



# EXTENDED ABSTRACT

Vier op een rij

Datum: 8 oktober 2021

Opleiding: HBO-ICT, Zuyd Hogeschool

Auteurs: Manon Schüle(1805193)

Clara Schoelitsz(1427709)

Anthony Schuman(1939319)

Docent: Drs. Snijders

# Inhoud

1. Abstract .....	3
2. Introductie .....	4
2.1. Aanleiding .....	4
2.2. Doelstelling .....	4
3. Theoretisch Kader. ....	5
4. Methode .....	6
5. Resultaten .....	8
5.1. Programma van Eisen .....	8
5.2. Onderzoek .....	8
5.3. Proof of Concept .....	9
5.3.1. Use-case .....	9
5.3.2. ERD .....	10
5.3.3. Het spel .....	11
5.3.4. Testen .....	13
6. Discussie .....	14
7. Conclusie .....	15

## 1. Abstract

De firma “4 op een rij online” heeft een Proof of Concept gevraagd aan de firma “Wij maken 4 op een rij online” om zo een algoritme te verkrijgen waarmee het spel 4 op een rij gespeeld kan worden. De firma 4 op een rij online wil dat het spel wereldwijd en door iedereen met een internetverbinding gratis gespeeld kan worden. Het spel moet door één speler en door twee spelers gespeeld kunnen worden. Het algoritme moest onder andere van de speler kunnen winnen. Verder moet het spel de originele regels volgen van het bordspel 4 op een rij. Het Proof of Concept bevat dan ook deze eisen.

Om dit te realiseren is er aan de hand van de Kitchenham methode en de DSR design research methode gezocht naar bestaande oplossingen voor het spel. Daarna zijn er modellen ontworpen om de code te kunnen realiseren.

## 2. Introductie

### 2.1. Aanleiding

Veel applicaties en programma's zijn afhankelijk van artificial intelligence en algoritmes. Voor bedrijven is het interessant dat hun werknemers hier een grip op hebben. Om de eerste stap te zetten in de wereld van algoritmes, wordt er een bestaand spel gerealiseerd.

Een bedrijf ziet graag een Proof of Concept versie van een console applicatie waarmee je vier op een rij tegen een andere speler en tegen een computer kan spelen. Dit spel is een bekende in de wereld van spel algoritmes en leent zich daarom goed naar onderzoek. Voor het bedrijf is het interessant om hier een werkende Proof of Concept van in te zien.

### 2.2. Doelstelling

Voordat "4 op een rij online" live wilde gaan, wilde zij eerst een Proof of Concept zien van de projectgroep. Dit hoefde nog geen mooie interface te hebben. Maar het spel moest wel werken. Zo moet het spel gespeeld kunnen worden door twee spelers of één speler tegen de computer. De computer mag niet altijd winnen, maar moet wel slim genoeg zijn om de winnende zet van spelers te blokkeren. Verder moet het bord moest 6 bij 7 groot zijn.

Het was voor de projectgroep dan ook belangrijk om een werkend algoritme op te leveren doormiddel van een Proof of Concept. 4 op een rij online zou dan zelf dit Proof of Concept uitwerken. Zij zullen hier een website voor creëren met een mooie grafische interface.

De Proof of Concept moest demonsteren dat het spel door één speler te spelen is tegen de computer, of door twee spelers tegen elkaar. Het moest duidelijk zijn wie welk steentje heeft. Spelers moesten hun spelersnaam kunnen opslaan zodat ze deze later weer konden kiezen. Hierdoor zouden er ook resultaten bijgehouden kunnen worden zodat spelers kunnen zien hoe vaak zijn gespeeld hebben en hoe vaak zij gewonnen, verloren of gelijk gespeeld hebben.

Verder moet het spel de originele regels volgen zoals stoppen bij 4 op een rij of gelijkspel en moet het steentje altijd in het onderste vrije vakje van de gekozen rij vallen zodat deze niet ergens blijft zweven.

### 3. Theoretisch Kader.

Uit het onderzoek naar de spelregels zijn de volgende regels geconcludeerd.

- Een speler heeft 21 steentjes.
- De spelers spelen om de beurt.
- Een vier op een rij win kan diagonaal, verticaal of horizontaal.
- Er is een gelijkspel als de steentjes op zijn.

Ook is gebleken dat er twee verschillende soorten borden bestaan, namelijk een bord van 8 kolommen en 8 rijen, en een bord van 7 kolommen en 6 rijen. De tweede is bedacht door Hasbro en is het spel waar het team het meest bekend mee is.

Een voorbeeld uitwerking van het spel in python laat een aantal methodes zien voor omringende functies voor het spel, zoals het plaatsen van een steentje, het bepalen van de beurten en het aanmaken van het bord. Voor het bord kan een matrix worden gebruikt, met strings in strings.

Over het algoritme is er literatuur gevonden die omschrijft hoe andere het spel in het verleden hebben gerealiseerd.

Met name wordt er veel gebruikt gemaakt van het minimax algoritme. Dit algoritme is veelvoorkomend in de wereld van spellen, zeker omdat de diepte van dit algoritme ook een moeilijkheidsgraat laat instellen; hoe dieper het algoritme kan gaan, hoe moeilijker het wordt om de computer te verslaan.

Volgens de literatuur is dit concept gebaseerd op de Monte Carlo tree search methode. Het idee is dat de computer een score toedient aan de verschillende zetten, en degene met de hoogste score pakt gebaseerd op een aantal factoren. Factoren zoals of de speler al 3 steentjes op een rij heeft staan. Hoe dieper het algoritme gaat, hoe meer stappen hij vooruit kan denken, door bijvoorbeeld ook te kijken of er ergens twee steentjes op een rij staan. Zijn missie is om zijn score steeds hoger te maken.

## 4. Methode

Het onderzoek in combinatie met een interview met de opdrachtgever, waarbij LSD is toegepast, heeft geleid tot de requirements die de scope vormen van wat er gemaakt moet worden voor de applicatie. Het interview diende om te begrijpen wat de opdrachtgever verwachtte van het product, zeker als het ging om de database.

Op basis van het interview was het programma van eisen gemaakt. Hier stond overzichtelijk en geprioriteerd welke eisen de opdrachtgever aan het Proof of Concept stelde. De eisen zijn geprioriteerd met de 100 punten methode.

Om de opdracht met succes uit te voeren, is de DSR-methode toegepast.

Na dat de scope en de requirements begrepen zijn, is er aan de hand van de onderzoeksmethode gezocht naar bronnen die mogelijke oplossingen bevatten. Er is voornamelijk gezocht naar oplossingen voor het algoritme en naar details over het spel zelf, om zeker te weten dat de regels zijn begrepen en dat het bord juist gemaakt wordt. Ook wees het onderzoek uit hoe complex het spel is, en waar eventuele zwakke punten zitten. Bijvoorbeeld, het is bekend dat de speler die de eerste steen plaatst, statistisch meer kans heeft om te winnen. Het vier op een rij spel bestaat al, dus werd er gezocht naar bestaande oplossingen die ter inspiratie konden bieden.

Voor het zoeken naar de bronnen is er een zoektermen lijst opgesteld. De termen zijn in verschillende zoekmachines gegooid, waar toen de resultaten zijn vast geskimmed.

Om te bepalen welke bronnen meerwaarde hebben voor ons project, is er een score systeem opgesteld. Deze is geïnspireerd door de Kitchenham methode.

De bronnen zijn beoordeeld op een aantal aspecten.

Criteria	Aantal punten
<b>Zoekwoord in de titel</b>	40
<b>Relevante onderwerpen in de samenvatting</b>	10
<b>Relevante onderwerpen in de tekst</b>	30
<b>Onderwerp geeft bijdrage aan de oplossing over het algoritme.</b>	20

De rank van een artikel is bepaald aan het aantal punten. Zo werd bepaald of een bron meerwaarde had voor het project. Als de score te laag was, werd deze ook niet meegenomen.

Als een bron voldoende scoorde, werd er een consensus gevormd over de meerwaarde, en de bron vervolgens in de referentielijst gezegd en de nodige informatie eruit gehaald.

Toen de verwachtingen en requirements rond waren, begon het ontwerpen. Hiervoor is er een use-case diagram gemaakt, en daarna een aantal UML modellen. Met deze ontwerpen is het realiseren begonnen. Voor deze fase is de Hevner methode toegepast. Bij deze cyclische werkwijze is de voortgang van de code gecontroleerd aan de hand van de gevonden literatuur, om zo de ontwikkeling te valideren.

Zo werd in stappen een stukje gecodeerd, deze vergeleken met de literatuur, vervolgens getest, en als er problemen waren of het werkte niet, dan werd er nog een keer gekeken naar de gevonden informatie.

Bij het ontwikkelen van de code, zijn een aantal ideeën uit de bronnen vertaald naar c# code en toegepast, zoals het programmeren van beurten, of het aanroepen van het algoritme op het moment dat de speler de naam computer krijgt. Toen alles geprogrammeerd was, werd er geconcentreerd op performance verbetering.

## 5. Resultaten

### 5.1. Programma van Eisen

De resultaten van dit interview zijn verwerkt in een programma van eisen. Op basis van deze geprioriteerde eisen is het Proof of Concept gebouwd.

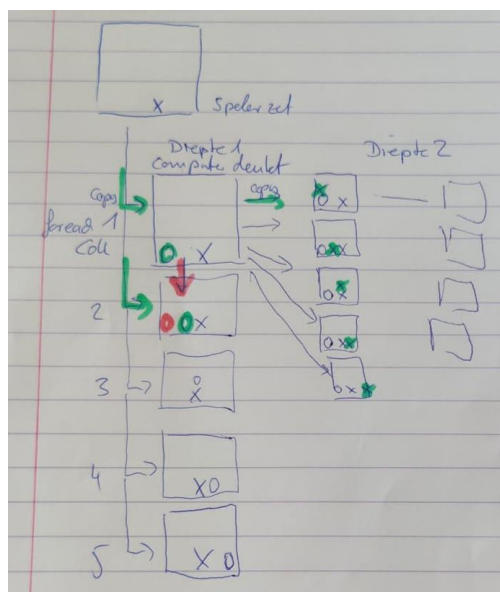
### 5.2. Onderzoek

Nadat de eisen van de opdrachtgever bekend waren, begon het onderzoek. Het onderzoek draaide om het inventariseren van nuttige bronnen en het vinden van inspiratie voor het algoritme.

Tijdens het onderzoek viel meteen op dat er meerdere bord groottes bestaan van 4 op een rij. De meest bekende is de variant van Hasbro dat 7 bij 6 is. Deze is door het projectteam dan ook aangehouden.

Uit onderzoek is gebleken dat er gebruik kan worden gemaakt van verschillende algoritmes zoals de Monte Carlo research tree. Dit is een zoek algoritme die gebruikt wordt tijdens beslissingsprocessen. Dit algoritme werd onder andere gebruikt in de autopilot van Tesla en in verschillende spellen zoals Go, schaken, en poker. Monte Carlo research tree werkt door elk van de mogelijke zetten uit te proberen die vanaf de huidige positie kunnen worden gespeeld. Het begint met het willekeurig kiezen van bewegingen, maar na verloop van tijd focust het zich op die bewegingen die het een betere kans lijken te geven om te winnen. Hierbij geldt, hoe meer tijd het algoritme ter beschikking krijgt, des te beter keuze wordt.

Echter is er gekozen voor het minimax algoritme toe te passen. Dit algoritme is volgens de gevonden informatie het beste toe te passen op spelletjes. Dit wordt ook bevestigd door de opdrachtgever. Het minimax algoritme berekend een score aan de hand van de zetten van zichzelf en de speler. Een goede zet van de speler verlaagd zijn score en dan zal het algoritme opzoek gaan naar de stap met de beste score om zijn score weer te verhogen. Zo zal hij bepalen welke zet hij moet zetten.



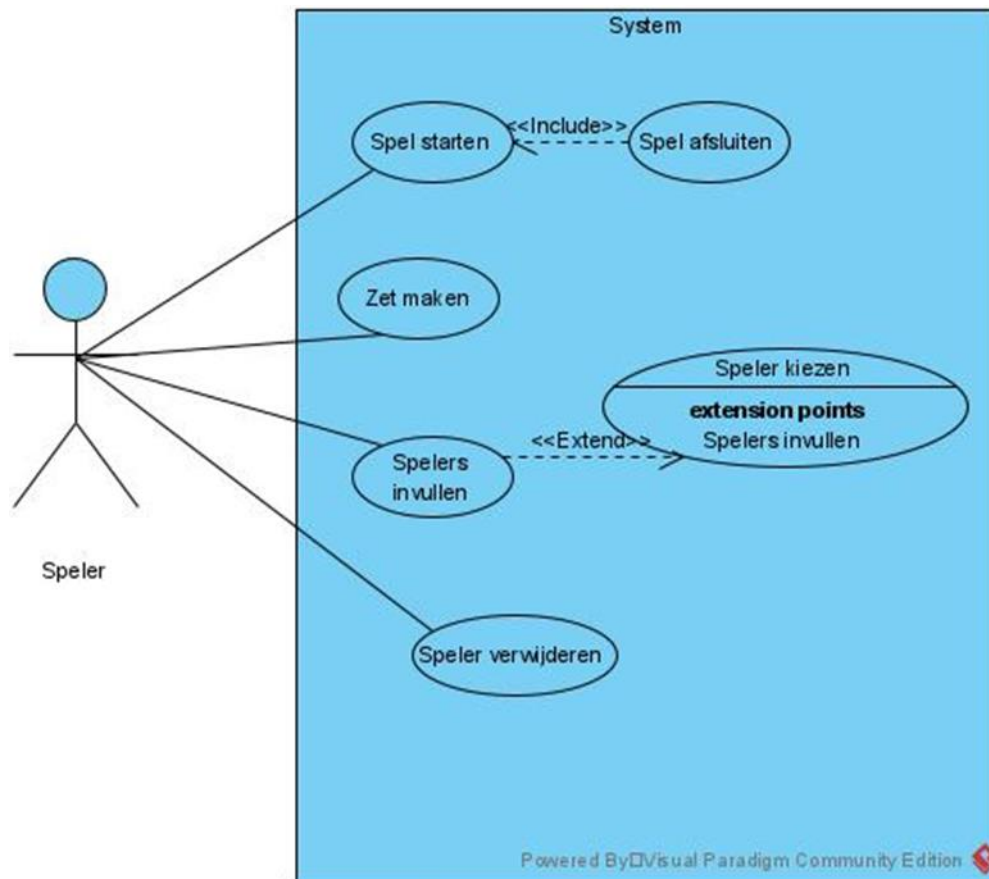
Figuur 1 – De werking van Minimax uitgetekend.



### 5.3. Proof of Concept

Er is een Proof of Concept gerealiseerd door de projectgroep van “Wij maken 4 op een rij online”. Deze 4 op een rij Proof of Concept kan gespeeld worden door twee spelers tegen elkaar of door één speler tegen de computer.

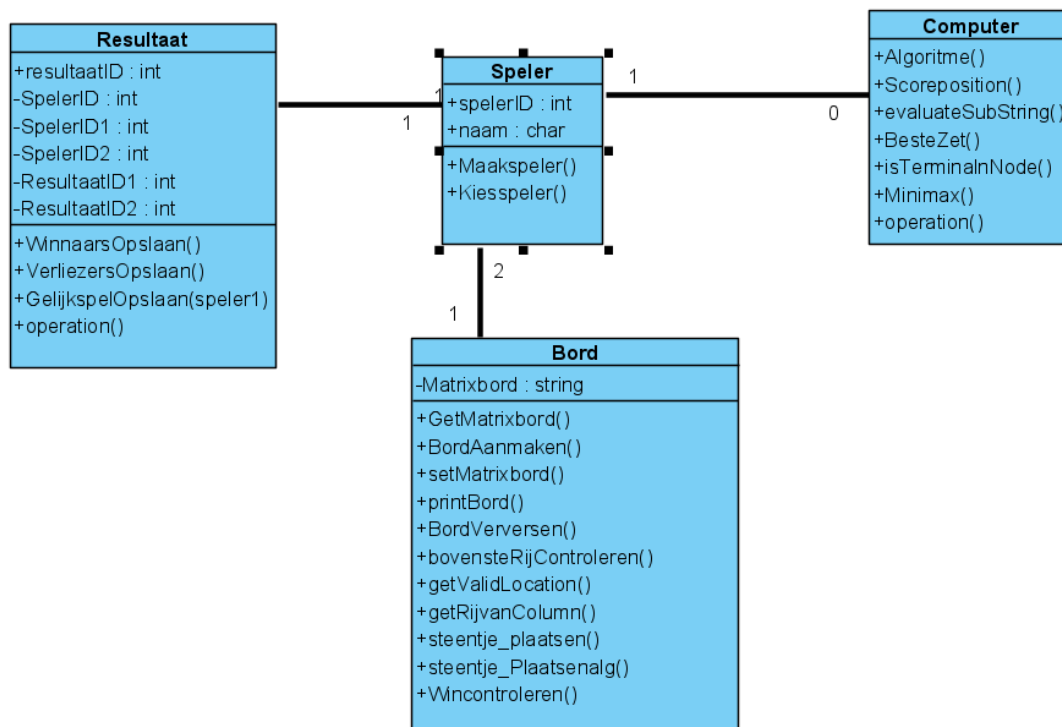
#### 5.3.1. Use-case



Figuur 2 – Use-case diagram

In dit diagram staan de hoogste geprioriteerde eisen aan het spel weergegeven, om zo te bepalen wat de speler moet kunnen om het spel te kunnen spelen. Zo kan er ook bepaald worden welke functies aanwezig moeten zijn in het algoritme.

## Klassediagram



Dit klassediagram laat de structuur van de code zien. Het “spel” is het hoofdprogramma waar alle functies en objecten in worden opgeroepen. Speler en Resultaten zijn verantwoordelijk voor het wegschrijven naar de database, en de klasse computer heeft alle functies van het algoritme.

### 5.3.2. ERD

Voor de database is een ERD ontworpen om een idee te krijgen wat de meest efficiënte manier is om de gegevens op te slaan en welke meerwaarde dat heeft voor het functioneren van het spel. De database wordt simpel gehouden met twee tabellen. Een tabel houdt alle spelers bij, en het andere slaat de resultaten op aan de hand van de spelersID. Er staat een totaal spellen, gelijkspellen en gewonnen spellen. Er staan geen verloren spellen, omdat dit herleidbaar is.



Figuur 3 - ERD

### 5.3.3. Het spel

Wanneer het spel gestart wordt, zal de speler de vraag krijgen of het een bestaande speler is of een nieuwe speler. Hierna kan de speler kiezen of hij tegen de computer wilt spelen of tegen een tweede speler. Wanneer er voor een tweede speler gekozen wordt, krijgt de tweede speler ook de vraag of het om een nieuwe speler gaat of een bestaande speler. Deze vragen worden gevraagd zodat de scores bijgehouden kunnen worden. Wanneer er gekozen wordt om tegen de computer te spelen zal het spel meteen starten.

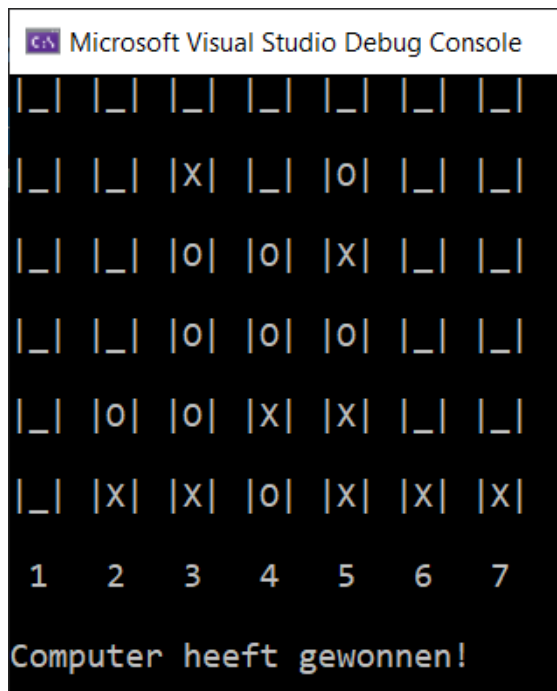
Als er gekozen wordt voor een nieuwe speler dan kan de speler elke naam tot 50 tekens invoeren. In het Proof of Concept wordt er geen controle uitgevoerd op bijvoorbeeld verboden woorden zoals scheldwoorden of op al bestaande namen. Wanneer er een bestaande naam wordt ingevoerd zal het spel starten en worden de resultaten op de zelfde naam weggeschreven. Hiervoor is gekozen omdat het voor de Proof of Concept geen hoge prioriteit eis was om een controle in te bouwen. Het systeem geeft wel een melding als er voor bestaande speler wordt gekozen maar er een naam wordt ingevoerd die niet in de database bestaat.

Resultaten worden weggeschreven naar een database, ook worden spelers opgeslagen in de zelfde database. Spelers hebben een uniek ID waarmee de speler resultaten gekoppeld kunnen worden aan de spelers. De resultaten bevatten de gewonnen spellen, de gelijkspellen en het totaal spellen. Verloren spellen zijn zo te herleiden.

De speler speelt het spel op een bord van 7 bij 6. De computer begint altijd als tweede, als het spel gespeeld wordt door twee spelers dan begint speler 1 als eerste. Speler 1 heeft altijd een kruisje en speler 2 is altijd het rondje, dit moeten de steentjes symboliseren van het bordspel. De speler die het eerste 4 op een rij heeft, horizontaal, verticaal of diagonaal wint het spel, hier komt een melding van in beeld.

De computer staat ingesteld op diepte zes omdat dit het optimaalste is qua bereken tijd. Hogere tijden vertragen het programma dusdanig dat het minder tot niet speelbaar wordt. Bij mindere dieptes wordt het te makkelijk voor de speler om te winnen van de computer. Het algoritme is gebaseerd op het Minimax algoritme.

De spelers kunnen uit 7 rijen kiezen waar het steentje geplaatst dient te worden, het steentje valt dan altijd op de onderste beschikbare plek. Wanneer er een foutieve rij wordt gekozen, zal er een melding op het scherm vertoond worden, zodat de speler weet dat hij een foutieve rij gekozen heeft. De melding zal verdwijnen wanneer de speler een juiste rij gekozen heeft en zijn beurt daardoor beëindigt is.

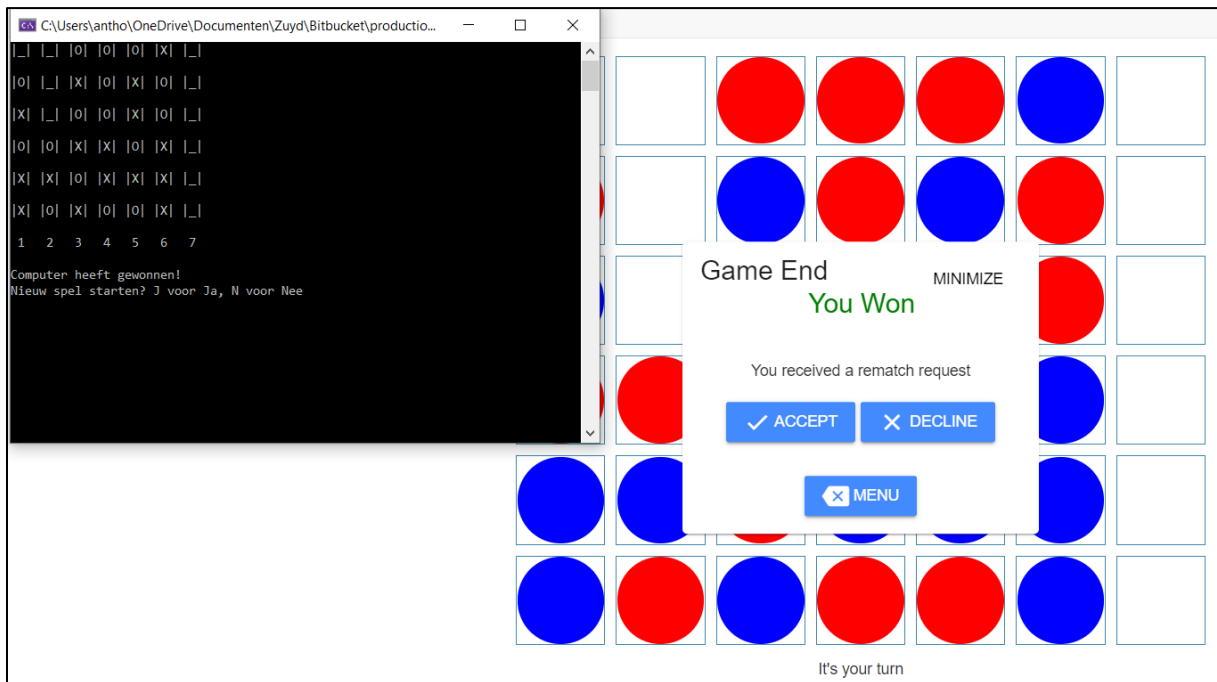


Figuur 4 – Screenshot van het spel, waarbij de computer wint.

#### 5.3.4. Testen

Uit testen is gebleken dat het spel niet altijd perfect werkt. Zo was het bij een eerdere versie mogelijk om verkeerde waarden in te voeren waardoor het spel vastliep. Dit gebeurde onder andere bij het kiezen van een rij. Wanneer er een rij kleiner dan 1 of groter dan 7 werd gekozen, stopte het spel. Ook wanneer er een spelersnaam ingevoerd werd die langer was dan 50 tekens stopte het spel.

Het algoritme is ook getest door middel van een field test. De projectgroep opende een online variant van 4 op een rij tegen andere spelers en elke zet die de andere speler invoerde, voerde de projectgroep in, het algoritme. Dit resulteerde tot meerde gewonnen spellen.



Figuur 5 – het algoritme tegen online spelers.

## 6. Discussie

Er kunnen heel veel vragen gesteld worden bij het onderzoeksmethodiek en de compleetheid van de aanpak in samenhang met de complexiteit van de opdracht. Hoe uitgebreid moet het onderzoek in contrast met de complexiteit? De Kitchenham methodiek stelt een uitgebreide controle aan het onderzoek om alle waarnemingen en conclusies te valideren, en de keuzes die uit het onderzoek komen te onderbouwen. Bij een onderwerp zoals het maken van een bekend spelletjes, dat al vaker gebouwd is, valt de onderbouwing en validatie veel in de schoot van het resultaat; als het resultaat van dit project overeenkomt met wat er al bestaat, dan is wat er al bestaat juist toegepast. Uiteraard, is het resultaat van dit project niet identiek aan andere resultaten. Het staat daarom ter discussie of een dieper onderzoek daaraan had kunnen bijdrage.

Een volgende keer zou de projectgroep de Kitchenham methodiek strenger naleven, aangezien dit de eerste keer was dat de projectgroep met Kitchenham werkte is dit in mindere mate gebeurd. Het is dan ook een aanbeveling om dit in het vervolg wel te doen.

## 7. Conclusie

Als eindresultaat is er een functioneel bordspel gecodeerd, met nog als vraagstuk hoe “functioneel” het is als het resultaat vergeleken wordt met andere resultaten. Bijgaande zijn er documenten gerealiseerd die het ontwerp en de eisen aantonen, en laten zien op welke bronnen het werk gebaseerd is.

Het programma is in staat om twee spelers tegen elkaar te laten spelen, en stelt de speler in staat tegen een computer te spelen. De computer roept een algoritme aan die bepaald welke stappen hij moet zetten om zijn win kans te vergroten. Dit algoritme is gebaseerd op de minimax, zoals omschreven in het onderzoek en ontwerp.