**Technisch ontwerp Copodi Gaming™**

**Project: Protask RPG**

**Opdrachtgever: Stijn van Gemert**

**Projectnummer: 1**

**Auteur: Peter Janssen & Alexander Deelen**

**Datum: 26-11-2019**

**Versie: 1**

Ondergetekenden verklaren zich akkoord met de inhoud van dit technisch ontwerp.

**Opdrachtgever Projectmanager**

***Paraaf gezien: Paraaf gezien:***

Datum: <Geef de datum op.> Datum: <Geef de datum op.>

Plaats: <Geef de plaats op.> Plaats: <Geef de plaats op.>

Inhoudsopgave

[Inleiding 3](#_Toc499208813)

[Technische Eisen 3](#_Toc499208814)

[Componenten en modules 3](#_Toc499208815)

[Diagrammen 3](#_Toc499208816)

[Klassen diagram 3](#_Toc499208817)

[Activiteiten diagram 3](#_Toc499208818)

[Sequentie diagram 3](#_Toc499208819)

[Beslissingen 3](#_Toc499208820)

[Afspraken 3](#_Toc499208821)

[Grenzen 4](#_Toc499208822)

[Plan van Aanpak 4](#_Toc499208823)

[Haalbaarheidsstappen 4](#_Toc499208824)

[Test plan 4](#_Toc499208825)

[Over dit document 4](#_Toc499208826)

[Afkortingen 4](#_Toc499208827)

[Referenties 4](#_Toc499208828)

[Definities 4](#_Toc499208829)

[Gebruikte materialen 5](#_Toc499208830)

[BIJLAGEN 5](#_Toc499208831)

[BIJLAGE B Accordering 6](#_Toc499208832)

# Inleiding

Dit technisch ontwerp is voor Protask RPG, dit is een videogame gemaakt in de Unity engine. Deze game bevat een automatisch gegenereerde wereld met veel customizability.

In dit document worden de technische aspecten uitgelegd over de game. Er wordt van de lezer verwacht dat ze enige kennis hebben van technische beschrijving en videogames.

# Technische Eisen

De game moet gaan werken zoals je van een RPG (Roleplaying Game) zou verwachten. Je zit in een wereld die je kunt verkennen hoever jij wilt. Tijdens het spelen kun je tegenstanders tegenkomen en met die tegenstanders een gevecht aangaan. Dit gevecht speelt zich af in beurten en het doel is om je tegenstander zonder levenspunten te krijgen.

De game moet zich snel weergeven, als er op een knop gedrukt wordt moet er meteen actie komen. Dit is vrij gemakkelijk, omdat de Unity engine in combinatie met C# goed is geoptimaliseerd en snel acties kan doorvoeren. Ook kan het gemakkelijk een Graphical User Interface tonen voor de gebruiker om alles duidelijk te maken.

# Componenten en modules

**Unity Engine**

De Unity Engine bevat veel van onze benodigdheden. Het bestuurd de hele game door een main loop die er tegelijkertijd voor zorgt dat elk element een Update functie aanroept. Het bevat genoeg elementen zoals hun eigen GUI componenten en sprite renderers, ook kan het gemakkelijk audio afspelen.

**Input System**

Unity heeft een gemakkelijk systeem om de input van een gebruiker aan te sturen. Zowel een keyboard en muis als een gamecontroller kunnen gemakkelijk worden herkent en aangestuurd.

# Diagrammen

## Klassen diagram

* Teken het klassen diagram van alle klassen, attributen en onderlinge relaties, dat bij je applicaties hoort
* Maak een klassen tabel met uitleg over wat een klasse precies doet

## Activiteiten diagram

* Teken het activiteiten diagram (flow diagram) dat bij je applicatie hoort

## Sequentie diagram

* Teken hoe de verschillende klassen moeten samen werken

# Beslissingen

Wij gebruiken voor dit project de Unity engine omdat deze gebruik maakt van C#, een programmeertaal waar wij al meer ervaring mee hebben. Ook is het tijdbesparend omdat het ontwikkelen van een eigen game engine meer tijd zou kosten dan de huidige scope van dit project.

# Afspraken

INSERT PLANNING HIERO

# Grenzen

Wij gaan geen gebruik maken van elementen en files die Unity niet ondersteund, het is niet mogelijk om sommige filetypes te gebruiken voor bepaalde onderdelen.

# Plan van Aanpak

Het project kan niet verdergaan als bepaalde onderdelen niet gerealiseerd kunnen worden. Als de procedural generation voor de speelwereld niet mogelijk is kan het originele idee niet doorgaan.

Er wordt eerst gekeken of de aangegeven hindernissen haalbaar zijn, als deze geen probleem schijnen te zijn kunnen we gewoon doorgaan volgens de planning.

## Haalbaarheidsstappen

* Beschrijf hier in een tabel welke stappen worden genomen in volgorde van afhankelijkheid.
* Voorbeeld: eerst testen of er communicatie is tussen PC en apparaat, dan pas complexe communicatieprotocollen ontwerpen.

# Test plan

* Geef aan wanneer en hoe je het systeem gaat testen.
* Dit betekent dat je dit in detail moet weergeven met test invoer en verwachte uitvoer.
* Je dient dit voorgaande het coderen van het systeem al aan te geven.
* Aan het einde van de rit dienen de resultaten van het testen in een test rapport te worden vervat en als onderdeel van de documentatie te worden meegeleverd.

# Over dit document

## Afkortingen

|  |  |
| --- | --- |
| Afkorting | Omschrijving |
|  | Plaats hier in dit document gebruikte afkortingen en de betekenis |
|  |  |

## 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 |
|  | Voorbeelden: hardware (elektronica, arduino, PC-type), software (IDE, DLL’s) |
|  |  |

# BIJLAGEN

[BIJLAGE B Accordering 6](#_Toc499208804)

BIJLAGE A Protocol van acceptatie

De punten zoals beschreven zijn in de onderstaande tabel opgenomen. Tijdens de oplevering zal op deze punten worden gecontroleerd.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Acceptatiepunt** | **Akkoord** | | **Opmerkingen** |
|  |  | **wel** | **niet** |  |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |
|  | <Geef het acceptatiepunt op.> |  |  | <Geef opmerkingen.> |

**Overige opmerking:**

|  |
| --- |
| <Geef eventueel opmerkingen.> |
|  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gezien namens: |  |  | Paraaf gezien: | Datum gezien: |
| Projectmanager | : | <Geef de eigen bedrijfsnaam op.> |  | <Geef de datum op.> |
| Opdrachtgever | : | <Geef de bedrijfsnaam van de opdrachtgever op.> |  | <Geef de datum op.> |

BIJLAGE B Accordering

Wanneer u akkoord gaat met de inhoud van dit technisch ontwerp, dan verzoeken wij u dit projectdocument ondertekend te retourneren.