Naam: Jiankai Zheng

Locatie: Hogeschool van Arnhem en Nijmegen, Arnhem

Studentnummer: 619927

Course: SPD

Docent: Ben van Hoof

Versie: 0.1

Datum: 10-09-2018

Ontwerp set spel

Functioneel en technisch ontwerp van het Set spel

# Wijzigingshistorie

|  |  |  |  |
| --- | --- | --- | --- |
| Versie | Beschrijving | Auteur | Datum |
| 0.1 | Document gecreëerd  Voorblad, Wijzigingshistorie toegevoegd.  Opdracht omschrijving | Jiankai Zheng | 09-10-2018 |
| 0.2 | Globale flow  Scherm ontwerpen  Software omschrijving | Jiankai Zheng | 10-10-2018 |

Inhoudsopgave

[Wijzigingshistorie 1](#_Toc526851170)

[1. Opdrachtomschrijving 3](#_Toc526851171)

[2. Globale flow 4](#_Toc526851172)

[2.1. Toelichting 4](#_Toc526851173)

[3. Schermontwerpen 5](#_Toc526851174)

# Opdrachtomschrijving

De opdracht betreft het realiseren van een spel dat gebaseerd is op het spel “Set”. Het wordt niet een identiek versie van het spel maar een kleiner versie daarvan. De regels van het spelletje lijken erg op die van Set. Er zijn 27 unieke kaarten in het spel die op drie eigenschappen van elkaar kunnen verschillen.

* De hoeveelheid figuren: 1, 2 of 3.
* De vorm van de figuren: driehoek, rechthoek of ellips
* De kleur van de figuren: rood, groen of blauw

En elke combinatie kan maar een keer voorkomen. Er kan dus maar een kaart zijn die 1 blauwe ellips heeft of 2 rode rechthoeken.

Er zijn op ieder moment 9 kaarten te zien op het scherm tenzij er geen kaarten meer van de stapel te halen zijn. En de speler heeft de taak om sets te vinden die op het scherm zijn. Een set bestaat altijd uit 3 kaarten. De kaarten zijn alleen een set als de eigenschappen op alle 3 kaarten of gelijk zijn of alle verschillend.

Voorbeeld:

1 rode rechthoek / 2 rode rechthoeken / 3 rode rechthoeken = (dit is een set omdat het aantal verschillend is op elke kaart, kleur steeds hetzelfde zijn en vorm steeds hetzelfde zijn)

Het programma moet aan de volgende eisen minimaal voldoen:

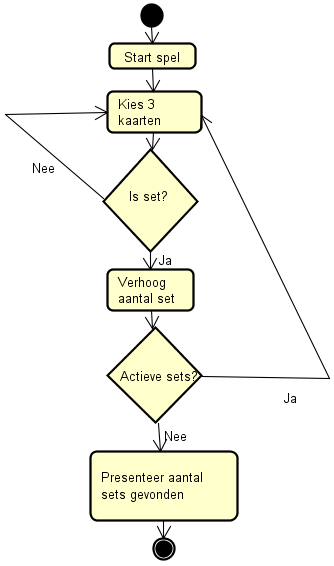
* Alle kaarten moeten worden aangemaakt
* De kaarten moeten geschud worden
* De kaarten moeten netjes op het beeld getekend worden
* De figuren op de kaarten moeten netjes getekend worden, dus mooi uitgelijnd.
* Een kaart mag maar een keer in het spel gebruikt worden
* De kaarten moeten geselecteerd worden en dat moet zichtbaar zijn voor de speler, door bijvoorbeeld de achtergrondkleur te veranderen
* Op ieder moment moet het bekend zijn hoeveel mogelijke sets op “tafel” liggen
* Het aantal gevonden sets door een speler moet bijgehouden worden en aan het einde gepresenteerd worden aan de speler.

Dit zijn de verplichte functionaliteiten, indien er nog tijd over is kunnen er extra functionaliteiten worden toegevoegd.

Meer informatie over het spel: <https://en.wikipedia.org/wiki/Set_(card_game)>

# Globale flow

Globaal gezien zijn onderstaande situaties mogelijk. De pijlen geven aan welke situaties in elkaar over kunnen gaan. Deze situatieverandering is altijd een gevolg van een gebeurtenis (zoals een gebruiker die ergens op klikt).



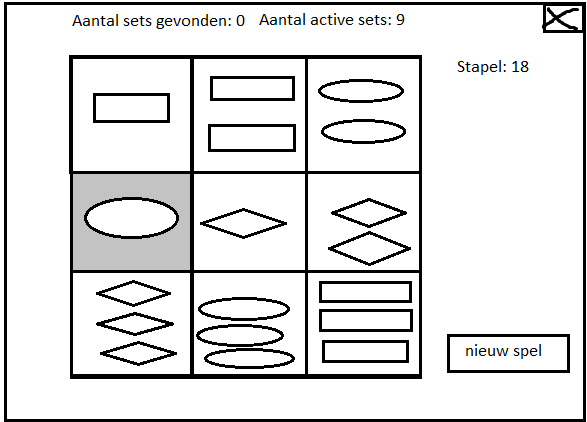
## Toelichting

Wanneer het spel start worden er 9 kaarten op het scherm getoond. De speler kan dan 3 kaarten selecteren, wanneer er 3 kaarten zijn geselecteerd dan worden er gekeken of de geselecteerde kaarten samen een set zijn. Indien het een set is dan verdwijnt de kaarten en krijgt de speler een punt, indien het niet correct is dan gebeurt er niets. De selectie wordt dan weggehaald en kan de speler weer verder spelen totdat er geen sets meer zijn.

Op ieder moment worden de actieve sets bijgehouden, indien er geen sets meer zijn dan worden ze van de stapel gepakt. Indien er geen kaarten meer zijn dan stopt het spel. Op het einde krijgt de speler te zien hoeveel sets hij/zij heeft gevonden.

De bijbehorende schermontwerpen kunt u vinden in het volgende hoofdstuk.

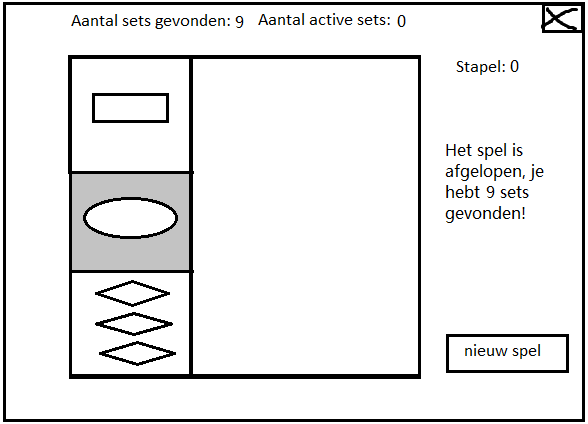
# Schermontwerpen



Scherm 1

## Toelichting

Op het scherm hierboven zie je 9 kaarten, elk kaart is uniek. Je kan de kaarten selecteren, geselecteerde kaarten worden aangegeven door een andere achtergrondkleur. Je kan een selectie verwijderen door er weer op te klikken. Er is ook een knop met ‘nieuw spel’, het huidige spel wordt dan gereset en je begint weer opnieuw vanaf het begin. Boven staan informatie over het spel, zoals: Aantal sets gevonden door een speler, aantal mogelijke sets die op het scherm staan en hoeveel kaarten nog op de stapel liggen.



Scherm 2

## Toelichting

Op het scherm zie je dat het spel is afgelopen, er zijn geen sets meer op het scherm. De speler heeft 9 sets in totaal gevonden, dat wordt op het einde ook aangegeven rechts. Indien de speler nog een spel wilt spelen is dat mogelijk door op nieuw spel te klikken, het spel wordt dan gereset.

# Softwarebeschrijving

In dit hoofdstuk worden de functionaliteiten beschreven door middel van het gebruik van events, methodes en variables etc.

## Globale softwarestructuur

Het programma zal de volgende globale structuur bevatten:

* Module Events: Alle gebruikeracties worden hier opgevangen
* Module Card: deze module zal de methodes bevatten om kaarten te genereren, husselen, tekenen etc.
* Module Game: deze module bevat alle logica om een spel te laten werken.
* Module Player: deze module bevat alle logica over de speler, zoals kaart pakken.
* Module Algemeen: deze module bevat de algemene functies zoals het spel opnieuw starten.

## Class, Events, Methodes en Variables

|  |  |  |
| --- | --- | --- |
| Module | Class, Events, Methodes en Variables | Toelichting |
| Events | Void Setup() | In de setup wordt de initialisatie gedaan van het spel. Ook worden de instellingen van het programma vastgesteld onder andere de schermbreedte en hoogte. Vervolgens worden de functies aangeroepen om het spel te kunnen te starten. |
|  | Void mouseClicked() | Alle muisklikken van de speler wordt opgevangen hierin, zoals wanneer de speler op een kaart klikt. De kaart moet dan geselecteerd worden. |
| Card | class Card | In de Card class worden alle logica van de kaarten gedefinieerd, zoals de eigenschappen |
|  | String[] cardProperties | In deze string array wordt vastgelegd welke eigenschappen de kaart kan hebben |
|  | String[] generateCards() | Deze functie genereert de kaarten voor het spel. |
|  | Boolean isSet(String[] selectedCards) | Deze functie controleert of de geselecteerde kaarten wel samen een set zijn, |
|  | Int activeSets() | Deze functie geeft terug hoeveel sets er nog op de “tafel” liggen. |
|  | String[] Shuffle(String cards[]) | Deze functie husselt de kaarten in een willekeurige volgorde. |
| Game | class Game | In de Game class worden alle logica van het spel gedefinieerd, zoals de kaartenlijst, welke kaarten op de “tafel” liggen. |