**Technisch Ontwerp**

**Persistent Seas**

datum: 16-05-2017

Versie: V1.0

door: Jordy Rutjens & Nordin Jansen

klas: 16AO.a

Begeleidend docent: DRJO

|  |  |  |
| --- | --- | --- |
| Naam: Joris | Naam: Joris | Naam: JRut & Njan |
| Functie: | Functie: | Functie: |
| Datum: | Datum: | Datum: |
| Goedgekeurd (handtekening) | Goedgekeurd (handtekening) | Goedgekeurd (handtekening) |

**Goedkeuring**

Inhoud

[Inleiding 2](#_Toc434054834)

[Eisen 2](#_Toc434054835)

[Beslissingen 2](#_Toc434054836)

[Afspraken 2](#_Toc434054837)

[Grenzen 2](#_Toc434054838)

[Plan van Aanpak 2](#_Toc434054839)

[Haalbaarheidsstappen 2](#_Toc434054840)

[Over dit document 2](#_Toc434054841)

[Afkortingen 2](#_Toc434054842)

[Referenties 2](#_Toc434054843)

[Definities 2](#_Toc434054844)

[Gebruikte materialen 2](#_Toc434054845)

[Bijlagen 2](#_Toc434054846)

Informatie voor de student:

De rode tekst dient ter informatie. Verwijder deze rode tekst, pas deze aan of vervang deze voor zwarte tekst.

WEL DOEN

* Minimaal xx woorden, maximaal xx woorden; indien niet wordt voldaan aan deze eisen wordt het niet nagekeken tenzij er een duidelijke verklaring is voor afwijken. Indien er afgeweken moet worden aan de eisen dient dit eerst te zijn besproken en goedgekeurd zijn door de docent.
* Volgende hoofstuk op dezelfde pagina als voorgaande, tenzij er een totaal ander onderwerp besproken wordt
* Voor iedere stijl hetzelfde lettertype en grootte (gebruik de stijlen van MS Word)
* Maximaal drie lagen in hoofdstukindelingen; indien er meer scheiding in teksten nodig is moeten kopjes in een normaal, maar vet, lettertype worden gebruikt

NIET DOEN

* Elk hoofdstuk op een nieuwe pagina
* Tekst typen om het aantal woorden te bereiken

EN OOK..

* Evalueren van de geplande vs gebruikte tijd van de realisatie en implementatie

# 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 wilt.

# 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

Beschrijf hier een inleiding van de stappen van de oplossing:

* Zorg ervoor dat de Showstoppers in beeld zijn.
* Zorg ervoor dat beschreven is dat er een haalbaarheidsonderzoek aan ten grondslag ligt, ofwel, zorg ervoor dat de zaken die de totale haalbaarheid in gevaar brengen EERST getest worden

## 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)

# Over dit document

## Afkortingen

|  |  |
| --- | --- |
| Afkorting | Omschrijving |
| JRut/Njan | Jordy Rutjens / Nordin Jansen |
| MMO | Massive Multiplayer Online |

## Referenties

|  |  |
| --- | --- |
| Referentie | Omschrijving |
|  | Voorbeelden: ‘bovenliggende’ documenten (ProjectPlan, Functioneel Ontwerp, etc.) |
|  |  |

## 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 | Arduino voor de Arduino controlls in Unity |
|  |  |

# Bijlagen