

# TDP005 Projekt: Oobjektorienterat system

## Kravspecifikation

Författare

Daniel Kaller, [danka340@student.liu.se](mailto:danka340@student.liu.se)  
Emil Gummus, [emigu041@student.liu.se](mailto:emigu041@student.liu.se)

# 1 Innehållsförteckning

## Innehåll

<b>1 Innehållsförteckning</b>	<b>1</b>
<b>2 Revisionshistorik</b>	<b>2</b>
<b>3 Spelidé</b>	<b>2</b>
<b>4 Målgrupp</b>	<b>2</b>
<b>5 Spelupplevelse</b>	<b>2</b>
<b>6 Spelmekanik</b>	<b>2</b>
<b>7 Regler</b>	<b>3</b>
7.1 Spelplanen . . . . .	3
7.2 Spelare . . . . .	3
7.3 Fiende . . . . .	3
<b>8 Visualisering</b>	<b>4</b>
<b>9 Kravformulering</b>	<b>7</b>
9.1 Krav . . . . .	7
9.2 Tilläggskrav . . . . .	7
<b>10 Kravuppfyllelse</b>	<b>7</b>
10.1 <i>Spelet ska simulera en värld som innehåller olika typer av objekt. Objekten ska ha olika beteenden och röra sig i världen och agera på olika sätt när de möter andra objekt. . . . .</i>	7
10.2 <i>Det måste finnas minst tre olika typer av objekt och det ska finnas flera instanser av minst två av dessa. Tex ett spelarobjekt och många instanser av två olika slags fiendeobjekt . . . . .</i>	8
10.3 <i>Ett beteende som måste finnas med är att figurerna ska röra sig över skärmen. Rörelsen kan följa ett mönster och/eller vara slumpmässig. Minst ett objekt, utöver spelaren ska ha någon typ av rörelse. . .</i>	8
10.4 <i>En figur ska styras av spelaren, antingen med tangentbordet eller med musen. Du kan även göra ett spel där man spelar två stycken genom att dela på tangentbordet (varje spelare använder olika tangenter). Då styr man var sin figur. . . . .</i>	8
10.5 <i>Grafiken ska vara tvådimensionell. . . . .</i>	8
10.6 <i>Världen (spelplanen) kan antas vara lika stor som fönstret (du kan göra en större spelplan med scrollning, men det blir lite krångligare). . . . .</i>	8
10.7 <i>Det ska finnas kollisionshantering . . . . .</i>	8
10.8 <i>Det ska vara enkelt att lägga till eller ändra banor i spelet. Detta kan exempelvis lösas genom att läsa in banor från en fil, eller genom att ha funktioner i programkoden som bygger upp en datastruktur som definierar en bana. . . . .</i>	8
10.9 <i>Spelet måste upplevas som ett sammanhängande spel som går att spela! . . . . .</i>	8

## 2 Revisionshistorik

Ver.	Revisionsbeskrivning	Datum
1.0	Första utkast	15/11/22

Tabell 1

## 3 Spelidé

Spelet kommer utspela sig in en 2D-värld där man kan förflytta sin figur i x-led och y-led. Spelaren kommer stöta på utmaningar under spelets gång iform av fiender som gör skada till spelaren om spelaren kolliderar med dem. Spelaren startar med ett x antal liv och tapper ett liv vid varje kollision med fiende eller farligt hinder". Utifrån att bara hoppa och springa så kan spelaren också byta gravitation på sin figur. Detta skapar möjligheten till många olika pussel.

Banorna avslutas då spelaren har tappa alla sina liv som kan ske på två olika sätt. Spelaren kolliderar med för mycket fiender/farliga objekt eller spelaren ramlar ut ur mappen. Banorna kan också avslutas via att nå målet vid slutet av banan. För att nå målet måste man både samla ett visst antal mynt placerade runt banan för att låsa upp dörren till målet.

## 4 Målgrupp

Spelet är en egen version av de klassiska Super Mario 2d spelen och är därför tillför de generationer som växte upp med dem spelen. Spelet är enkelt att lära sig och behöver ingen tidigare erfarenhet i datorspel för att spela det.

## 5 Spelupplevelse

Spelet kommer vara roligt och ha en hög återspelningsvärde då spelet håller koll på snabbast tid för att klara en bana skapar det en vilja att spela om banorna för att klara dem snabbare. Vid en klarad bana får man också ett antal stjärnor baserad på hur bra man klarade banan, detta skapar också ett återspelningsvärde för att hämta alla stjärnor.

## 6 Spelmekanik

Man ska kunna gå runt och byta gravitation

Tangent	Vad den gör
D	Flyttar åt höger
A	Flyttar åt vänster
Space	Spelaren hoppar
G	Spelaren byter gravitation
E	Spelaren aktiverar power-ups

Tabell 2

## 7 Regler

### 7.1 Spelplanen

- Spelplanen har en fixerad storlek
- Start position på rörliga objekt i spelplanen är slumpade
- Spelplanens statiska objekt är förbestämda
- Spelplanens golv och tak har håll
- Vid slutet av spelplanen finns ett rymdfarkost, när spelaren går in i rymdfarkosten är banan slut

### 7.2 Spelare

- Spelaren är det objekt som styrs av användaren
- Kameran är fokuserad på spelaren där spelaren alltid är i centrum av kameran
- Spelaren kan röra sig till vänster och höger
- Spelaren kan enkel hoppa
- Spelaren kan byta mellan att gå på golv och väg
- Spelaren har en väntetid efter att en ändring i gravitationen har skett
- Spelaren har x antal liv
- Vid kollision mellan spelare och fiende tappar spelaren alltid ett liv/hjärta
- Efter kollision mellan spelare och fiende har spelaren en kort tidsperiod där spelaren är immun mot all skada
- Vid kollision mellan spelare och statiska objekt stannar spelaren till
- Vid kollision mellan spelare och en power-up box ska spelaren få en slumpad power-up
- Spelaren kan bli av med sin power-up genom att använda den eller kollidera med en ny power-up box
- Spelaren kan ramla ner i håll i spelplanen
- När spelaren faller ner för ett hål tappar den alla sina hjärtan
- När spelaren har noll hjärtan kvar startar spelet om med spelaren på senaste checkpoint(första checkpointen är i början av banan)

### 7.3 Fiende

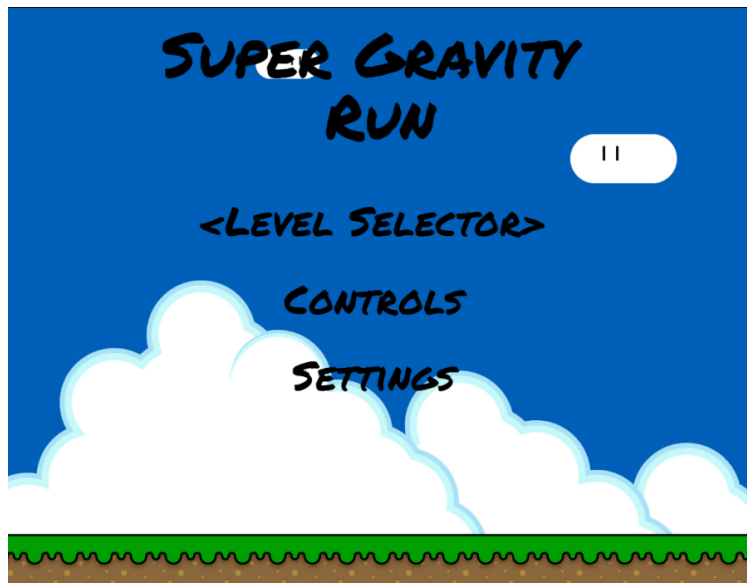
Fiender har olika egenskaper men följer till standard punkterna nedan (Vi ska skriva upp vilka fiender som finns men har inte bestämt oss om vilka som finns än)

- Fiender kan röra sig till vänster och höger, men ej styra sin riktning
- Alla fiender börjar med att röra sig framåt
- Vid kollision mellan fiende och statiskt objekt börjar fienden gå i motsvarande riktning

Spelet kommer ha ett golv och ett tak och rörliga objekt som spelare och fiender ska ej kunna gå/ramla igenom taket/golvet. Kollisioner ska finnas i spelet för att undvika att spelaren/fiender ska kunna gå igenom objekt. När fienden går in i ett objekt ska den stanna till, vända sig om och börja gå mot motsatt riktning.

## 8 Visualisering

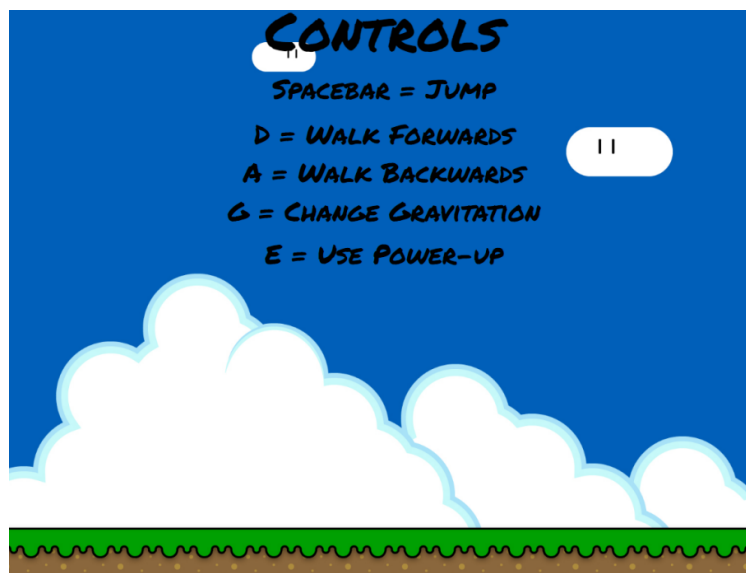
Här är fina bilder



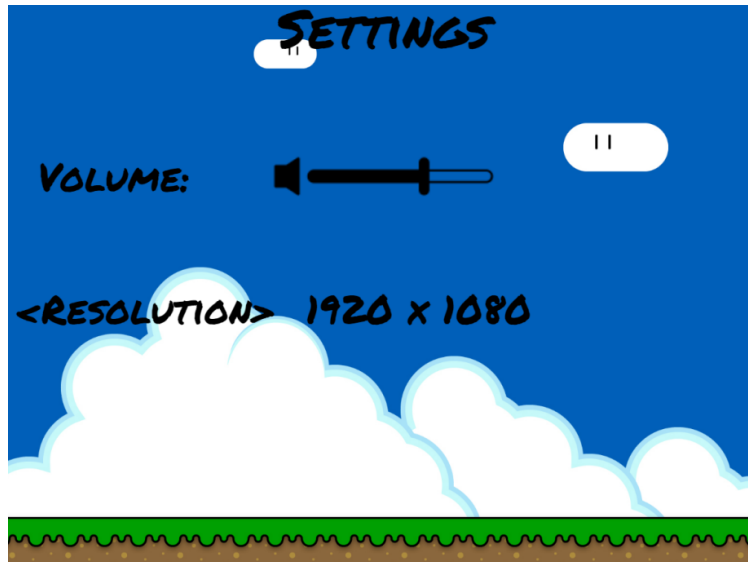
Figur 1: Huvudmeny där spelaren får välja vad den vill gå in på.



Figur 2: Level väljar meny där det visas upp rekord tiden på banan och hur många stjärnor man tjänat in.



Figur 3: Här ser spelaren kontrollerna för att spela spelet.



Figur 4: Här kan spelaren höja och sänka spelvolymen samt ändra upplösningen.



Figur 5: Här är en bild på hur ungefär spelet kommer se ut

## 9 Kravformulering

Här både kraven och tilläggskraven.

### 9.1 Krav

1. Vid start av en bana ska en tvådimensionell spelplan ritas upp i samma storlek som fönstret på applikationen
2. En spelare skall finnas
3. En typ av fiende skall finnas
4. Ett statiskt objekt (t.ex en box) skall finnas
5. Spelet ska kunna hantera kollisioner mellan olika objekt
6. Spelare och fiender ska kunna röra sig
7. Spelare ska kunna röra sig framåt och bakåt med hjälp av tangentbordet
8. Spelaren ska kunna hoppa och byta gravitationen med hjälp av tangentbordet
9. Spelare och fiender antal hjärtan/liv ska minskas vid kollision
10. Banan ska avslutas när spelaren har tagit sig till slutet av banan och gått in i rymdfarkosten

### 9.2 Tilläggskrav

11. Det skall finnas en startmeny där användaren väljer mellan Level selector, controls och settings
12. Tiden under en bana ska räknas
13. Bakgrunden på spelplanen ska ändras beroende på om spelaren går fram eller bak
14. Det ska finnas power-ups boxar
15. Spelaren ska kunna använda power-ups
16. Det ska finnas fler än en fiende
17. Sista banan ska vara en boss fight
18. Det ska finnas settingsmeny där användaren kan ändra ljudvolymen samt ändra upplösning

## 10 Kravuppfyllelse

Detta händer under spelets gång.

### 10.1 *Spelet ska simulera en värld som innehåller olika typer av objekt. Objekten ska ha olika beteenden och röra sig i världen och agera på olika sätt när de möter andra objekt.*

Uppfylls av krav: 1, 2, 3, 4, 5, 6, 7, 8, 9



**10.2** *Det måste finnas minst tre olika typer av objekt och det ska finnas flera instanser av minst två av dessa. T.ex ett spelarobjekt och många instanser av två olika slags fiendeobjekt*

Uppfylls av krav: 2, 3, 4

**10.3** *Ett beteende som måste finnas med är att figurerna ska röra sig över skärmen. Rörelsen kan följa ett mönster och/eller vara slumpmässig. Minst ett objekt, utöver spelaren ska ha någon typ av rörelse.*

Uppfylls av krav: 6

**10.4** *En figur ska styras av spelaren, antingen med tangentbordet eller med musen. Du kan även göra ett spel där man spelar två stycken genom att dela på tangentbordet (varje spelare använder olika tangenter). Då styr man var sin figur.*

Uppfylls av krav: 6, 7, 8

**10.5** *Grafiken ska vara tvådimensionell.*

Uppfylls av krav: 1

**10.6** *Världen (spelplanen) kan antas vara lika stor som fönstret (du kan göra en större spelplan med scrollning, men det blir lite krångligare).*

Uppfylls av krav: 1

**10.7** *Det ska finnas kollisionshantering*

Uppfylls av krav: 5, 6

**10.8** *Det ska vara enkelt att lägga till eller ändra banor i spelet. Detta kan exempelvis lösas genom att läsa in banor från en fil, eller genom att ha funktioner i programkoden som bygger upp en datastruktur som definierar en bana.*

Uppfylls av krav: 5, 6

**10.9** *Spelet måste upplevas som ett sammanhängande spel som går att spela!*

Uppfylls av krav: Alla krav