

# Documentatie

## GDV2, Kernvak 4

### Jeffrey Saydam

#### Concept

**Genre:** 2D Turn-based Cards.

**Play mode:** 1vs1.

Er zijn twee facties die tegen elkaar moeten battelen d.m.v. de vier elementen.

#### Facties:

1. The Light (Solaria)
2. The Darkness (Nesol)

#### Flow

1. De speler moet eerst een account aanmaken.
2. De speler moet inloggen.
3. De speler zoekt een game.
4. Twee spelers van elk een andere factie vormen samen een sessie.
5. Elke speler legt omstebeurd een kaart neer.
6. De kaarten vormen samen een uitkomst.
7. Beiden spelers krijgen een nieuwe kaart.
8. De spelers spelen door tot één van hen dood is.
9. Alleen de winner krijgt ranked points.
10. De speler kan weer een nieuwe game zoeken.

#### Kaarten combinaties

##### **Vuur | Aarde | Water | Lucht**

Vuur + Vuur = Beiden krijgen damage.

Vuur + Aarde = Aarde blokt Vuur.

Vuur + Water = Water dooft Vuur.

Vuur + Lucht = Lucht versterkt Vuur. Vuur doet extra damage.

Aarde + Aarde = Aarde slaat Aarde kapot, niemand krijgt damage.

Aarde + Water = Water healt zichzelf.

Aarde + Lucht = Aarde valt lucht aan.

Water + Water = Beiden spelers healen.

Water + Lucht = Lucht steelt Water, alleen Lucht healt.

Lucht + Lucht = Er ontstaat een tornado, beiden krijgen damage.

# Eindopdracht

## Turn-based element

In een potje moeten beide spelers één kaart neerleggen in de turn. Daaruit komt een combinatie dat effect heeft op de gamestate. Daarna komt er een nieuwe turn met een lege tafel waar de spelers op nieuwe een kaart neer kunnen leggen.

## Speler profielen

De speler moet eerst een account aanmaken via de website (kun je komen via de client > create account). Daar word er een spelerprofiel aangemaakt dat effect heeft op de gameplay. In de game moet de speler eerst inloggen om toegang te krijgen tot zijn profiel.

## Scores

Na het inloggen komt de speler bij het clientmenu, daar ziet hij zijn naam, score en faction (als visuals) staan. Na elk potje gespeeld te hebben krijgt de winner van het potje RankedScore. Deze wordt geüpdated in de database en zijn daarom ook gelijk weer te zien in de clientmenu.

## Local Multiplayer

Alleen spelers van tegenovergestelde facties kunnen tegen elkaar vechten. De game is 1vs1.

## High-score list

In de clientmenu is er een knop om de highscores te zien. Door hierop te klikken zal de speler naar een pagina geleid worden met de highscores.

## Toepassing theorie

De game maakt gebruik van een Server-Client Network. Ik heb hiervoor gekozen ter beveiliging van de game, zodat de server de baas is over de informatie die verstuurd word. Aangezien de game weinig informatie hoeft door te sturen, heeft dit geen slecht effect op de snelheid van de server. Ook is dit handiger voor de matchmaking, zodat alle spelers via 1 server met elkaar verbonden worden.

## Matchmaking

Ik heb mijn eigen matchmaking systeem gemaakt. De speler moet aan de server aangeven dat hij een potje zoekt. Zo komt hij in de matchmaking lijst te staan. Per factie is er een lijst gemaakt. Na elke aanmelding gaat de matchmaking kijken of er een potje aangemaakt kan worden. De matchmaking zoekt naar de meest compatible tegenstander aan de hand van de RankedScores van de spele.

## Verbetering voor in de toekomst

Voor in de toekomst wil ik graag het matchmaken verbeteren door hem in een loop te zetten, zodat hij om de zoveel tijd matches probeert te maken. Zo wordt de matchmaking pool groter en is er meer kans dat een speler tegen een eerlijkere tegenstander moet vechten.

Ook wil ik het kaartensysteem updaten door de combinaties interessanter te maken. Vooral door er meer utilities toe te voegen. Nu heb je drie voornamelijke combinaties, zoals damage, healing en defense, deze zou ik willen uitbreiden door bijvoorbeeld mana toe te voegen. Een ander idee is om de facties meer invloed te laten hebben in de gameplay door elke factie een buff te geven en de berekeningen van elke kaart daarmee te beïnvloeden. Bijvoorbeeld Healing van The Light is sterker en de Damage van The Dark is sterker.

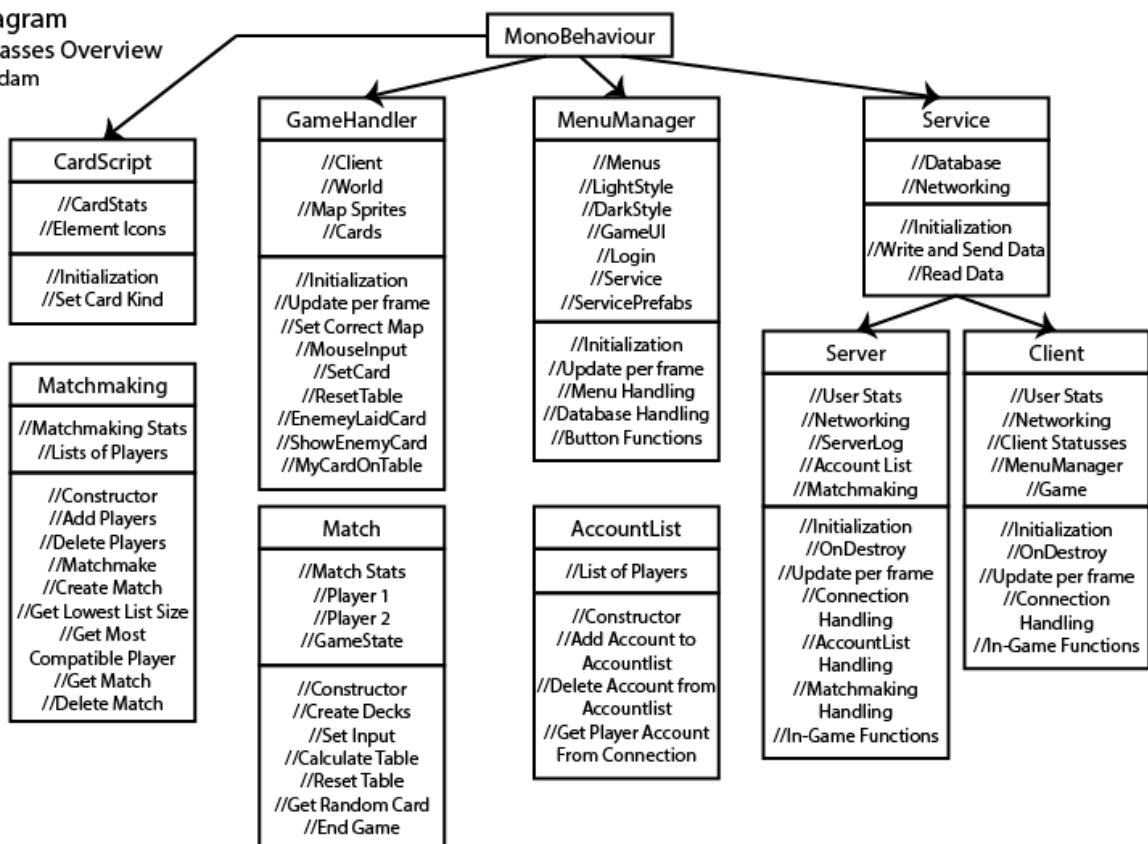
Daarnaast mag de art ook een grote update krijgen en mogen er special effects komen vanuit de combinaties.

## UML-Diagrams

### Class Diagram

#### Simple Classes Overview

Jeffrey Saydam

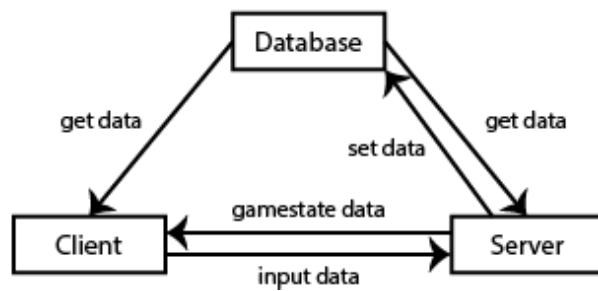


Hier boven een versimpelende Class Diagram om het nut van de classes overzichtelijker te maken.

## Activity Diagram

### Networking Communication Flow

Jeffrey Saydam



Hier boven de networking communication flow. Hoe data wordt verstuurd over de drie main componenten; Client, Server en Database.

## Bronnen

### Programming:

- Gebruik gemaakt van de Networking package.
- Alle andere code door mij Jeffrey Saydam geprogrammeerd (inclusief Matchmaking).

### 2D art:

- Factie iconen, Map en Card achtergrond door mij Jeffrey Saydam gemaakt.
- Element iconen en Factie achtergronden van google afbeeldingen.