre allvarligt så är man kanske inte borta så länge från arbetet och måste man det kan man troligtvis programmera på håll.
- Vi har tagit på oss för mycket arbete.
Medel risk. Genom att planera ordentligt och gå igenom vad vi ska göra så motverkar vi denna risk så gott det går. Om det ändå skulle inträffa så får man ta upp det med kunden som förhoppningsvis förstår.
- Hårdvara går sönder.
Låg risk. Risken är inte stor att detta händer, men om det nu skulle hända så har vi så pass mycket hårdvara tillgängligt att det inte blir ett problem.
- Missuppfattning av krav.
Medel risk. Detta kan göra att vi gör något på ett sätt som inte kunden önskar, vilket betyder slösad arbetstid om man inte kan övertala kunden om att ändra på kravet.
- Aktiviteter blir ej klar i tid så vi hamnar efter i planeringen.
Hög risk. Detta kan försena hela projektet och till och med göra så vi inte blir klara med systemet i tid.
- Buggar/problem i koden som drar ut på tiden.
Hög risk. Detta kan leda till att aktiviteter ej blir klara i tid som i sin tur försenar hela projektet.
- Någon medlem inte arbetar den tid som är planerad.

Hög risk. Om någon inte arbetar den tid som är planerad så kan det leda till att aktiviteter ej blir klara i tid som i sin tur försenar hela projektet.

12 Prioriteringar

I första hand ligger fokus på krav med högst prioritet. Övriga krav eller förslag kommer prioriteras i mån av tid.

13 Projektavslut

Projektet avslutas med en demonstration av mjukvaran, och handledning på ny funktionalitet. En efterstudie skall också produceras och lämnas in. Efter godkänt projekt så tas beslut om att upplösa projektgruppen.

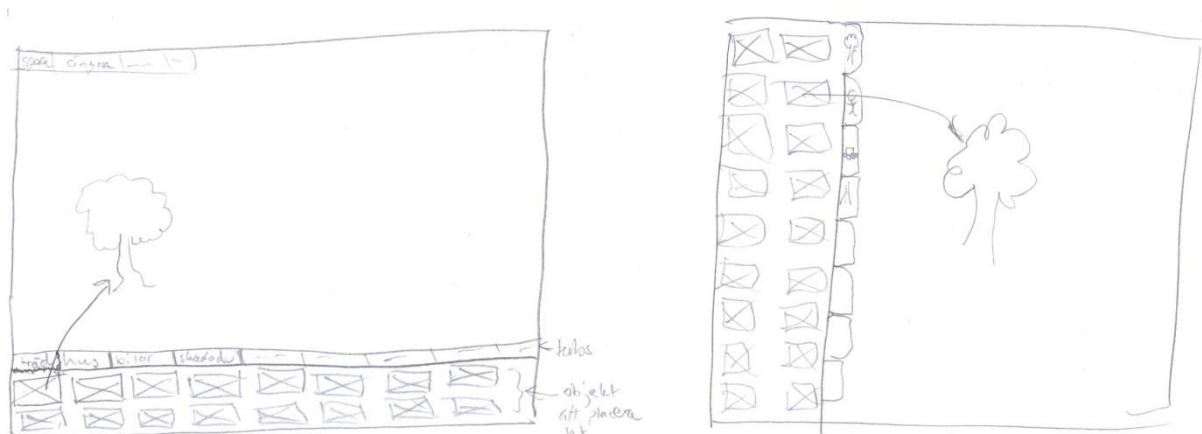
14 Lösningförslag

Nedan följer en rad förslag på hur systemets olika delar kan se ut från användarens perspektiv. Vid varje förslag står för- och nackdelar listade.

Editor

Förslag 1

Man har en panel med objekt som går att dra ut till scenariot. Objekten är sorterade i olika flikar beroende på vilken typ av objekt det är. Man kan till exempel ha en flik med fordon, en flik med vägar, en flik med träd osv. För att placera ut ett objekt tar man tag i den bilden i panelen som liknar det objektet man vill ha, sedan drar man ut det till scenariot. Panelen kan man placera på olika ställen på skärmen.



Nackdelar:

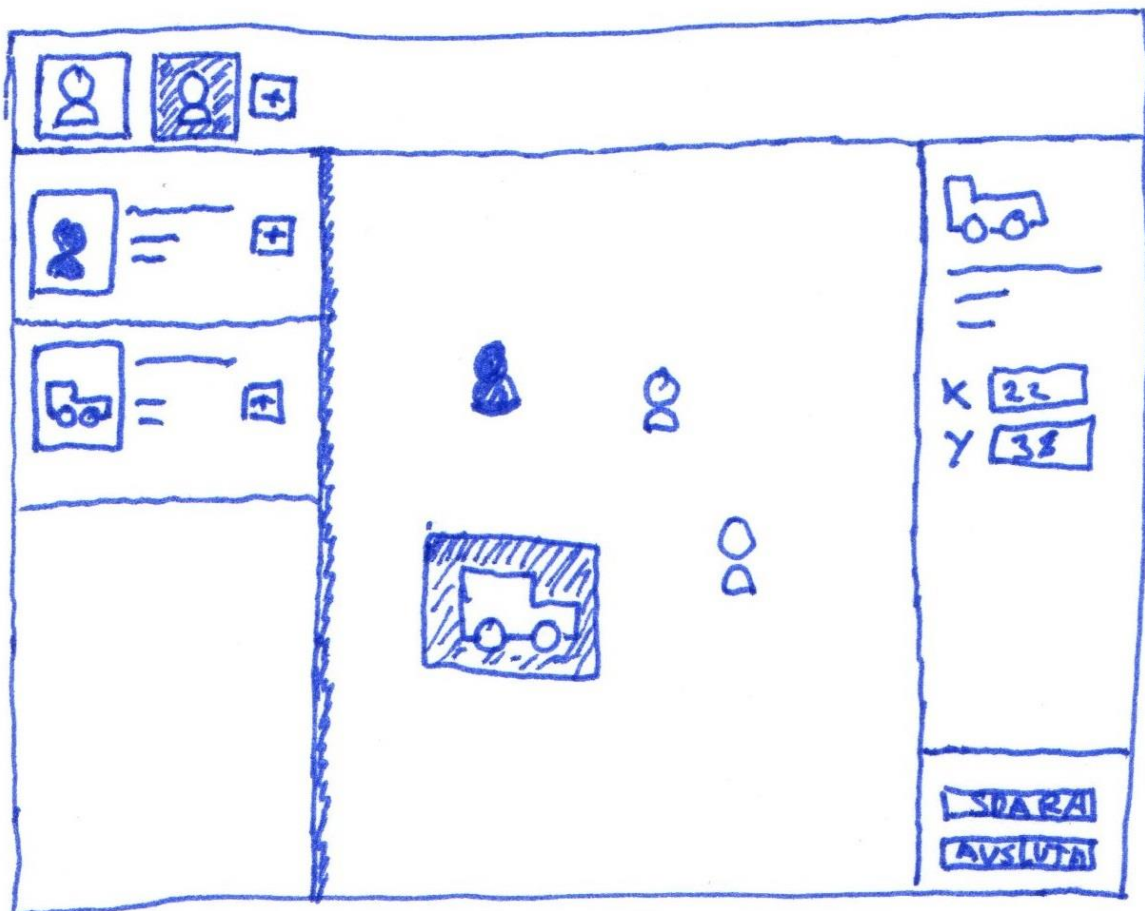
- Kan bli många objekt i en flik så man får scrolla mycket för att hitta det man vill ha.
- Panelen med objekt tar upp en stor del utrymme på skärmen.

Fördelar:

- Enkelt att navigera
- Enkelt att hitta det objekt man vill ha när allt är uppdelat i flikar

Förslag 2

Panelen överst innehåller grupper av objekt medan panelen till vänster innehåller objekt man kan lägga till scenariot. I panelen till höger kan man mata in mer detaljerad information om ett objekt såsom position eller andra attributer.



Nackdelar:

- Många paneler som tar mycket plats på skärmen.
- Inte så mycket sortering av objekt, svårt att hitta det man vill ha.

Fördelar:

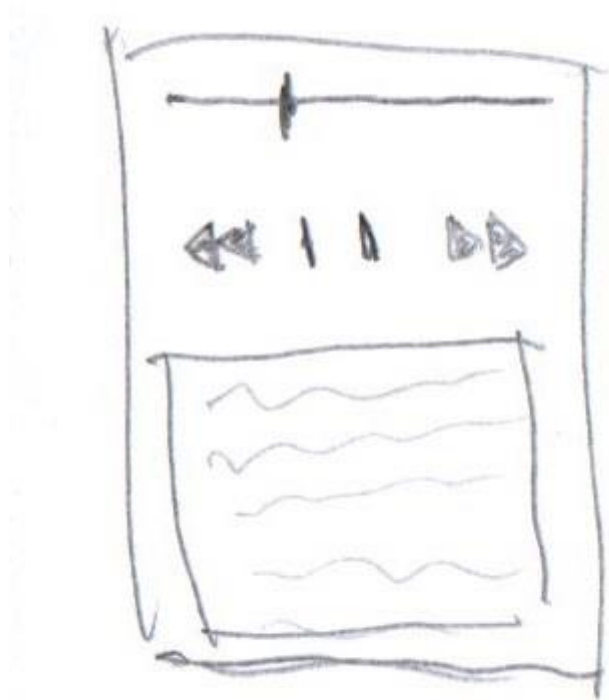
- Man kan gruppera objekt.
- Tydligt vart man anger ett objekts attributer.

Livekörning

Under körning av ett scenario ska användaren kunna pausa/starta scenariot samt ändra olika parametrar, förslagsvis via en tablet eller smartphone.

Förslag 1:

Skriva en app till ios/android/windowsphone som gör det möjligt att pausa/starta och ändra parametrar för ett scenario som körs. Appen ska också visa diverse statistik i form av grafer och tabeller.

Nackdelar:

- Mycket jobb att skriva appar till olika tablet/telefon-operativsystem
- Eventuellt krångligt att få appen att kommunicera med systemet
- Många olika enheter, olika storlekar på skärmar

Fördelar:

Enkelt att använda app.

Förslag 2:

Skriva en webbapplikation med HTML5/javascript/PHP anpassad till diverse tablets/telefoner som gör det möjligt att pausa/starta och ändra parametrar för ett scenario som körs. Man ska även kunna visa statistik i realtid i form av grafer och/eller tabeller.

Nackdelar:

- Många olika enheter, olika storlekar på skärmar.

Fördelar:

- Behöver bara anpassa det visuella till olika enheter, annan kod kan vara samma.
- HTML5/javascript/PHP, slipper native-kod
- Går att använda på alla enheter som har webbläsare och nätverksanslutning.

Resultat

Resultatskärmen kan visa olika grafer på diverse statistik, såsom hur många patienter som fick den behandling de behövde eller hur många som avled under scenariots gång. Man kan tänka sig att det ska gå att välja hur man vill se statistiken, om man bara vill se tabeller, grafer eller både och.

Här borde också finnas möjlighet att visa en "film" av scenariot, starta om scenariot, spara scenariot för att kunna visa senare eller starta ett nytt scenario.

Förslag på systemstruktur

Förslag på systemstruktur finnes i det separata dokumentet "Architecture notebook".