qwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnm

|  |
| --- |
| Designdocument vier op een rij  4-11-2016  Dibran Dokter, 587252  HAN ICA Arnhem  SPD I1D  *Rein Harlé*  *Versie 1.0* |

**Inhoudsopgave**

[1. Inleiding 3](#_Toc1274251543)

[2. Analyse 4](#_Toc1203308991)

[2.1 schermontwerpen 4](#_Toc187677572)

[2.2 Globale flow 7](#_Toc1289209385)

[2.3 Schermuitleg 8](#_Toc967943717)

[2.4 Functionaliteiten 9](#_Toc854573134)

[3. Ontwerp 10](#_Toc22534749)

[3.1 Softwarebeschrijving 10](#_Toc289869318)

[3.1.1 Globale structuur 10](#_Toc2098069748)

[3.1.2 Programma flow 10](#_Toc1457669931)

[3.1.3 Variabelen en functies 11](#_Toc680664922)

[3.1.4 Events 12](#_Toc191861080)

[3.2 Globaal stappenplan 12](#_Toc1823914347)

[4. Testcases 13](#_Toc2111215831)

[5. Testrapportage 14](#_Toc1177849238)

[5.1 Conclusie 14](#_Toc932259799)

# 1. Inleiding

Dit document gaat over het maken van het programma vier op een rij.

Het doel van dit document is om duidelijk aan te geven wat er gemaakt gaat worden en waaraan voldaan moet worden en in welke volgorde.

Ook is hier informatie over de structuur en flow van het programma te vinden, door uitleg over onder de variabelen en functies en hoe deze aan elkaar verbonden zijn.

Hiernaast zijn ook de functies van het programma geformuleerd.

Ten slotte zijn de testcases gedefinieerd en getest.

# 2. Analyse

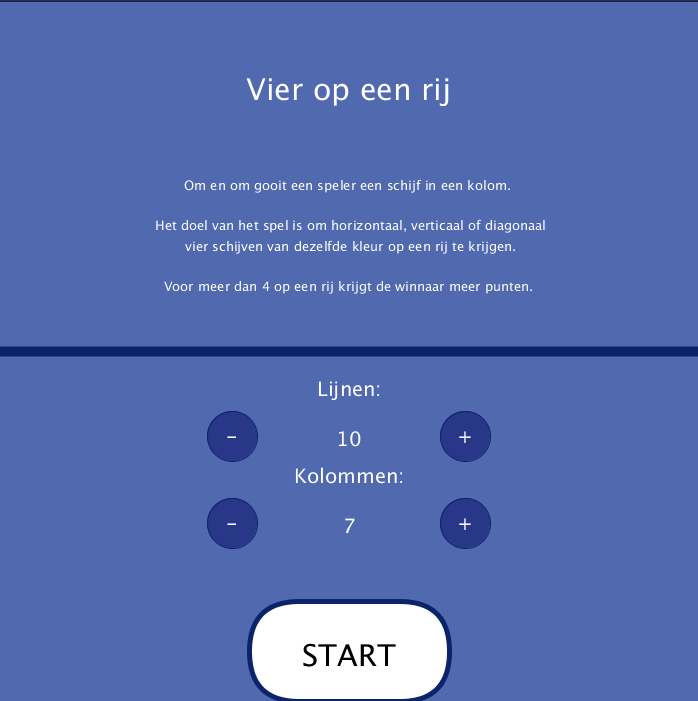
## 2.1 schermontwerpen

**Startscherm**

Als het spel gestart is toon het programma een startscherm. Het nut van dit scherm is om de speler de regels van het spel te laten zien en de mogelijkheid te geven om zelf het aantal lijnen en kolommen in te stellen door op de “+” en “-” knoppen te klikken.

Wanneer er op start geklikt word gaat het programma over naar het spelscherm.

Afbeelding 1: *Startscherm.*



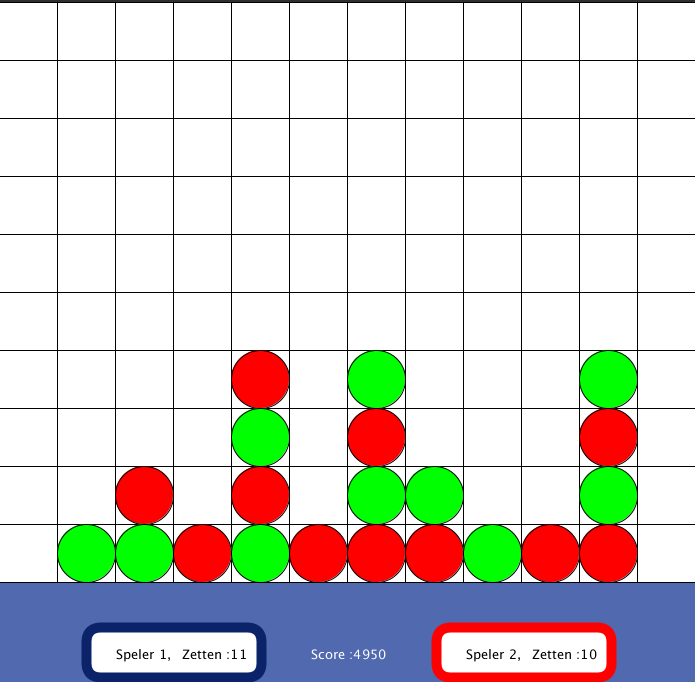
**Speelscherm**

Het spelscherm is waar het spel zelf gespeeld word, er kan op de vierkanten geklikt worden om de schijven te plaatsen. Ook staat er onderaan duidelijk aangegeven welke speler aan de beurt is door middel van een rode rand om de speler.

Ook staat hier het aantal zetten per speler en de score.

Als er vier of meer op een rij is gaat het programma over naar het scorescherm.

Afbeelding 2: *Speelscherm.*



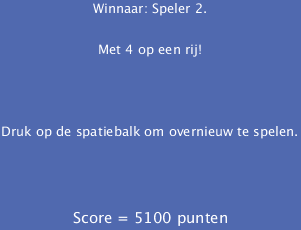
**Scorescherm**

Het scorescherm toont de winnaar en hoeveel schijven er op een rij zijn gescoord.

Ook is hier de score te zien en een mogelijkheid gegeven om het spel opnieuw te spelen door op de spatiebalk de drukken.

Als dit gedaan word gaat het programma terug naar het startscherm en onthoudt alleen de ingestelde lijnen en kolommen.

Afbeelding 3: *Scorescherm.*



## 2.2 Globale flow

Wanneer het spel gestart is laat deze het startscherm zien en kunnen de lijnen en kolommen ingesteld worden. Wanneer er op start gedrukt is word het bord gemaakt en getekend op het spelscherm.

Als er in een hokje geklikt is word er gekeken welke ruimte in de gekozen kolom leeg is en word deze gevuld met de kleur van de speler.

Ook word er gekeken of er vier op een rij is door vanaf de gekozen positie naar links en rechts dezelfde kleur schijven te tellen, en deze waarden daarna op te tellen.

Als deze waarde gelijk of hoger dan vier is is er vier op een rij.

Hetzelfde word verticaal en diagonaal gedaan.

Als er geen vier op een rij wordt geconstateerd gaat de beurt over naar de volgende speler.

De score wordt berekend door middel van de volgende berekening:

De spelhoogte wordt keer de spelbreedte gedaan om alle vakken te tellen. Dit wordt word weer keer een zogenaamde multiplier gedaan (in dit geval 50). Elke zet die er gemaakt wordt word er een vak van de score afgehaald.

Als het spel over is krijgt de winnaar extra punten voor hoeveel hij op een rij gescoord heeft (in dit geval 100 per extra op een rij).

Dit geeft de volgende berekening:

(((spelhoogte \* spelbreedte) \* 50) - 50 \* aantal zetten) + (100 \* aantal op een rij).

## 2.3 Schermuitleg

Hier worden de mogelijke acties en gevolgen binnen de schermen uitgelegd.

**Startscherm:**

Het startscherm is om te laten zien aan de speler wat het doel van het spel is en het instellen van de lijnen en kolommen van het spel.

Er kan op de plus en min koppen van de lijnen en kolommen geklikt worden, de lijnen kunnen niet lager dan 7 en de kolommen niet lager dan 10. Omdat het spel schaalt als het spelbord groter wordt zouden de lijnen en kolommen oneindig ingesteld kunnen worden. Om ervoor te zorgen dat de vakken groot genoeg blijven is er een maximum van 50 bij 50 ingesteld.

Ook is er een knop om het spel te starten, als hierop geklikt wordt gaat het programma over naar het spelscherm en neemt de ingestelde lijnen en kolommen als instellingen voor het spelbord.

**Spelscherm:**

Op dit scherm staat het spelbord, deze is geschaald zodat het deze op het scherm past.

Wanneer er in een vak geklikt is komt er in het eerste lege vakje vanaf onderen een schijf met een kleur afhankelijk van welke speler er aan de beurt is.

Dit is rood voor speler 1 en groen voor speler 2.

Hier staan ook twee vierkanten met de namen van de spelers en het bijhorende aantal zetten. Als een speler aan de beurt is heeft zijn vierkant een dikke rode rand.

Tussen deze twee vierkanten staat de huidige score.

Als er vier of meer op een rij word gescoord gaat het programma over naar het scorescherm.

**Scorescherm:**

Dit scherm word getoond als het spel over is en de winnaar bekend is.

Hier wordt de winnaar getoond en met hoeveel schijven op een rij deze gewonnen heeft.

Ook wordt hier de gehaalde score van de winnaar getoond.

Verder is hier de mogelijkheid om overnieuw te spelen door op de spatiebalk te drukken aangegeven.

Wanneer er op de spatiebalk gedrukt wordt worden alle waardes teruggezet naar de waarden zoals ze voor het spelen van het spel waren en het spelbord wordt geleegd, wel blijven de ingestelde lijnen en kolommen dezelfde waardes behouden.

Daarna toont het programma weer het startscherm met de ingestelde rijen en kolommen.

## 2.4 Functionaliteiten

De volgende functionaliteiten zijn in het programma verwerkt:

* Twee spelers

Er word gespeeld door twee spelers, deze zijn om de beurt aan zet en hebben een verschillende kleur

* Aantal lijnen en kolommen instelbaar

Het aantal lijnen en kolommen zijn instelbaar van de minimale waarden tot 50 bij 50.

Er kunnen niet minder of meer lijnen en kolommen worden ingesteld dan de minimale en maximale waarden.

* Checken hoeveel op een rij

Er wordt gekeken hoeveel er op een rij zijn als de speler klikt, er word ook gekeken of er meer dan vier op een rij is.

* Gedane zetten per speler bijhouden

De gedane zetten in totaal en van beide spelers worden bijgehouden.

* Duidelijk aangeven welke speler aan zet is
* Scorescherm met winnende speler en score

Ook laat dit scherm het aantal op een rij van de winnende speler zien.

* Berekenen score aan de hand van het aantal zetten

Het behalen van meer dan vier op een rij levert meer punten op

Eventuele toevoegingen:

* Timer voor het plaatsen van een schijfje

De spelers moeten binnen een bepaalde tijd het schijfje plaatsen anders vergaat de beurt naar de andere speler.

# 3. Ontwerp

## 3.1 Softwarebeschrijving

### 3.1.1 Globale structuur

In dit hoofdstuk word de globale structuur van het programma besproken, het gaat hier onder andere over de gekozen tabs en welke functies hierin horen.

De gekozen tabs zijn als volgt:

**-Vier op een rij:**

Dit is de stam tab waarvan alle functies worden aangeroepen.

In deze tab worden de globale variabelen gedeclareerd en geïnitieerd waar nodig.

Ook staan hier de mousePressed en Draw functie.

De mousePressed functie voert dingen uit wanneer er op een knop word geklikt zoals de startknop of de knoppen om de lijnen en kolommen in te stellen.

Bij het spelen van het spel word hier bepaald in welke kolom de speler heeft geklikt en word het schijfje op het bord gezet.

Ook word hier bijgehouden hoeveel zetten er per speler zijn gedaan en welke speler aan de beurt is.

**-Functies:**

Hier staan alle functies die nodig zijn voor het berekenen van vier op een rij en het plaatsen van de schijfjes.

Deze functies hebben een returnwaarde die gebruikt wordt bij de stam tab.

**-Tekenfuncties:**

Hier staan alle tekenfuncties in die geen returnwaarden hebben.

Dit zijn de tekenfuncties van het startscherm, speelscherm en scorescherm.

Ook staat het tekenen van het speelveld in deze tab omdat deze geen returnwaarde geeft.

### 3.1.2 Programma flow

In dit hoofdstuk word de flow van het programma beschreven, dit gaat onder andere over wanneer en hoe de functies worden aangeroepen.

Wanneer het programma start word de functie tekenStartscherm aangeroepen, deze tekent het startscherm met alle knoppen en teksten van dien.

Wanneer er dan op een bepaalde knop geklikt word word dit geregistreerd door de bijhorende statement in de functie mousePressed. Daar worden de waardes van de lijnen en kolommen ook verlaagd en opgehoogd.

Als er op de startknop geklikt wordt word het spel gestart en worden de functies tekenSpeelscherm en tekenSpeelbord aangeroepen vanuit de draw functie.

Als er daarna op het bord geklikt word word eerst de functie volgendeRuimte aangeroepen, deze geeft aan wat de volgende vrije ruimte in de gekozen kolom is.

Daarna word de schijf geplaatst en word de functie geefOpeenrij aangeroepen met de rij en kolom van het gekozen veld.

Deze functie geeft als returnwaarde de hoeveelheid op een rij die deze speler heeft.

Als deze waarde gelijk is aan of hoger is dan 4 word de functie tekenScorescherm aangeroepen binnen de draw functie.

### 3.1.3 Variabelen en functies

Dit hoofdstuk gaat over de gekozen variabelen en functies, hier word toegelicht wat deze doen.

Ik heb gekozen om de waardes die niet veranderen als final variabelen te declareren en om dit duidelijk aan te geven deze met hoofdletters te definiëren.

Lijst met variabelen:

|  |  |  |
| --- | --- | --- |
|  | Variabel | Functie |
|  | int spelhoogte = 10, spelbreedte = 7: | Ingestelde bordhoogte en breedte. |
|  | int speler = 1: | Wie er aan de beurt is. |
|  | final int MINSPELHOOGTE= 7, MINSPELBREEDTE= 10  final int MAXSPELHOOGTE = 50, MAXSPELBREEDTE = 50: | Minimale en maximale waarde van de instelbare lijnen en kolommen. |
|  | int blokgrootte: | Blokgrootte, deze schaalt als er meer lijnen en rijen gekozen worden. |
|  | int[][] bord: | Initiëren van het tweedimensionale array voor het spelbord. |
|  | bord = new int [spelhoogte] [spelbreedte]: | De hoogte en breedte van het spelbord aangeven in het array. |
|  | boolean gestart = false: | Of het spel gestart is. |
|  | int aantalZetten = 0: | Het aantal gemaakte zetten. |
|  | final int PUNTENPERZET = 50 | De score die per geplaatste schijf verloren gaan. |
|  | int hoogsteScore: | hoogste haalbare score. |
|  | final int OPEENRIJSCORE = 100: | Toegevoegde score voor meer op een rij. |
|  | int winnaar: | Wie de winnaar is. |
|  | int aantalopeenrij: | Hoeveel er op een rij is. |

Tabel 1: *Lijst met variabelen.*

Lijst met functies:

|  |  |  |
| --- | --- | --- |
|  | Functie | Doel |
|  | Int volgendeRuimte(int kolom) | Volgende lege ruimte in de gekozen kolom geven. |
|  | Int geefOpeenrij(int gekozenrij, int gekozenkolom) | Hoeveel op een rij teruggeven als het 4 of meer is. |
|  | void resetSpel(): | Alle waardes terugzetten naar de standaardwaarden en het spel herstarten. |
|  | void tekenStartscherm() | Het startscherm tekenen. |
|  | void tekenSpelbord() | Het spelbord tekenen met alle schijven. |
|  | Void tekenSpelscherm() | Het spelscherm tekenen. |
|  | Void tekenScorescherm() | Het scorescherm tekenen. |

Tabel 2: *Lijst met functies*

### 3.1.4 Events

In dit hoofdstuk staat hoe het programma op de events moet reageren.

Het klikken op de plus-en-min knoppen moet het aantal lijnen en kolommen verhogen en verlagen.

Het klikken op de startknop moet het spelscherm laten zien met het spelbord en de scoren en zetten van de spelers.

Het klikken op het spelveld moet een schijfje met de kleur van de speler in de laagst mogelijke rij van de gekozen kolom plaatsen

Als er vier op een rij is moet het scorescherm getoond worden met de winnaar en de score.

Als er op de spatiebalk gedrukt wordt terwijl het scorescherm getoond wordt moet het het spel terug gaan naar het startscherm.

## 3.2 Globaal stappenplan

In dit hoofdstuk staat beschreven in welke volgorde het programma wordt geïmplementeerd.

- Eerst maak ik het startscherm met de benodigde informatie, daarna de knoppen om het spelveld in te stellen.

- Hierna implementeer ik dat er twee verschillende kleuren per speler om en om in de vakken geplaatst kunnen worden.

- Daarna zorg ik ervoor dat deze kleuren vanaf onderin het veld naar boven geplaatst worden.

- Daarna maak ik de functie die ervoor zorgt dat er meer dan vier op een rij gecontroleerd wordt en de winnaar aangegeven word.

- Daarna zorg ik ervoor dat het spel herstart kan worden.

- Dan tel ik het aantal zetten en teken ik deze informatie op het speelscherm.

- Daarna implementeer ik de score telling en laat deze zien op het scorescherm.

Hierna kunnen eventueel nog extra functionaliteiten geïmplementeerd worden.

# 4. Testcases

In dit hoofdstuk worden de testcases gedefinieerd, hierbij wordt rekening gehouden met de functionaliteiten die het programma moet hebben.

Testcases:

-Bij het instellen van het aantal lijnen en kolommen blijven deze binnen de gedefinieerde minimale en maximale waarden.

Dit kan getest worden door op de min knop te drukken als deze al op de minimale waarde staat, of op de plus knop als deze al op de maximale waarde staat.

Het gewenste resultaat is dat de waarde hetzelfde blijft.

Hetzelfde kan voor de andere waardes gedaan worden.

-Als er buiten het speelveld wordt geklikt komt er geen schijfje bij.

Dit kan getest worden door zowel rechts als onder het speelveld te klikken, het gewenste resultaat is dat er geen schijfje bij komt en zich geen errors voordoen.

-Als een kolom helemaal vol zit kan de beurt niet verdaan worden door nogmaals op dezelfde kolom te klikken.

Dit kan getest worden door een kolom helemaal te vullen en dan nogmaals op dezelfde kolom te klikken, het gewenste resultaat is dat er niks gebeurt.

-De score telling is juist en telt ook extra punten wanneer er meer dan vier op een rij is gescoord.

Dit kan getest worden door zelf naast het spel de score bij te houden aan de hand van de berekening en deze te vergelijken met de eindscore.

-Alle soorten op een rij voor beide spelers zorgen ervoor dat het spel gewonnen word.

Dit kan getest worden door alle verschillende manier van op een rij te testen, het gewenste resultaat is dat ze er allemaal voor zorgen dat het spel gewonnen wordt en het scorescherm de juiste score laat zien.

-Het spel wordt gewonnen door de juiste speler.

Dit kan getest worden door het spel te winnen door beide spelers, het gewenste resultaat is dat de juiste speler op het scorescherm als winnaar staat.

# 5. Testrapportage

-Bij het instellen van het aantal lijnen en kolommen blijven deze binnen de gedefinieerde minimale en maximale waarden.

De minimale en maximale waardes worden niet overschreden.

-Als er buiten het speelveld wordt geklikt komt er geen schijfje bij.

Als er buiten het spelbord geklikt wordt gebeurt er niks.

-Als een kolom helemaal vol zit kan de beurt niet verdaan worden door nogmaals op dezelfde kolom te klikken.

Alle kolommen kunnen volledig ingevuld worden en er word geen zet geteld als er nogmaals geklikt wordt.

-De score telling is juist en telt ook extra punten wanneer er meer dan vier op een rij is gescoord.

De eindscore met meer op een rij is hetzelfde als ik zelf berekend had aan de hand van de berekening

-Alle soorten op een rij voor beide spelers zorgen ervoor dat het spel gewonnen word.

Alle soorten rijen werken en meer op een rij werkt.

-Het spel word gewonnen door de juiste speler.

De juiste spelers winnen.

## 5.1 Conclusie

De meeste tests waren gelijk voldoende maar het klikken buiten het scherm zorgde nog voor wat problemen, eerst werd dit ook als een kolom gezien en kreeg ik een out-of-bounds error. Dit heb ik opgelost door een controlestructuur.

Toen kreeg ik alsnog out-of-bounds errors maar nu alleen als ik op specifieke plekken klikte, dit kwam omdat er niet gecheckt werd of het spel gestart was bij de “+” en “-” knoppen van de rijen en kolommen.

Hiervoor heb ik ook een controlestructuur geïmplementeerd.

Het maken van het spel ging redelijk goed maar het vinden van een goede manier op meer dan vier op een rij te controleren zorgde nog voor wat moeite.

Ik heb zeker veel geleerd van het maken van deze opdracht, zoals debugging en het maken van heldere code.