**Projectplan**

**Persistent Seas**

Datum: 23/05/2017

Versie: V1.1

Door: Nordin Jansen & Jordy Rutjens

Klas: IC.16AO.a

Begeleidend docent: DRJO

|  |  |  |
| --- | --- | --- |
| Naam: DRJO | Naam: DRJO | Naam: Nordin Jansen & Jordy Rutjens |
| Functie: Docent | Functie: Docent | Functie: Student |
| Datum: 09/05/2017 | Datum: 09/05/2017 | Datum: 09/05/2017 |
| Goedgekeurd (handtekening) | Goedgekeurd (handtekening) | Goedgekeurd (handtekening) |

**Goedkeuring**

Inhoud

[Inleiding 3](#_Toc483295083)

[Opdracht 3](#_Toc483295084)

[Doelstelling 4](#_Toc483295085)

[Afspraken 4](#_Toc483295086)

[Grenzen 4](#_Toc483295087)

[Plan van Aanpak 4](#_Toc483295088)

[Eisen en planning (WorkBreakDown) 5](#_Toc483295089)

[Inleiding 5](#_Toc483295090)

[Eisen 5](#_Toc483295091)

[Beslissingen 5](#_Toc483295092)

[Afspraken 5](#_Toc483295093)

[Grenzen 6](#_Toc483295094)

[Plan van Aanpak 6](#_Toc483295095)

[Haalbaarheidsstappen 6](#_Toc483295096)

[Business perspectief 6](#_Toc483295097)

[Huidige situatie 6](#_Toc483295098)

[Gewenste situatie 6](#_Toc483295099)

[Business requirements 6](#_Toc483295100)

[Gebruikersperspectief 7](#_Toc483295101)

[Use cases, scenarios en activity diagrams 7](#_Toc483295102)

[Traceability matrix 9](#_Toc483295103)

[Wire Frames 10](#_Toc483295104)

[Traceability matrix 10](#_Toc483295105)

[Over dit document 10](#_Toc483295106)

[Afkortingen 10](#_Toc483295107)

[Referenties 10](#_Toc483295108)

[Definities 10](#_Toc483295109)

[Gebruikte materialen 11](#_Toc483295110)

[Bijlagen 11](#_Toc483295111)

# Inleiding

In dit projectplan wordt verteld en informeren over ons project voor de proftaak van 2017. Het is de bedoeling dat na het lezen van dit projectplan duidelijk is wat er gaat gebeuren.

# Opdracht

Dit project is een open-world game. Het is de bedoeling dat mensen/spelers vrij kunnen rondlopen in een wereld waar verschillende eilanden zijn en meerdere dorpen/forten voor mensen om te kunnen handelen. Het speelt zich af in de gouden eeuw. Verschillende functies die erin komen zijn:

* Boten
  + Hiermee kun je rondvaren en de wereld ontdekken.
* Materialen verzamelen
  + Er zijn mijnen en bossen waar men materialen kunnen verzamelen
* Materialen verkopen
  + Hierdoor kan je geld verdienen door verzamelde materialen te verkopen
* Winkels
  + Hier kan men boten, voedsel en water kopen
* Grond kopen
  + Er zijn stukken grond met huizen erop die als een huis kunnen dienen voor de speler
* Geldeenheid
  + Hiermee kunnen spelers grond of voedsel kopen
* Kisten met materialen
  + Deze zijn verspreid over de wereld en zijn zeldzaam
* Dag/nacht cyclus
  + Hiermee het dag en nacht gemaakt in de game
* Honger en dorst systeem
  + Dit zorgt ervoor dat je voedsel en water moet vinden of kopen
* Inventaris
  + Hier zie je alles wat je bijhebt
* Wapens
  + Hiermee kun je meedoen in PvP
* Accountsysteem
  + Dit betekent dat je een account nodig hebt om te kunnen spelen
* Arduino controller
  + Dit is een hardware controller waar twee joysticks opzitten en knoppen/displays opzitten

# Doelstelling

Met dit project worden verschillende doelen bereikt. Nordin wil namelijk graag meer over Arduino en de connectie met C# leren. Jordy wil meer leren over Unity en networking via Unet. En uiteraard is er een werkende game.

# Afspraken

Algemeen

* Eerst wordt er aan het singleplayer script gewerkt en dan wordt het omgezet naar een multiplayer script
* Het ES-gedeelte speelt een belangrijke rol in de must

Nordin

* Focust vooral op de Arduino controller
* Maakt af en toe Unity scripts

Jordy

* Focust vooral op Unity scripts
* Maakt af en toe ook voorwerpen voor in de wereld via blender

# Grenzen

Do’s

* NPC’s die dienen als verkopers en kopers voor spelers.
* Verschillende dorpen/forten verspreidt over de eilanden.
* Stukken grond waar huizen op staan waar ook een stuk land bij ligt om gewassen te planten
* Uitgebreide Arduino controller is belangrijk in dit project.

Don’ts

* Character customization
* Verschillende seizoenen
* Natuurkundige factoren (wind, temperatuur)
* Na de oplevering wordt dit project niet onderhouden

# Plan van Aanpak

Als eerst gingen wij een concept bedenken van het design van de game. Zonder dit concept kunnen wij niet verder aangezien we dan geen rekening kunnen houden met elkaar. Nadat dit klaar is gaan we programmeren. Eerst zorgen we ervoor dat het lopen en interactie systeem goed is opgebouwd. Dan begint Jordy met het maken van het honger en dorst systeem. Nordin is dan ook al begonnen aan de Arduino controller om een object te laten bewegen net als de player beweegt. Op dit punt is een groot deel van de basis klaar. Nu begint Jordy aan het verzamelen van materialen en het toevoegen van een boot. Nordin begint dan aan het verbeteren van de Arduino controller met een camera functie. Nu dit klaar is begint Jordy aan de NPC’s en Nordin helpt hier ook aan mee. Uiteindelijk wordt alles gefinetuned en worden er extra’s (shoulds en coulds) toegevoegd.

# Eisen en planning (WorkBreakDown)

# Inleiding

Dit is het technisch ontwerp voor ons project “Persistent Seas” wat een open-world multplayer game is. In dit verslag uitleggen hoe alles in elkaar zit en werkt. Je hebt een doel, maar die is redelijk breed omdat je kan doen wat je zelf wil.

# Eisen

Het gaat een openworld multiplayer game worden, waarbij er meerdere dingen om te doen zijn en een doel in het spel is. Alles zou redelijk snel moeten gaan omdat er veel clients verbonden zijn dus het moet goed “gesynced” zijn zodat niemand achterloopt op de server. De programmeertaal die Unity gebruikt is C# maar wij voegen ook Arduino toe en dat is C. Er is ook een GUI om de player te laten zien wat er gebeurd, wat zijn inventory is en wat de status van zijn wapen is. Ook laat het zien hoeveel “Hunger & Thirst” je hebt, zodat je weet wanneer je weer iets moet eten.

# Beslissingen

Alle dingen die er zijn in de game zijn gefocused op makkelijk leesbaare code zodat je na meerdere script alsnog ziet wat iets doet en hoe het werkt. Sommige dingen moesten aangepast worden omdat het te moeilijk was en dus hebben we een makkelijkere oplosssing hadden toegevoegt.

# Afspraken

Materiaal en Code word ingeleverd door Jordy. Nordin doet de Arduino waar Jordy niks aan gaat doen, en Nordin gaat niet zelf veel script schrijven in Unity. Maar natuurlijk wel Arduino script toevoegen aan Unity zodat ze in het spel werken. We hebben dus een strikte planning zodat we geen tijd verliezen, Jordy praat met de Opdrachtgever wanneer er veranderingen moeten worden gedaan, of als er vragen zijn. Ook zorgen dat we beiden een bestand hebben zodat we niks kwijtraken.

# Grenzen

In-game trading gaat niet toegevoegd worden, de functie wat het meeste dichtbij komt is dat je gold (een vorm van geld) kan laten vallen. In-game trading is namelijk iets waarbij we niet veel verstand van hebben en beter kunnen negeren.

Grote servers met veel mensen ( c.a 25+/- mensen focusen we op ) is te hoog gezocht, want als we zo’n eigen server willen moeten we veel scripts toevoegen aan networking wat te moeilijk is en dan moesten we natuurlijk ook een Server moeten hosten, en dat hebben we niet.

# Plan van Aanpak

Een paar Showstoppers zijn als mutiplayer en openworld ontabreekt, het is namelijk iets wat we zeker in het spel willen hebben. Als we dat missen ontbreekt het hoofddoel, dus eerst gaan we alles klein testen om te zien of het allemaal werkt en werkt zoals wij het gepland hadden. Als het niet helemaal werkt, passen we het aan of gebruiken we verschillende technieken zodat we het alsnog kunnen maken.

## 

## Haalbaarheidsstappen

Eerst willen we dingen toevoegen zodat er iets te doen is in het spel, en een doel toevoegen, dus geld, huizen, spullen, of andere dingen die je kan kopen door geld te verzamelen. Ook een arduino controller toevoegen is belangrijk, en is iets wat er zeker inmoet. Daarna gaan we de minder belangrijke dingen toevoegen die de game alleen maar beter maken en het dus ook een echte game maakt. Helemaal op het einde, als we nog tijd overhebben kunnen we proberen om grotere servers te krijgen en overtestappen naar een andere networking framework zodat we naar een “MMO” kunnen gaan ( Dus rond de 100 mensen op een server)

# Business perspectief

## Huidige situatie

Dit project is niet ontstaan uit een specifiek probleem, maar is bedoeld om te kijken of het gewenst is door gamers.

## Gewenste situatie

Dat het project gewenst wordt door gamers, waardoor ze het gaan spelen

## Business requirements

De opdrachtgever wil:

* Dat de functies eerst als SP werken en als dit werkt het wordt aangepast zodat het als MP werkt.
* Dat Jordy zich vooral bezighoudt met Unity scripts
* Dat Nordin zich vooral bezighoudt met Arduino scripts

# Gebruikersperspectief

## Use cases, scenarios en activity diagrams

|  |  |
| --- | --- |
|  | |
| Naam | Server joinen |
| Actor | Speler |
| Pre | Actor moet ingelogd zijn |
| Beschrijving | De actor klikt op een server, daarna klikt hij op connect |
| Alternatief | Actor kan geen verbinding met server maken |
| Resultaat | De Actor kan de server joinen |

|  |  |
| --- | --- |
|  | |
| Naam | Materialen verkopen |
| Actor | Speler |
| Pre | Actor moet een server gejoind zijn. |
| Beschrijving | De actor kan bij een verkoper zijn materialen verkopen. |
| Alternatief | De actor beschikt niet over de benodigde materialen. |
| Resultaat | De actor krijgt gouden munten voor zijn materialen. |

|  |  |
| --- | --- |
|  | |
| Naam | Boten/varen |
| Actor | Speler |
| Pre | Actor moet een server gejoind zijn. |
| Beschrijving | De actor koopt een boot waarna hij deze kan plaatsen in haven en besturen. Als hij de boot wil verlaten moet hij bij een haven aanwezig zijn. |
| Alternatief | De actor beschikt niet over de gouden munten die nodig zijn op een boot te kopen, de actor heeft geen boot om te plaatsen, de actor is niet aanwezig bij een haven om zijn boot te verlaten. |
| Resultaat | De Actor kan een boot kopen, ermee varen en verlaten |

## Traceability matrix

|  |  |
| --- | --- |
| Use cases | Business requirements |
|  | Server joinen is nodig om met andere mensen te spelen. |
|  | Materialen verkopen is nodig om geld te verdienen om je speelervaring uit te breiden. |
|  | Boten/varen is nodig om naar andere eilanden toe te gaan. |

# Wire Frames

## Traceability matrix

Matrix waarbij use cases aan de verschillende forms(b.v.) zijn gekoppeld1

# Over dit document

## Afkortingen

|  |  |
| --- | --- |
| Afkorting | Omschrijving |
| NPC | Non Player Charachters |
| PvP | Player versus Player |
| ES | Embedded Systems |
| MP | Multiplayer |
| SP | Singleplayer |
| Njan | Nordin Jansen |
| JRut | Jordy Rutjens |
| MMO | Massive Multiplayer Online |

## Referenties

|  |  |
| --- | --- |
| Referentie | Omschrijving |
| Project Plan | Het plan hoe wij het project aanpakken |
| Functioneel Ontwerp | Een uitleg die voor de “normale” persoon te snappen is. |

## Definities

|  |  |
| --- | --- |
| Definitie | Omschrijving |
| Showstopper | Technisch onderdeel van een ontwerp dat er voor kan zorgen dat de totale haalbaar heid van het project in gevaar komt. Het is belangrijk dat de eerst de showstoppers geïdentificeerd worden en getest worden.  Voorbeeld: als een apparaat bestuurd moet worden vanuit een PC maar de communicatie werkt niet tussen PC en apparaat, zal het dus nooit vanuit een PC bestuurd kunnen worden. Oplossing: eerst dit testen, dan pas naar de volgende stap. |
|  |  |

## Gebruikte materialen

|  |  |
| --- | --- |
| Onderdeel | Omschrijving |
| Arduino | Hardware die voor besturing in de game kan zorgen |
| Unity | De software waar de game in wordt gemaakt |
| MS Word 2016 | Documentatie |
| MS Excel 2016 | WBS en MoSCoW |
| Blender | Voor game models maken |

# Bijlagen