

# Analyse- en ontwerpdocument

## (sub) Set



Naam: **Joey Stoffels**  
Studentnr: **609589**  
Docent: **Mark van der Maas**  
Klas: **SEB-B**  
Datum: **23-03-2018**  
Versie: **1.0**

# 1 INHOUDSOPGAVE

---

2	Inleiding.....	3
3	Analyse.....	4
3.1	Schermontwerpen .....	4
3.2	Flow.....	8
3.3	Acties per scherm.....	9
3.4	Functionaliteiten .....	11
3.4.1	MoSCoW .....	11
3.4.2	User Stories .....	12
4	Ontwerp .....	16
4.1	Softwarebeschrijving.....	16
4.1.1	Globale structuur .....	16
4.1.2	Variabelen .....	21
4.2	Globaal stappenplan .....	23
5	Test cases .....	24
5.1	Opgenomen test cases.....	24
5.2	Testrapportage en debugging.....	24
6	Conclusie en reflectie.....	26
7	Bronnen.....	27

## 2 INLEIDING

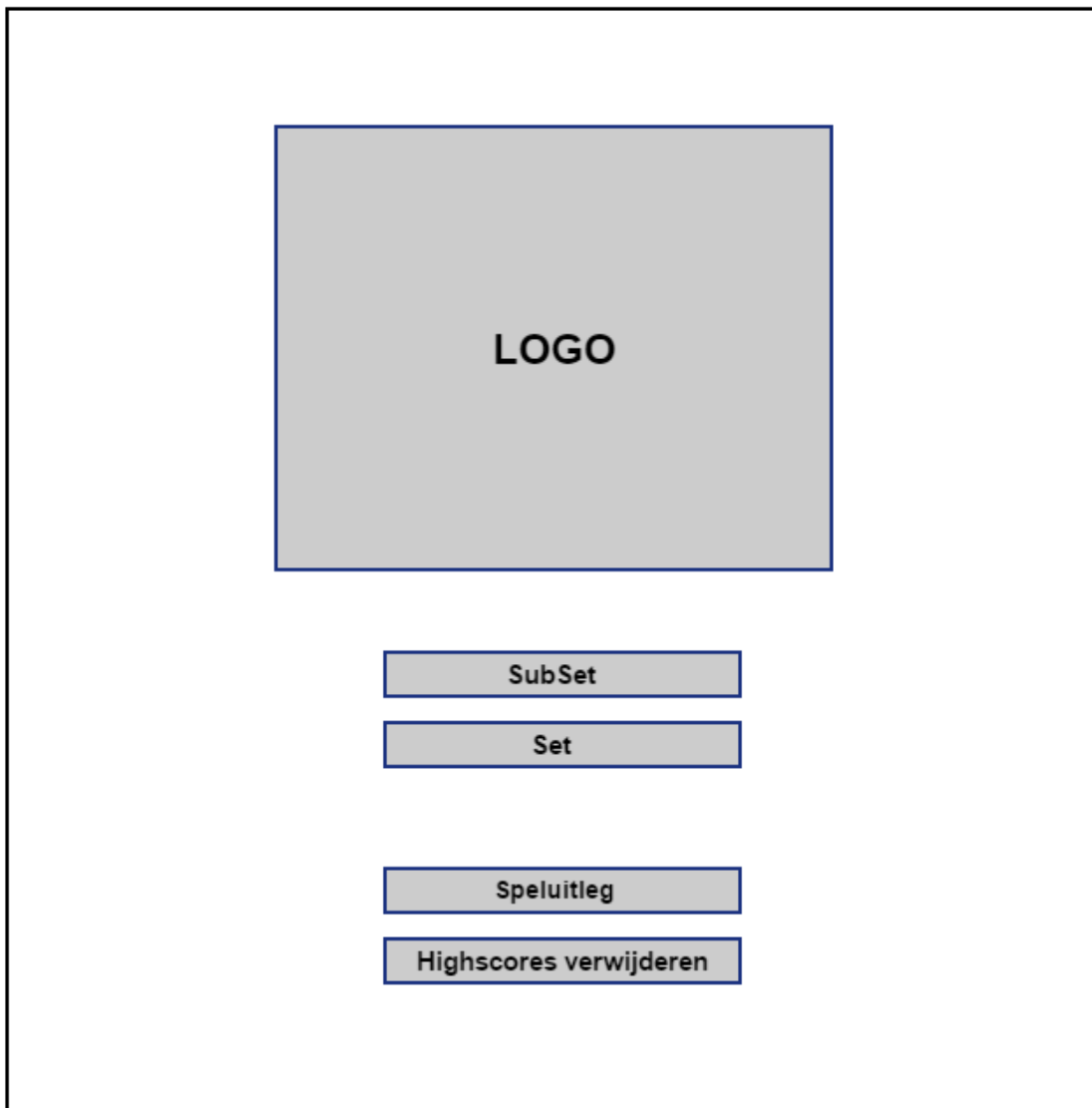
---

## 3 ANALYSE

---

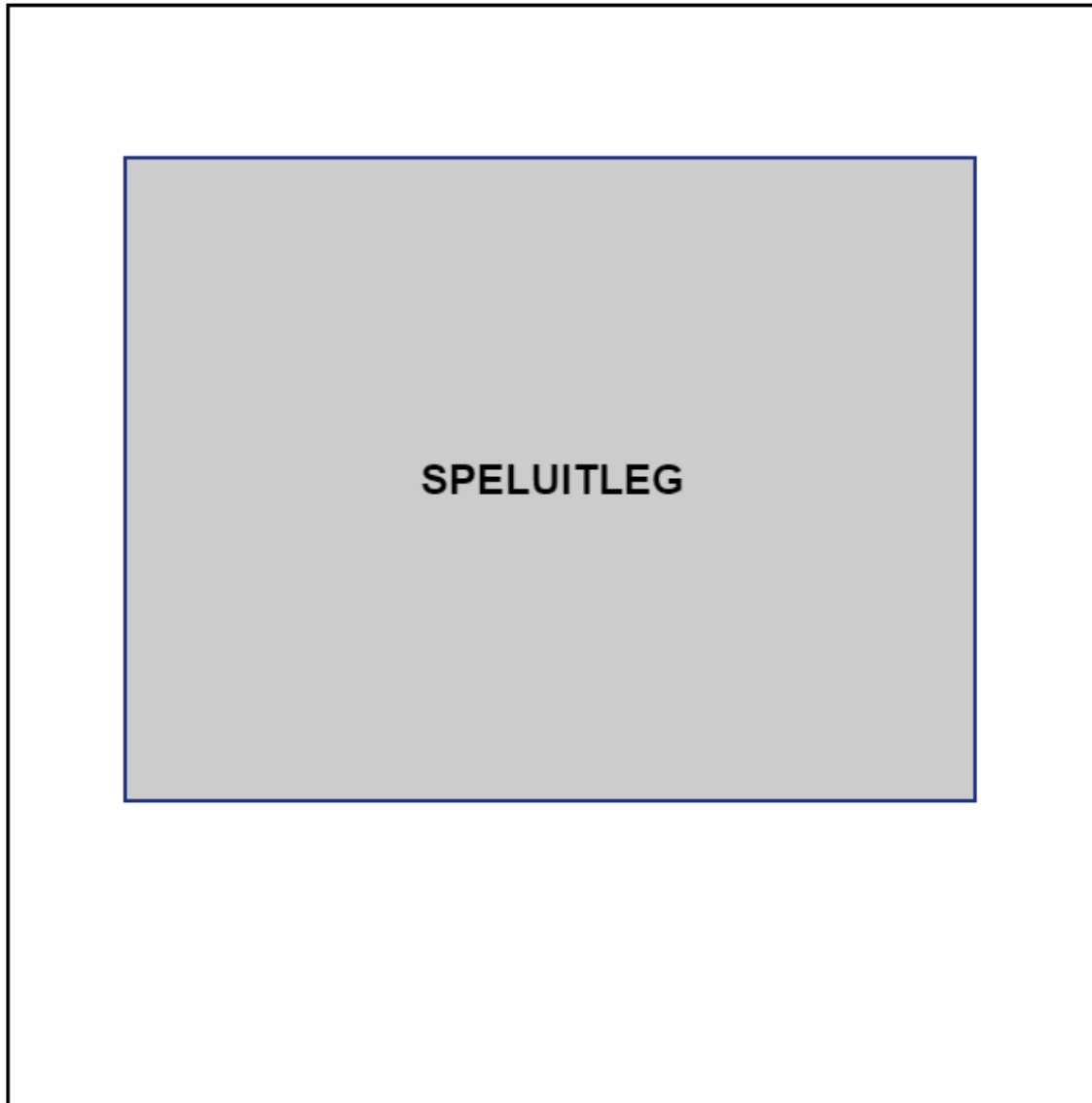
### 3.1 SCHERMONTWERPEN

In de onderstaande figuren zijn de schermontwerpen te zien voor het te bouwen spel. De functionele werking van de schermen staat eronder beschreven.



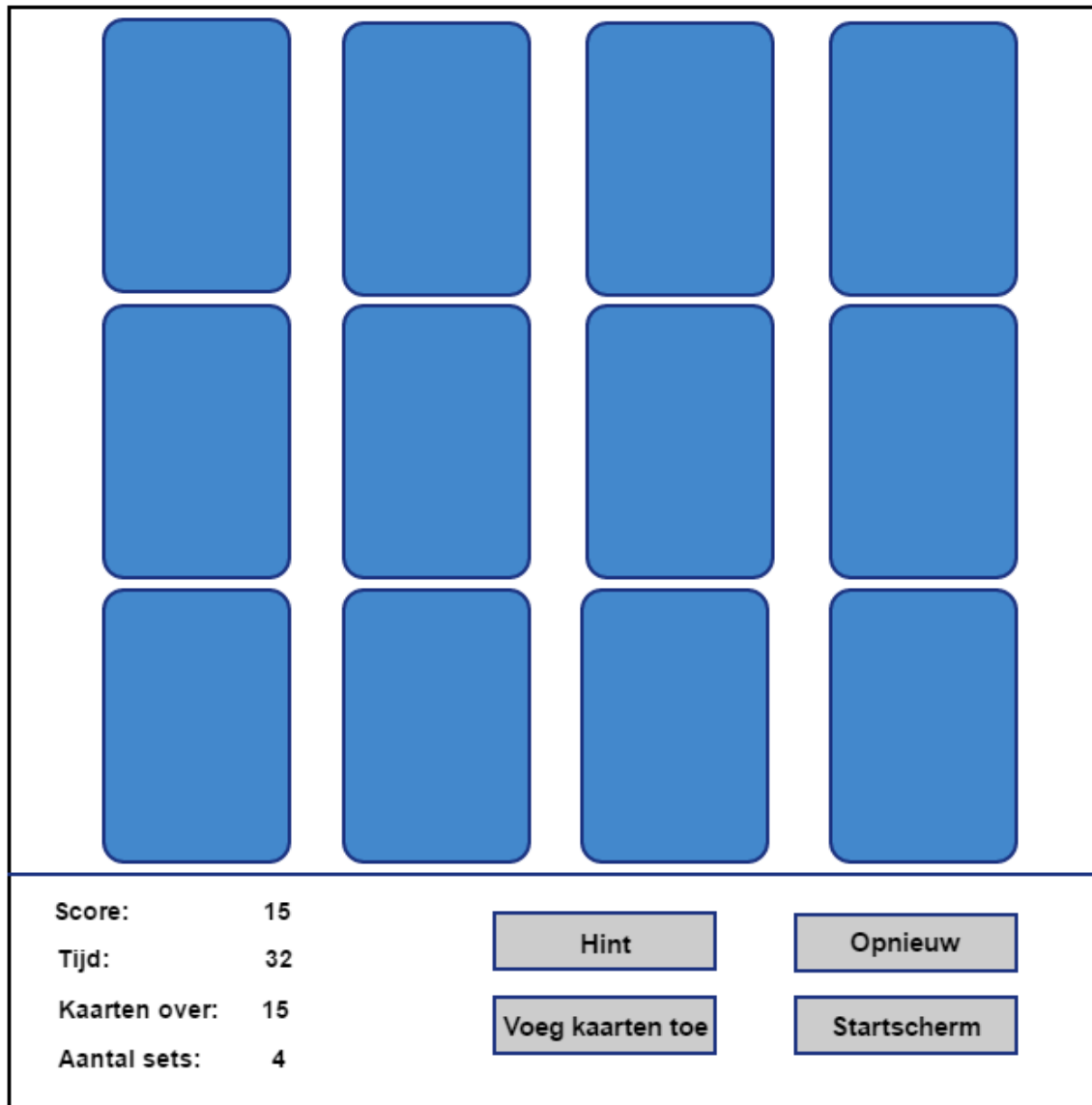
Figuur 1: Schermontwerp startscherm

Het startscherm zal de speler van het spel als eerste te zien krijgen. De speler kan op dit scherm kiezen uit twee spelvarianten van Set, namelijk SubSet of het klassieke Set. Tevens kan de speler op dit scherm ervoor kiezen om speluitleg te bekijken, of om de highscores te verwijderen.



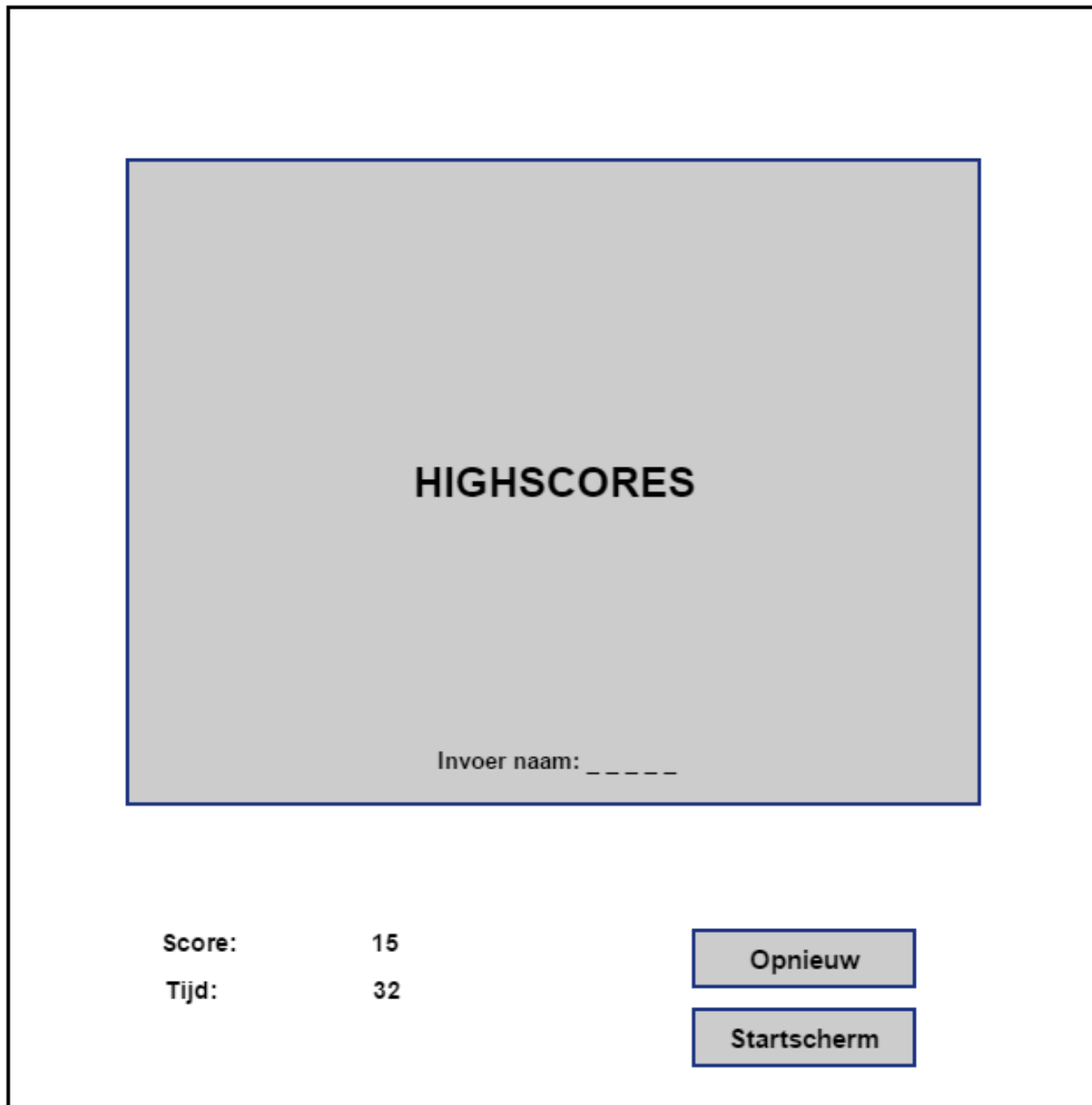
Figuur 2: Schermontwerp speluitleg

Wanneer er op het beginscherm is gekozen voor de speluitleg wordt er een scherm getoond met daarin informatie over hoe set gespeeld wordt. Hierin met name uitgelegd hoe een set bepaald wordt. Om terug te keren naar het startscherm dient er op een willekeurige plek op het scherm geklikt te worden.



Figuur 3: Schermontwerp speelveld

Wanneer er op het beginscherm is gekozen voor Set of SubSet wordt het speelveld getoond. Wanneer er voor de spelvariant SubSet is gekozen zullen er 9 kaarten worden weergegeven in plaats van 12 voor het spelvariant Set. De overige informatie en knoppen blijven voor beide spelvarianten hetzelfde. De speler kan op dit scherm sets gaan maken door op de weergegeven kaarten te klikken. Rechtsonder kan de speler eventueel een hint vragen, kaarten toevoegen, opnieuw beginnen of terug gaan naar het startscherm. Linksonder wordt er informatie weergegeven over de voortgang van het huidige spel.

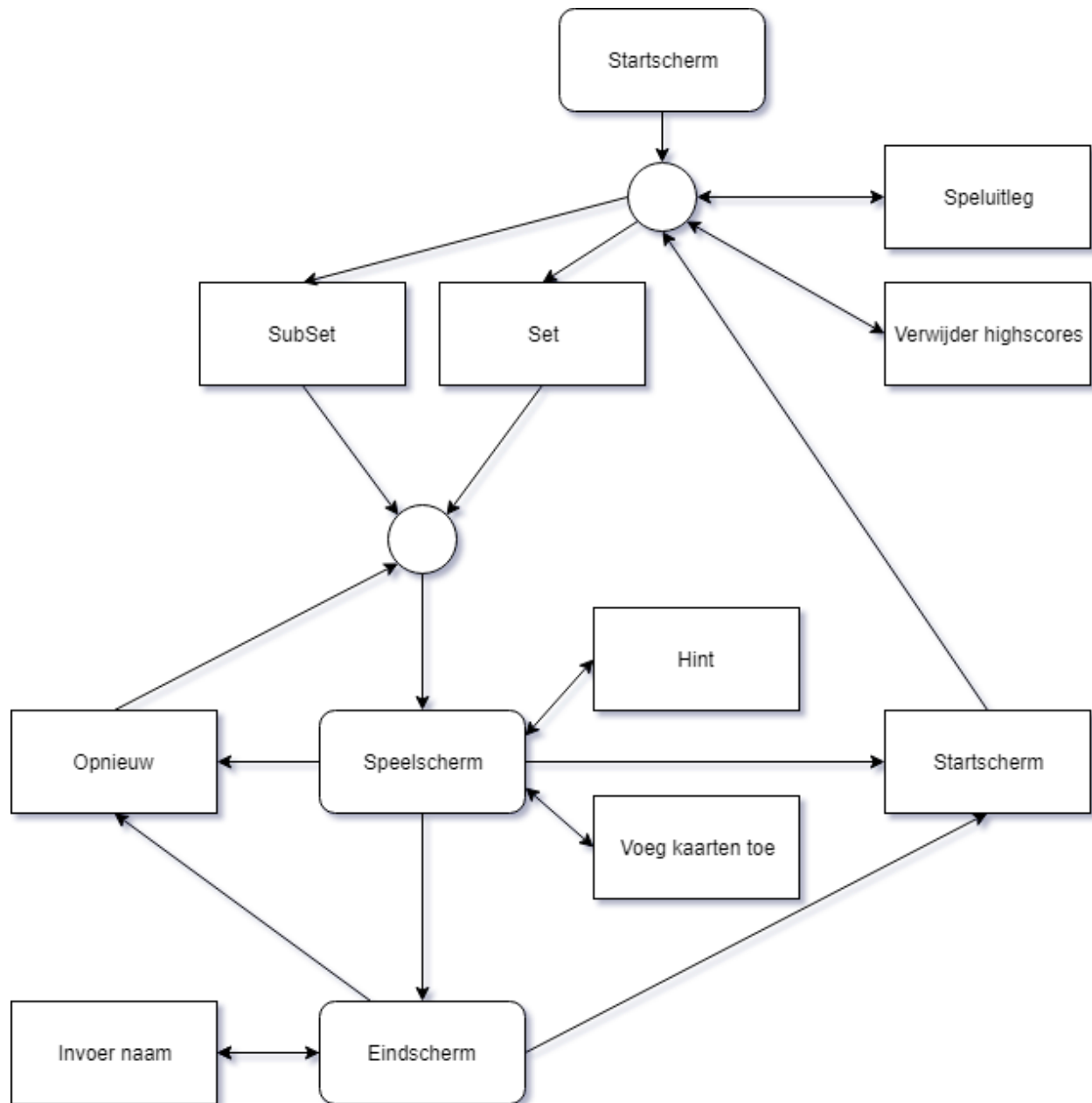


Figuur 4: Schermontwerp eindscherm

Wanneer er geen sets meer in het speelveld te vinden zijn, en de voeg kaarten toe hulplijn is gebruikt, is het spel afgelopen en wordt het eindscherm getoond. De speler kan op dit scherm zijn naam invoeren en de huidige highscores zien van de spelvariant die gespeeld is. Linksonder is de score te zien van het gespeelde spel. Rechtsonder blijven enkel nog de 'opnieuw' en 'startscherm' knop in beeld zodat de speler opnieuw kan beginnen of terug kan gaan naar het startscherm.

### 3.2 FLOW

In onderstaande figuur is de globale flow door het programma in een flowchart weergegeven. In 3.3 wordt deze flow verder toegelicht.



Figuur 5: Flowchart (sub)set



### 3.3 ACTIES PER SCHERM

#### *Startscherm:*

De speler ziet het logo van het spel (Set). Hij/zij kan kiezen uit een aantal knoppen:

- Subset
- Set
- Speluitleg
- Highscores verwijderen

#### *Knop 'SubSet' of 'Set':*

Het spel start met 3 variaties op de speelkaarten en 9 kaarten op het speelveld wanneer voor SubSet gekozen is. Is er voor Set gekozen dan zal het spel opstarten met 4 variaties op de speelkaarten en 12 kaarten op het speelveld. Het speelscherm wordt tenslotte weergegeven.

#### *Knop 'Spelregels':*

Het spelregels scherm wordt actief.

#### *Spelregels:*

Op dit scherm wordt er een speluitleg getoond. Wanneer de speler ergens op het spel klikt zal er worden teruggedaan naar het startscherm.

#### *Knop 'Highscores verwijderen':*

Alle highscores worden verwijderd. Startscherm blijft actief.

#### *Speelscherm:*

Het speelscherm is actief en de speler ziet het speelveld met de kaarten in beeld. Daarnaast wordt er spelinformatie (score, tijd, aantal sets op het veld en aantal kaarten over) weergegeven. De tijd begint te lopen en de score staat op 0. Wanneer de speler een set vindt zal de score met 1 verhoogd worden. Wanneer de speler om een hint vraagt zal zijn score met een halve punt afnemen. De speler kan een aantal knoppen kiezen, 'hint', 'opnieuw', 'voeg kaarten toe' en 'startscherm'.

Wanneer de speler het spel uitspeelt doordat er geen sets meer op het speelveld beschikbaar zijn (ook niet nadat de 'Voeg kaarten toe' knop is gebruikt) is het spel afgelopen. Het eindscherm zal automatisch verschijnen.

#### *Knop 'Hint':*

Op het speelveld krijgen twee kaarten die onderdeel zijn van een set zichtbaar door de rand van de kaart een andere kleur te geven. De speler hoeft dus nog maar één kaart erbij te zoeken voor een set. De score wordt verminderd met een halve punt wanneer op deze knop is gedrukt.

#### *Knop 'Voeg kaarten toe':*

Wanneer de speler op deze knop drukt wordt het spel uitgebreid door een extra kolom kaarten aan het bestaande speelveld toe te voegen. Dit zorgt ervoor dat er meer sets op het speelveld komen te liggen waardoor de speler wellicht sneller een set kan vinden. Door deze knop neemt de breedte van het speelscherm toe.

#### *Knop 'Opnieuw':*

Het spel wordt opnieuw gestart met een nieuwe stapel kaarten en een nieuw speelveld. Het

speelscherm blijft actief en alle relevante spelvariabelen worden gereset zodat het spel weer op 'nul' begint.

*Knop 'Startscherm':*

De speler kan door middel van deze knop weer terugkeren naar het startscherm. Door deze actie worden ook alle spelvariabelen gereset zodat er met een nieuw spel kan worden gestart.

*Eindscherm:*

Op het eindscherm worden de highscores weergegeven. De scores van het afgeronde spel zijn zichtbaar. De speler heeft de mogelijkheid om zijn naam in te voeren en middels de enter toets zijn of haar score tenslotte aan de highscores toe te voegen. Wanneer dit gedaan is kan de speler tenslotte teruggaan naar het startscherm of opnieuw het spel starten door op de gelijknamige knoppen te klikken.

## 3.4 FUNCTIONALITEITEN

In hoofdstuk 3.4.1 worden de functionaliteiten die benodigd zijn voor het programma op volgorde van prioriteit gezet met behulp van de MoSCoW-methode. De 'Must haves' in deze lijst is opgemaakt uit functionaliteiten die nodig zijn voor het 'minimum viable product' en hebben dus de hoogste prioriteit bij het maken van dit programma. De 'Should haves' hebben na de 'Must haves' de hoogste prioriteit en die hebben een sterke voorkeur om meegenomen te worden in de applicatie. De overige items in de MoSCoW lijst zijn optioneel, maar opgedeeld in 'Could haves' en 'Would haves'. Tenslotte wordt er een eerste opzet voor user stories gemaakt voor deze functionaliteiten in hoofdstuk 3.4.2.

### 3.4.1 MoSCoW

Must haves:

- Aanmaken kaarten stapel
- Kaarten aanmaken adhv. 3 eigenschappen
- Kaarten schudden
- Kaarten weergeven/tekenen
- Kaarten maar eenmaal gebruiken in het spel
- Figuren op kaarten weergeven/tekenen
- Kaarten stapel actueel houden
- Kaarten selecteren op het speelveld
- Aantal sets op tafel
- Aantal gevonden sets
- Spel afgelopen wanneer kaarten stapel + speelveld leeg is
- Spel afgelopen wanneer 0 sets op tafel

Should haves:

- Eindscherm met score weergeven
- Kaart deselecteren
- "Geef hint" knop toevoegen
- Beginscherm (selecteren speltype, subset/set)
- Kaarten aanmaken adhv. 4 eigenschappen
- 4 eigenschappen variant (Set)
- "Voeg kaart toe" knop toevoegen
- Spel afgelopen wanneer 0 sets op tafel + "voeg kaarten toe" knop gebruikt

Could have:

- Highscore lijst met naam invoer
- Twee speler functionaliteit
- Cheats
- Speluitleg scherm

Would have:

- Netwerk functionaliteit
- Multiplayer optie over netwerk

### 3.4.2 User Stories

Functionaliteit	<ul style="list-style-type: none"><li>• Aanmaken kaarten stapel</li></ul>
User story	Als gebruiker van het programma wil ik dat de kaartenstapel automatisch wordt aangemaakt wanneer een speltype gekozen is zodat het spel gespeeld kan worden met deze kaarten stapel.

Functionaliteit	<ul style="list-style-type: none"><li>• Kaarten aanmaken adhv. 3 eigenschappen</li></ul>
User story	Als gebruiker van het programma wil ik dat de kaarten aan de hand van 3 eigenschappen kunnen worden gemaakt zodat ik de SubSet spelvariant kan spelen.

Functionaliteit	<ul style="list-style-type: none"><li>• Kaarten schudden</li></ul>
User story	Als gebruiker van het programma wil ik dat de kaarten geschud worden zodat ieder spel start met een uniek speelveld.

Functionaliteit	<ul style="list-style-type: none"><li>• Kaarten weergeven</li></ul>
User story	Als ontwikkelaar van het programma wil ik de kaarten op een gestructureerde manier op het speelveld tekenen zodat het makkelijk is om later de figuren hierop te tekenen.

Functionaliteit	<ul style="list-style-type: none"><li>• Kaarten maar eenmaal gebruiken in het spel</li></ul>
User story	Als gebruiker van het programma wil ik dat alle kaarten slechts eenmaal in het spel voorkomen zodat het spel volgens de spelregels gespeeld kan worden.

Functionaliteit	<ul style="list-style-type: none"> <li>• Figuren op kaarten weergeven/tekenen</li> </ul>
User story	Als ontwikkelaar van het programma wil ik de figuren van de speelkaarten op een gestructureerde, generieke manier kunnen tekenen op de kaarten zodat de code hiervoor overzichtelijk blijft.

Functionaliteit	<ul style="list-style-type: none"> <li>• Kaarten stapel actueel houden</li> </ul>
User story	Als gebruiker van het programma wil ik kunnen zien hoeveel kaarten er nog in het spel zitten zodat ik een idee krijg van het aantal sets dat er nog te behalen valt.

Functionaliteit	<ul style="list-style-type: none"> <li>• Kaarten selecteren op het speelveld</li> </ul>
User story	Als gebruiker wil ik kaarten kunnen selecteren zodat ik een set kan maken.

Functionaliteit	<ul style="list-style-type: none"> <li>• Aantal sets op tafel</li> </ul>
User story	Als gebruiker wil ik kunnen zien hoeveel sets er op het speelveld liggen zodat ik niet hoeft te blijven zoeken wanneer er geen sets meer op het speelveld liggen.

Functionaliteit	<ul style="list-style-type: none"> <li>• Aantal gevonden sets</li> </ul>
User story	Als gebruiker van het programma wil ik kunnen zien wat het aantal gevonden sets is zodat ik weet wat mijn score wordt.

Functionaliteit	<ul style="list-style-type: none"> <li>• Spel afgelopen wanneer kaarten stapel + speelveld leeg is</li> </ul>
User story	Als gebruiker van het programma wil ik dat het spel eindigt zodra er geen kaarten meer zijn en het speelveld leeg is zodat duidelijk is dat het spel afgelopen is en er een eindscore bekend is.

Functionaliteit	<ul style="list-style-type: none"> <li>• Spel afgelopen wanneer 0 sets op tafel</li> </ul>
User story	Als gebruiker van het programma wil ik dat het spel eindigt wanneer er geen sets meer op het speelveld liggen zodat het spel niet vast komt te zitten als dit niet zou gebeuren.

Functionaliteit	<ul style="list-style-type: none"> <li>• Eindscherm met score weergeven</li> </ul>
User story	Als gebruiker van het programma wil ik dat wanneer het spel is afgelopen er een eind scherm met de behaalde score wordt weergegeven zodat ik kan zien dat het spel is afgelopen en wat mijn score is.

Functionaliteit	<ul style="list-style-type: none"> <li>• Kaart deselecteren</li> </ul>
User story	Als gebruiker van het programma wil ik een geselecteerde kaart kunnen deselecteren zodat ik een foute kaart keuze ongedaan kan maken.

Functionaliteit	<ul style="list-style-type: none"> <li>• “Geef hint” knop toevoegen</li> </ul>
User story	Als gebruiker van het programma wil ik om een hint kunnen vragen zodat ik een moeilijk te vinden set kan vinden.

Functionaliteit	<ul style="list-style-type: none"> <li>• Startscherm (selecteren speltype, subset/set)</li> </ul>
User story	Als gebruiker van het programma wil ik een startscherm met een keuze tussen subset en set zodat ik kan kiezen welke spelvariant ik ga spelen.

Functionaliteit	<ul style="list-style-type: none"> <li>• Kaarten aanmaken adhv. 4 eigenschappen</li> </ul>
User story	Als ontwikkelaar van het programma wil ik een generieke functie hebben waarmee ik kaarten aan kan maken met 4 eigenschappen zodat de gebruiker de klassieke set variant kan spelen en zodat ik herbruikbare code gebruik.

Functionaliteit	<ul style="list-style-type: none"> <li>• 4 eigenschappen variant (Set)</li> </ul>
User story	Als gebruiker van het programma wil ik de klassieke set kunnen spelen met 4 kaarteigenschappen zodat ik de keuze heb tussen een makkelijke en moeilijke spelvariant.

Functionaliteit	<ul style="list-style-type: none"> <li>• “Voeg kaarten toe” knop toevoegen</li> </ul>
User story	Als gebruiker van het programma wil ik met een knop een nieuwe kolom kaarten kunnen toevoegen zodat er meer kaarten op tafels komen te liggen en ik sneller een set kan vinden.

Functionaliteit	<ul style="list-style-type: none"> <li>• Spel afgelopen wanneer 0 sets op tafel + voeg kaarten toe knop gebruikt</li> </ul>
User story	Als gebruiker van het programma wil ik dat het spel is afgelopen wanneer er 0 sets op tafel liggen én wanneer de voeg kaarten toe knop is gebruikt zodat het spel ten einde komt omdat het anders vast zit.

Functionaliteit	<ul style="list-style-type: none"> <li>• Highscore lijst met naam invoer</li> </ul>
User story	Als gebruiker van het programma wil ik op het eindscherm mijn naam kunnen invoeren zodat mijn score wordt opgenomen in de highscore lijst.

Functionaliteit	<ul style="list-style-type: none"> <li>• Twee speler functionaliteit</li> </ul>
User story	Als gebruiker van het programma wil ik een optie om met twee spelers te kunnen spelen zodat ik dit spel samen met iemand competitief kan spelen.

Functionaliteit	<ul style="list-style-type: none"> <li>• Speluitleg scherm</li> </ul>
User story	Als gebruiker van het programma wil ik een pagina waarop ik een speluitleg kan bekijken zodat ik weet hoe ik het spel moet spelen.

Functionaliteit	<ul style="list-style-type: none"> <li>• Netwerk functionaliteit</li> </ul>
User story	Als ontwikkelaar van het programma wil ik netwerk functionaliteit toevoegen zodat op een later moment er via internet of een lokaal netwerk het spel gespeeld kan worden.

Functionaliteit	<ul style="list-style-type: none"> <li>• Multiplayer optie over netwerk</li> </ul>
User story	Als gebruiker van het programma wil ik het spel via een netwerk kunnen spelen zodat ik met andere mensen op afstand het spel competitief kan spelen.

## 4 ONTWERP

---

### 4.1 SOFTWAREBESCHRIJVING

#### 4.1.1 Globale structuur

De volgende tabs worden binnen de code opgezet om de code per ‘thema’ overzichtelijk gescheiden te kunnen houden. Per tab zijn tenslotte alle functies genoemd. Daarnaast wordt er per functie aangegeven of het andere functies aanroept zodat de flow tussen de functies onderling duidelijk wordt.

##### SubSet

- void settings()
- void setup()
  - *toonLaadScherM()*
  - *maakSpelScherM()*
  - *laadMediaBestanden()*
- void draw()
  - *toonLaadScherM()*
  - *toonSpelregels()*
  - *toonAchtergrondKaarten()*
  - *tekenKaartFiguren()*
  - *tekenScorebord()*
  - *updateTijd()*
  - *checkEindeSpel()*
  - *maakSpelScherM()*
  - *toonEindscherM()*
- void setupSpel()
  - *resetSpelVariabelen()*
  - *maakSpelScherM()*
  - *maakKaartenStapel()*
  - *maakSpeelveld()*
  - *telAantalSetsSpeelveld()*

##### Cheats

- void setAchtergrondVideo(Movie)
- void toonAchtergrondVideo()
- void activeerEindeSpel()



## Eindscherm

- void checkEindeSpel()
  - *toonEindscherm()*
- void toonEindScherm()
  - *toonHighscoreLijst()*
  - *tekenNaamInvoer()*
  - *tekenEindeTitel()*
  - *tekenKnopScorebord()*
  - *tekenTextInKnopScorebord()*
  - *tekenSpellInfoTextTitel()*
  - *tekenSpellInfoTextWaarde()*
- void tekenNaamInvoer()
- void tekenEindeTitel()
  - *verdanaBold()*

## Helper

- boolean contains(String[], String)
- String[] removeFromArray(String[], int)
- String[][] append2DArray(String[][], String[])
- boolean arrayContains(String[], String)
- int arrayIndexOf(String[], String)
- int array2DIndexOf(String[][], String[])

## Highscore

- void opslaanHighscore()
  - *sorteerHighscores()*
  - *getPlaatsInHighscoresLijst()*
- int getPlaatsInHighscoresLijst(String[], String)
- String[] sorteerHighscores(String[])
- void toonHighscoreLijst()
  - *sorteerHighscores()*
- void verwijderHighscores()

## Laadscherm

- void toonLaadScherm()

## Events

- void mousePressed()
  - *setupSpel()*
  - *verwijderHighscores()*
  - *voegKaartenToe()*
  - *geefHint()*
  - *selecteerKaart()*
- void selecteerKaart()
  - *verifieerSet()*
- void keyPressed()
  - *keyPressedCoded()*
  - *keyPressedNonCoded()*
- void keyPressedCoded()
  - *setAchtergrondVideo()*
  - *activeerEindeSpel()*
- void keyPressedNonCoded()
  - *opslaanHighscore()*
  - *setNaam()*
- void geefHint()
  - *resetSpeelveldAchtergrond()*
  - *contains()*
- void voegKaartenToe()
  - *maakSpelScherm()*
  - *maakSpeelveld()*
  - *telAantalSetsSpeelveld()*
- void verifieerSet()
  - *isSet()*
  - *verwijderKaarten()*
  - *telAantalSetsSpeelveld()*
  - *resetSpeelveldAchtergrond()*
- void resetSpeelveldAchtergrond()
- boolean isSet(String, String, String)
  -
- void movieEvent(Movie)
- void resetSpelVariabelen()
  - *resetSpeelveldAchtergrond()*
- void laadMediaBestanden()

## Spelregels

- void toonSpelRegels()

## Kaarten

- String pakKaart()
  - *removeFromArray()*
- void maakKaartFiguur(String, int, int)
- void toonAchtergrondKaarten()
- void maakKaartenStapel(int)

## Scorebord

- void tekenScorebord()
  - *tekenInfoTextScorebord()*
  - *tekenKnoppenScorebord()*
- void tekenInfoTextScorebord()
  - *tekenSpellInfoTextWaarde()*
  - *tekenSpellInfoTextTitel()*
- void tekenKnoppenScorebord()
  - *tekenKnopScorebord()*
  - *tekenTextInKnopScorebord()*
- void tekenKnopScorebord(float, float)
- void tekenTextInKnopScorebord(String, float, float)
  - *verdanaBold()*
- void tekenSpellInfoTextWaarde(String, float)
  - *verdanaBold()*
- void tekenSpellInfoTextTitel(String, float)
  - *verdanaBold()*

## Speelveld

- void maakSpeelveld()
  - *pakKaart()*
- void tekenKaartFiguren()
  - *maakKaartFiguur()*
- void verwijderKaarten(String[])
- void telAantalSetsSpeelveld()
  - *setInSetsLijst()*
  - *isSet()*
- boolean setInSetsLijst(String[])

## Startscherm

- void toonStartScherm()
  - *tekenLogo()*
  - *tekenKnoppenStartScherm()*
- void tekenLogo()
- void tekenKnoppenStartScherm()
  - *tekenKnopStartScherm()*
- void tekenKnopStartScherm(String, float)
  - *tekenTextInKnopStartScherm()*
- void tekenTextInKnopStartScherm(String, float)

## Variabelen

- void setNaam(String)
- PFont verdanaBold(int)
- void updateTijd()

#### 4.1.2 Variabelen

In onderstaande tabel zijn alle variabelen te zien die in de code worden opgenomen. De variabelen die beginnen met 'final' zullen tijdens het spel niet wijzigen, dit zijn constanten binnen de applicatie.

Naam en type variabele	Beschrijving
int schermBreedte	Waarde voor de breedte van het scherm. Waarde kan veranderen tijdens het spel, vandaar niet final.
final int schermHoogte	Waarde voor de hoogte van het scherm.
final int scorebordHoogte	Vaste waarde voor de hoogte van het scorebord.
final int speelveldHoogte	Vaste waarde voor de hoogte van het speelveld.
int xVelden	Aantal x-as velden (kaarten) op het speelveld. Deze waarde kan veranderen tijdens het spel.
final int yVelden	Aantal y-as velden (kaarten) op het speelveld.
int kaartGridBreedte	Breedte van een cel/grid op een kaart.
int kaartGridHoogte	Hoogte van een cel/grid op een kaart.
final int kaartBreedte	Breedte van een kaart.
final int kaartHoogte	Hoogte van een kaart.
final String legeKaart	Default waarde voor een 'legekaart', ofwel een leeg vlak op het speelveld.
final color rood	Variabele voor de kleur rood.
final color groen	Variabele voor de kleur groen.
final color blauw	Variabele voor de kleur blauw.
final color wit	Variabele voor de kleur wit.
final color zwart	Variabele voor de kleur zwart.
final float[][] ellipseConfig	Configuratiewaarden om een ellips te tekenen. Array bevat factoren die gebruikt worden om de ellips(en) te tekenen op een kaart.
final float[][] rechthoekConfig	Configuratiewaarden om een rechthoek te tekenen. Array bevat factoren die gebruikt worden om de rechthoek(en) te tekenen op een kaart.
final float[][] driehoekConfig	Configuratiewaarden om een driehoek te tekenen. Array bevat factoren die gebruikt worden om de driehoek(en) te tekenen op een kaart.
String achtergrondVideoBestandsLocatie	Locatie van het achtergrondVideoBestand.
String setImgBestandsLocatie	Locatie van het set logo img bestand.
String setSpelregelsBestandsLocatie	Locatie van het set spelregels img bestand.
String airwolfVideoBestandsLocatie	Locatie van het airwolf video bestand.
String airwolfLogoBestandsLocatie	Locatie van het airwolf logo img bestand.
PImage setLogo;	Het set logo plaatje.
PImage setSpelregels;	Het set spelregels plaatje.
PImage airwolfLogo;	Het airwolf logo plaatje.
Movie airwolfVideo;	De airwolf video.
Movie achtergrondVideo;	De achtergrond video.

Movie rainbowVideo;	De regenboog video.
String kaartenInSpel[]	Array bevat alle huidige kaarten die in het spel zitten.
String[][] speelVeld	Tweedimensionale array dat gebruikt wordt om het speelveld te voorzien van kaarten.
color[][] speelVeldKleur	Tweedimensionale array dat gebruikt wordt om de kaarten op het speelveld te voorzien van een achtergrondkleur. De selecteerKaart() functie kan deze waarden wijzigen.
color[][] speelkaartBorderKleur	Tweedimensionale array dat gebruikt wordt om de kaarten te voorzien van een borderkleur. De geefHint() functie kan deze waarden wijzigen.
final String[] aantalFiguren	Array met daarin waarden wat het aantal figuren op een kaart betekend.
final String[] kleurFiguur	Array met daarin enkele letters wat de kleur van de figuren op een kaart betekend.
final String[] figuurType	Array met daarin enkele letters wat een figuur op een kaart betekend.
final String[] vullingFiguur	Array met daarin enkele letters wat de vulling van een figuur op een kaart betekend.
String[] geselecteerdeKaarten	Array met daarin de huidige geselecteerde kaarten op het speelveld.
float scoreSpelerEen	Score van speler 1.
int aantalSetsSpeelveld	Het aantal sets dat momenteel op het speelveld aanwezig is.
ArrayList<String[]> setsLijst	ArrayList met de sets die momenteel op het speelveld liggen.
int aantalVariaties	Het aantal variaties dat gebruikt wordt in het huidige spel. 3 Variaties is SubSet, 4 variaties is Set.
boolean spelActief	Boolean om aan te geven of het spel actief is.
boolean startSchermActief	Boolean om aan te geven of het startscherm actief is.
boolean spelAfgelopen	Boolean om aan te geven of het spel is afgelopen.
boolean spelregelsActief	Boolean om aan te geven of de spelregels getoond worden.
boolean kaartenToegevoegd	Boolean om aan te geven of de 'voeg kaarten toe' functie is gebruikt.
boolean airwolfThemeActive	Boolean om aan te geven of de Airwolf theme actief is.
boolean scoreOpgeslagen	Boolean om aan te geven of de speler zijn highscore heeft opgeslagen.
int aantalHintKaarten	Het aantal kaarten dat wordt opgelicht door middel van een andere border kleur. Kan worden aangepast met de F10 knop.
String naam	De naam van de speler. Te gebruiken bij het opslaan van een highscore.
float tijd	De tijd dat het spel al bezig is.
float restartTijd	Waarde om de tijd te 'resetten' wanneer er een nieuw spel gestart wordt.
int aantalKaartenOpSpeelveld	Het aantal kaarten dat momenteel op het speelveld ligt.

## 4.2 GLOBAAL STAPPENPLAN

De gehele applicatie wordt globaal volgens onderstaand stappenplan opgebouwd.

1. Speelveld array maken op basis van 3 variaties
2. Speelveld tekenen met string weergave van kaarten
3. Kaarten tekenen
4. Speelveld tekenen met kaarten
5. Kaarten (de)selecteren
6. Logica voor het bepalen van een set
7. Score/tijd bijhouden
8. Scorebord tekenen
9. Spelinfo weergeven in het scorebord
10. Functionaliteit + weergave voor de 'opnieuw' knop
11. Functionaliteit + weergave voor de 'voeg kaarten toe' knop
12. Functionaliteit + weergave voor de 'hint' knop
13. Eindscherm toevoegen
14. Beginscherm toevoegen
15. Functionaliteit voor de highscoreslijst
16. Highscores toevoegen aan het eindscherm
17. Naam invoer voor highscore
18. Functionaliteit om een highscoreslijst leeg te maken
19. Speluitleg scherm
20. Beginscherm uitbreiden met 'speluitleg' en 'highscores verwijderen' knoppen
21. Achtergrondvideo functionaliteit inbouwen
22. Inbouwen 'cheats' tbv. oa. snel testen/debuggen
23. Toevoegen laadscherm
24. Geheel visueel verder opmaken

## 5 TEST CASES

---

### 5.1 OPGENOMEN TEST CASES

Concrete testcases opgenomen in het ontwerp.

Testcase	Actie	Verwacht	Geslaagd
1	Spel afgelopen	Bij eindscherm wordt de gebruikte speeltijd correct weergegeven.	Nee
2	Speelveld tekenen	Iedere kaartpositie op het speelveld wordt ingevuld door een kaart.	Nee
3	Einde spel	Het spel gaat automatisch naar het eindscherm wanneer er geen sets meer op het speelveld liggen én na gebruik van de 'voeg kaarten toe' knop.	Ja
4	Knop 'beginscherm'	Het spel gaat terug naar het beginscherm en alle spelvariabelen worden gereset zodat een nieuw spel niet waarden van een oud spel hebben.	Ja
5	F10 knop ingedrukt	Aantal hint kaarten switched tussen 2 en 3.	Ja
6	Tekenen kaartfiguren	De figuren worden juist, volgens de textuele voorstelling, getekend op het speelveld.	Ja

Toelichting testcase 1:

Bij het testen van de timer functionaliteit van de timer binnen de applicatie bleek de tijd initieel nog door te lopen op het eindscherm. Als oplossing hiervoor heb ik de updateTijd functie geschreven die tenslotte enkel in de draw zit wanneer het spel actief is. Wanneer het spel dus is afgelopen wordt de variabele tijd niet meer geupdatet en blijft de tijd dus staan op de laatste waarde. Deze waarde kan tenslotte opgenomen worden in de highscore lijst samen met de behaalde score.

Toelichting testcase 2:

Doordat alle kaarten in een array zitten en deze uit de array worden gehaald wanneer het kaart in het spel is geweest komt het natuurlijk voor dat de kaartenstapel array op een gegeven moment leeg is. Zodra de applicatie tenslotte, bijvoorbeeld, na een gevonden set een kaart wilt tekenen krijgen we een NullPointerException. Als oplossing hiervoor is er een default kaart aangemaakt welke enkel de achtergrond van de kaarten nog laat zien wanneer het speelveld niet meer kan worden aangevuld met nieuwe kaarten.

### 5.2 TESTRAPPORTAGE EN DEBUGGING

De testcases in 5.1 zijn onder andere meegenomen tijdens het ontwikkelen van de applicatie. Enkele testen slaagden niet en vereiste extra werk om te kunnen repareren. Tijdens de ontwikkeling is er



veelvuldig getest. Dit gebeurde in ieder geval na iedere implementatie van nieuwe functionaliteit. Het testen is op verschillende niveaus gebeurd, namelijk de volgende drie:

- Debug functionaliteit binnen Processing.
- Laten loggen van data naar het console binnen Processing.
- Het functioneel testen van de applicatie.

Tenslotte is voor de snelheid van het testen en debuggen nog een aantal sneltoetsen (cheats) mee opgenomen in de code van het spel. Middels bijvoorbeeld de F10 toets worden er drie kaarten gehint in plaats van twee. Dit zorgt ervoor dat er snel door de functionaliteiten van het spel heen gegaan kan worden. Met name bij het testen of het spel vanzelf eindigt alsmede het tellen van het aantal sets op het speelveld is dit ten goede gekomen. Tenslotte is er ook nog de F12 toets wat het spel direct naar het eindscherm laat gaan. Dit is met name praktisch geweest bij het ontwikkelen en testen van het eindscherm.

## 6 CONCLUSIE EN REFLECTIE

---

TODO

## 7 BRONNEN

---

<https://processing.org/reference/>

<https://pencil.evolus.vn/>

<https://www.draw.io/>

<http://www.carrieretijger.nl/functioneren/communiceren/schriftelijk/modellen/rapport>

Reader SPD 3.5