**Project Plan ROC ter Aa**

**Project: WireWorld**

**Client: ROC Ter Aa**

**Project Number: 000005**

**Auteur: Giel Willemsen, Guylian Gilsing**

**Date: 13-11-2019**

**Version: 0.1**

Perhaps a relevant image here

The undersigned declare their agreement with the content of this Project Plan document

**Client Project Manager**

***Initial Seen: Initial Seen:***

Date: <Enter Sign Date.> Date: <Enter Sign Date.>

Place: <Enter Sign Place.> Place: <Enter Sign Place.>

Contents

[Achtergrond 3](#_Toc24969150)

[Het Doel 3](#_Toc24969151)

[Projectactiviteiten 3](#_Toc24969152)

[Project grenzen 3](#_Toc24969153)

[Requirements 4](#_Toc24969154)

[Must 4](#_Toc24969155)

[Should 4](#_Toc24969156)

[Could 4](#_Toc24969157)

[Won’t 4](#_Toc24969158)

[Producten 5](#_Toc24969159)

[Projectorganisatie 5](#_Toc24969160)

[Planning 6](#_Toc24969161)

[Risico’s 6](#_Toc24969162)

[Bijlagen 7](#_Toc24969163)

[Programmeer standaarden 7](#_Toc24969164)

# Achtergrond

Wij wouden graag een project gaan maken met C++ en vonden het maken van graphics altijd wel interessant. Ook hadden wij wel vaker gezien dat mensen probeerde een cellular automata te maken. Wij zijn toen opzoek gegaan naar een variant die wij beide wel leuk vonden en hebben die toen gekozen. Wij vonden dit namelijk wel een goed idee om aan de hand van dit project te leren in C++ programmeren en hoe we graphics in C++ moeten doen.

# Het Doel

Ons doel is eigenlijk om gebruik te leren maken van C++ en daarbij graphics. Het product om bij dit doel te komen is om een simulatie te maken. Deze simulatie houdt in dat wij de gebruiker de mogelijkheid willen bieden om logische schakelingen te kunnen simuleren. Dit project is **GEEN** schema tool en onze intentie is ook niet om een schema tool te maken. Onze intentie is om een simulatie te maken.

Het eigenlijke doel, programmeren in C++, is dan ook ons hoofddoel. Wij gaan de simulatie game gebruiken om te leren C++ te programmeren zoals gezegd, maar ook hoe we UI’s maken en multithreading werkt in C++ naast het programmeren met graphics bibliotheken.

# Projectactiviteiten

De doelen die gesteld zijn willen wij gaan halen door een goede planning te gaan maken en ook goed overleg te houden met elkaar. Ook gaan wij een aantal tools gebruiken die het samen werken aan bijvoorbeeld broncode maar ook tegelijkertijd aan document mogelijk moet maken. Hierdoor kunnen we veel efficiënter en beter de doelen halen.

Wanneer de deadline van het project nabij nadert gaan wij nog eens goed bekijken wat we af hebben en of het voldoet aan de minimumeisen van de het project. Om te voorkomen dat we te laat uitkomen wordt de planning dan ook goed in de gaten gehouden en overlegd met dhr. S. Van Gemert, mochten we zien dat we steeds verder achter aan het lopen zijn op de planning.

# Project grenzen

Wij zijn bereid om veel te doen voor dit project, zo wordt er niet uitgesloten dat er geen nieuwe dingen geleerd gaan worden, wat wij niet bereid zijn om te doen is af te dwalen van het originele doel, of om op het laatste moment het gehele project om te gooien.

# Requirements

## Must

* Mogelijkheid tot een wereld te simuleren
  + Pauze/Start
  + Reset
  + Volgt correcte regels
    - Leeg → leeg
    - Kop → staart
    - Staart → geleider,
    - Geleider → kop als exact 1 of 2 buren koppen zijn, anders blijft het een geleider
  + Gebruik van Moore buren
* Mogelijkheid tot het maken van een wereld
  + Kan blokje veranderen tussen inert, geleider of kop

## Should

* Mogelijkheid tot een wereld simuleren
  + Kan generatie informatie zien (nummer, tijd etc.)
  + Kan wereld opslaan
  + Starten van een opgeslagen wereld
* Mogelijkheid tot het maken van een wereld
  + Mogelijkheid tot het opslaan van een gemaakte wereld
  + Mogelijkheid tot beschrijving en auteur toevoegen
* Link naar helpfile geven

## Could

* Mogelijkheid tot wereld te simuleren
  + Zelf simulatie snelheid in kunnen stellen
* Mogelijkheid tot het maken van een wereld
  + Kan drag & drop blokken toe voegen aan de wereld voor bijvoorbeeld OR of AND-schakelingen
* Mogelijkheid tot het opslaan van de wereld
  + Mogelijkheid om een templatewereld te maken
* In-program help

## Won’t

* De naam “Elektronen” gebruiken.
* Wereld simuleren
  + Volgens elektronica regels werken.
  + Natuurkunde volgen
* Niet van het projectdoel afwijken

# Producten

Aan het einde van de ontwikkel periode hebben wij een applicatie klaar die zo veel mogelijk van de requirements punten geïntegreerd heeft, tevens is dit ook wat wij willen gaan presenteren.

Om het duidelijker te zeggen hebben wij aan het einde van de ontwikkel periode een applicatie die op zijn minst de eerste punten van de MoSCoW lijst heeft geïmplementeerd (Musts) en zou dan ook eigenlijk moeten voldoen aan de Shoulds maar deze hebben een mindere prioriteit. Tijdens de presentatie is dit ook wat er gepresenteerd gaat worden.

# Projectorganisatie

In dit project zijn 3 mensen betrokken:

1. De klant, dhr. S. van Gemert.
2. De projectleider/Ontwikkelaar, Giel Willemsen.
3. Ontwikkelaar, Guylian Gilsing.

De klant (S. van Gemert) zorgt ervoor dat de applicatie naar zijn wens ontwikkeld wordt. Ook wordt er elke week (m.u.v. vakanties, examens of ziekte) een gesprek met hem gehouden, hierin bespreken wij de voortgang en eventuele problemen.

De projectleider (Giel Willemsen) zorgt ervoor dat de ontwikkeling van het project op schema loopt, en dat eventuele problemen tijdens de ontwikkel fase samen met ontwikkelaar Guylian Gilsing opgelost gaan worden. Samen met Guylian Gilsing is hij verantwoordelijk voor de realisatie van de verschillende functionaliteiten van de applicatie.

Ontwikkelaar (Guylian Gilsing) zorgt ervoor dat de verschillende functionaliteiten van de applicatie zo goed mogelijk uitgevoerd gaan worden, en volgens de documentatie uitgevoerd gaan worden. Ook moet hij eventuele problemen zo snel mogelijk doorspelen zodat er overlegd kan worden hoe dit opgelost gaat worden.

De communicatie tussen de teamleden heeft geen specifieke tijden, wel wordt er gebruik gemaakt van agile werkmethodes en gaan wij gebruikmaken van SCRUM. Dit betekent dat wij elke ochtend gaan bespreken wat er gedaan is, wat er gedaan moet worden en wat we voor die dag gaan doen.

# Planning

|  |  |  |
| --- | --- | --- |
| **Wie** | **Wat** | **Duratie (in uren)** |
| Beide | Integration testing | 12 |
| Giel | Selecteren van stukje voor de wereld als template | 2 |
| Guylian | Template importeren van wereld mogelijk maken | 4 |
| Guylian | Drag & drop blokken | 12 |
| Giel | UI control maken voor de vertraging | 2 |
| Giel | Uitzoeken hoe een vertraging te implementeren voor sim | 2 |
| Giel | Start scherm zelf maken | 3 |
| Giel | Navigatie functionaliteit maken | 1 |
| Guylian | UI maken voor de properties | 2 |
| Giel | Kijken hoe sim generation en tijd properties bij te houden en waar | 2 |
| Giel | Help file PDF openen | 2 |
| Guylian | Extra info velden voor opslag | 2 |
| Giel | Mechanisme voor het inladen van de wereld die geladen is | 2 |
| Giel | Mechanisme voor het inlezen van een opgeslagen wereld | 2 |
| Giel | Mechanisme voor het opslaan van de wereld | 3 |
| Giel | Global editor UI maken | 4 |
| Guylian | Cellen plaatsen/veranderen UI | 4 |
| Guylian | Cellen plaatsen/veranderen functionaliteit | 3 |
| Guylian | Verplaatsen van de viewport mogelijk maken | 3 |
| Giel | Functionaliteit van de eigenlijk knoppen maken | 3 |
| Giel | De layout maken voor de UI van de basis simulator | 3 |
| Guylian | Cellen render code | 2 |
| Guylian | Grid systeem maken | 6 |
| Giel | Multithreading implementeren voor het update | 3 |
| Giel | Regels implementeren voor update van Gen | 4 |
| Giel | Cell base class maken + eigenschappen wereld | 1 |
| Beide | Instellen Libraries | 4 |

# Risico’s

Tijdens het plannen van WireWorld zijn een aantal risico’s naar boven gekomen die de realisatie kunnen belemmeren.

Zo lopen wij het risico dat het realiseren van specifieke functionaliteiten veel langer kan duren dan eigenlijk nodig is. Dit heeft te maken met de kennis die wij hebben van C++ en OpenGL, dat is namelijk niet veel. Dit kan opgelost kunnen worden door buiten project uren om meer ervaring met C++ en OpenGL op te doen dit moet dan individueel gedaan worden.

Het managen van het computergeheugen kan ook een probleem zijn. C++ is een zogenoemde “low level language”, dit betekent dat je in de taal meer controle hebt over het geheugen verbruik van de computer. Als wij tijdens het programmeren niet goed opletten hoe wij geheugen gebruiken, dan kunnen er crashes ontstaan, of dat de applicatie de computer langzamer laat lopen. Dit kan tegen gegaan worden door zo min mogelijk ‘new’ gebruiken en extra vaak elkaars code te checken.

# Bijlagen

## Programmeer standaarden

Tijdens de realisatie van dit project gebruiken wij een handvol standaarden om de code zo overzichtelijk mogelijk te houden.

* Alle functie argument gebruiken de “a\_” prefix
* Alle methode variabele gebruiken de “m\_” prefix
* Alle functies en methodes gebruiken Pascal casing
* Alle variabele en argumenten gebruiken camel casing
* Alle classes gebruiken Pascal casing
* Geen C++ code in header(.h) files