**ACADEMIEJAAR:**

2016 – 2017

**STUDENTEN:**

Jasper De Sutter

Hein Pauwelyn

Michiel Vermeersch

Michiel Zyde

**DOCENT:**

Johan Vannieuwenhuyse

Kevin Derudder

Stijn Walcarius



|  |
| --- |
|  |
| **WATCHFRIENDS** |
| **Project: Cross Platform Apps,**  **Frontend & Backend webdevelopment** |
|  |
|  |

# Inleiding

WatchFriends is een app die ontstaan is vanuit de gemeenschappelijke interesse van vier studenten: series bekijken. Deze studenten en heel wat andere jongeren ondervinden hierbij echter de uitdaging om te onthouden welke series zij bekijken en vooral ook aan welke aflevering van een bepaalde serie zij toegekomen zijn. Als gepassioneerde kijker is het nooit aangenaam om eerst te moeten controleren welke aflevering je nu het laatst bekeken hebt, vooraleer je kan starten met de volgende. Net daarin wil WatchFriends verandering brengen.

De applicatie is in eerste instantie bedoeld om de gebruiker toe te laten series toe te voegen aan zijn/haar watchlist en daarbij te kunnen aanduiden welke afleveringen bekeken werden. In een oogopslag kan de gebruiker zien waar hij/zij gekomen is en begaat men zo nooit meer de fout een aflevering over te slaan of een reeds geziene aflevering opnieuw te bekijken. Tevens beschikt WatchFriends over functionaliteiten om nieuwe series te ontdekken en meldingen te ontvangen wanneer een nieuwe aflevering van je favoriete serie uitgezonden wordt.

Uiteraard zijn al apps op de markt gebracht met een soortgelijke werking. Toch probeert WatchFriends zich te distantiëren door het functionele te koppelen aan het sociale aspect. De naam zegt het zelf: de app laat toe vrienden toe te voegen met als bedoeling hun watchlist te kunnen bekijken. Deze functionaliteit adresseert meteen ook een tweede probleem bij jongeren: een doorwinterde enthousiast komt ooit op een bepaald moment dat hij/zij niet meer weet welke serie als volgende te bekijken. Wanneer je dan de kans krijgt om eens een kijkje te nemen wat vrienden momenteel bekijken en met hen de discussie op gang te brengen, kan WatchFriends ook daarbij een uitstekende hulp bieden.

Ten slotte wordt er ook nog een competitief kantje toegevoegd: het is mogelijk achievements te verdienen en deze te vergelijken met die van je vrienden. Zo kan bijvoorbeeld het bekijken van je duizendste aflevering in totaal een gouden medaille opleveren op dat vlak en kan dit een vriend motiveren om ook die medaille te behalen. WatchFriends is echter niet verantwoordelijk voor verslaving of verwaarlozing van andere zaken.

**Concrete functies:**

* Bijhouden van series in de vorm van een watchlist
* Bijhouden van bekeken afleveringen per serie
* Ontdekken van nieuwe series via categorieën, acteurs, …
* Ontvangen van notificaties wanneer een nieuwe aflevering uitgezonden wordt
* Toevoegen van vrienden om ook hun watchlist raad te plegen
* Chatten met vrienden om series te bespreken
* Behalen van achievements bij bepaalde mijlpalen (aantal afleveringen, aantal minuten, …)

# Team

We zijn in totaal met vijf personen voor dit project Niels Bril, Jasper De Sutter, Hein Pauwelyn, Michiel Vermeersch en Michiel Zyde. Aangezien Niels Bril al voor een ander project de backend en front-end maakt, doet hij enkel mee aan dit project voor Cross Platform Applications. Michiel Zyde volgt enkel front-end en werkt dus ook niet mee aan de backend.

1. Hein Pauwelyn

Hein is projectleider, hij zorgt ervoor dat de tools in orde zijn (slack, github, travis) en dat de rest van het team zijn werk verricht op een juiste manier. Dit laatste gebeurt vooral door navolging van taken en projecten op github en code checks in pull requests. Hij heeft vooral gewerkt aan de code om lijsten te tonen op de home- en explore pagina, en om achievements te kunnen behalen. Daarnaast heeft hij ook sterk meegeholpen aan het implementeren van designs, en het verbinden van de TMDB api.

1. Jasper De Sutter

Jasper is developer, hij heeft de backend code aangemaakt met Express, en de front-end met Angular-CLI. Hij heeft voor authenticatie gezorgd in de backend, en de autorisatie in de front-end. Daarnaast komt de implementatie van de API in de front-end, en de implementatie van de code in de componenten van hem. Hij heeft ook de externe packages onderzocht, om te zien welke we best implementeerden.

1. Michiel Vermeersch

Michiel is ook developer, hij werkte voornamelijk aan de feed en events, en dus ook aan de implementatie van de sockets. Daarnaast zijn ook een aantal componenten door hem uitgewerkt. Hij hielp ook mee aan de back-end routes en het aanspreken van de database.

1. Michiel Zyde

Michiel werkt enkel mee aan Front-end want hij volgt geen backend, dit maakt hem de beste keuze als CSS en designer. Hij heeft alle designs voor web uitgewerkt, van wireframes tot de uiteindelijke scss files.

1. Niels Bril

Niels werkt enkel mee aan de android applicatie, maar hij zorgt voor de hosting van de webserver en database.

# Technische analyse

De mobiele applicatie wordt geschreven voor Android toestellen met behulp van Java. De api-gebaseerde backend maken we hiervoor zelf, maakt gebruik van een NoSQL database en de TMDB. TMDB is een collectie van alle series en films dat onderhouden wordt door een zeer grote developer gemeenschap. Andere gegevens zoals de achievements, volgen van vrienden en de watchlist zullen opgeslagen worden in onze eigen NoSQL database. Deze zal wordt op een cloud service zoals Azure of Amazon.

De server code wordt geschreven in Javascript, met behulp van Node.JS en het Express framework. Voor authenticatie gebruiken we een aantal packages van Passport, onze database spreken we aan via Mongoose, web sockets werken via socket.io, en om externe api calls naar TMDB te doen maken we gebruik van Request.

Authenticatie implementeren we met access-tokens, zo kunnen we zowel de app en de website op dezelfde manier implementeren. Inloggen kan via lokale accounts, met een email en een wachtwoord, of met sociale- netwerkaccounts, zoals facebook en google. We willen ook dat de gebruikers kunnen zoeken in de app naar series of andere gebruikers. Het zoeken van de series werkt via TMDB, en het zoeken van gebruikers maken we zelf.

We maken naast onze mobiele applicatie ook een responsive webapp, die dezelfde functionaliteit moet bieden als de android applicatie. Deze wordt geschreven in Typescript, met het framework Angular (2). De code voor de templates wordt geschreven in standaard HTML-syntax, met de toevoeging van Angular directives. De styles schrijven we in Sass omdat dit meer mogelijkheden biedt dan standaard CSS. Als front-end design framework gebruiken we bootstrap, aangezien we hier al vertrouwd mee zijn en omdat de stijl past bij ons design. Om de functionaliteit te behouden die dit framework biedt, maken we gebruik van ng2-bootstrap, een package die de bootstrap componenten naar Angular omzet.

Sockets zijn zeer interessant op websites, en daarom maken we er zeker ook gebruik van. We maken een feed waarin meldingen van nieuwe acties van jezelf en vrienden terecht komen. Deze acties zijn bijvoorbeeld het volgen van een serie of een vriend, of het behalen van een achievement.

Onze code wordt op github gezet om gemakkelijk samen te werken, en om alle development resources en tools op één plek te hebben. Hier werken we met milestones om taken te plannen, en issues om taken te verdelen. De projects zijn ook zeer handig om te zien wie waaraan bezig is. We hebben ook een travis build server gekoppeld om te controleren of geschreven code wel werkt voordat we het met de master branch. Deze werkt met een gulpfile waarin we code controle voor zowel front- als backend definiëren.

# Planning

Om taken in te plannen hebben we net zoals bij de android app met milestones gewerkt. Zo kunnen we de functionaliteit deel per deel implementeren, gelijklopend met het android project.

## Milestone 1

Deadline: 25 November 2016

In deze milestone beginnen we met het project, we maken de code aan en beginnen met testen. De wireframes worden nu ook al uitgewerkt.

|  |  |  |
| --- | --- | --- |
| Taak | Wie | Status |
| Beslissen wat we gaan doen | Iedereen | Voltooid |
| Project aanmaken (github) | Hein, Jasper | Voltooid |
| Code aanmaken | Hein | Voltooid |
| Gulp in orde | Jasper, Michiel V | Voltooid |
| Achievements statisch | Hein | Voltooid |
| Hosting | Niels | Voltooid |
| WBS | Michiel Z | Voltooid |
| Technische analyse | Hein | Voltooid |
| TMDB API verbinden | Hein, Jasper, Michiel V | Uitgesteld |
| Wireframes | Michiel Z | Gedeeltelijk |
| Travis | Hein | Voltooid |

## Milestone 2

Deadline: 12 December 2016

We zorgen voor een database, en onze eigen schemas voor data die we hierin gaan steken. Nu de wireframes af zijn beginnen we ook met frontend. Aangezien we in de les met express werken, veranderen we onze backend hier ook naar. We willen ook al authorisatie om te kunnen inloggen.

|  |  |  |
| --- | --- | --- |
| Taak | Wie | Status |
| Angular code aanmaken | Jasper | Voltooid |
| Mongo server | Hein | Voltooid |
| Gulp voor frontend | Jasper | Voltooid |
| Home pagina | Michiel Z, Hein | Gedeeltelijk (CSS & HTML) |
| Server naar express | Jasper | Voltooid |
| Navbar | Michiel V | Gedeeltelijk |
| Mongoose schemas | Jasper | Voltooid |
| TMDB API verbinden | Hein | Voltooid |
| Authorisatie | Jasper, Hein | Uitgesteld |

## Milestone 3

Deadline: 26 December 2016

We stappen af van systemjs en gaan naar webpack, via Angular-CLI. We beginnen met de volgende pagina’s met functionaliteit, en werken verder aan de backend. De authorisatie wordt volledig uitgewerkt in backend.

|  |  |  |
| --- | --- | --- |
| Taak | Wie | Status |
| Navbar | Michiel V | Voltooid |
| Authorisatie backend | Jasper | Voltooid |
| Design achievements | Michiel Z | Voltooid |
| Omzetten naar Angular-CLI | Jasper | Voltooid |
| Volgen van series | Hein, Michiel V | Gedeeltelijk |
| API-calls met request | Hein | Voltooid |
| Serie details pagina | Michiel V | Gedeeltelijk |
| Login & register front-end | Jasper | Voltooid |
| User shema | Hein, Jasper, Michiel V | Gedeeltelijk |
| Profiel pagina | Michiel Z, Jasper | Uitgesteld |
| Home verder uitwerken | Hein | Voltooid |

## Milestone 4

Deadline: Examen

We implementeren sockets voor de feed en events, en we maken alle resterende front-end pagina’s. Het markeren van Afleveringen als gezien moet ook werken, en achievements moeten uitgereikt worden.

|  |  |  |
| --- | --- | --- |
| Taak | Wie | Status |
| Profiel pagina | Michiel Z, Jasper | Voltooid |
| Explore pagina | Michiel Z, Hein | Voltooid |
| Settings pagina | Michiel Z, Jasper | Voltooid |
| Series details pagina | Michiel V | Voltooid |
| Seizoen details pagina | Michiel V | Gedeeltelijk |
| Feed pagina | Michiel V | Voltooid |
| Volgen van series | Jasper | Voltooid |
| Sockets | Michiel V | Gedeeltelijk |
| Autoriseren van pagina’s | Jasper | Voltooid |
| Search | Michiel Z, Hein, Jasper | Voltooid |
| Achievements verkrijgen | Hein | Gedeeltelijk |
| Volgen van gebruikers | Jasper | Voltooid |
| Following pagina | Michiel Z, Jasper | Uitgesteld |
| Token beveiliging | Hein, Jasper | Voltooid |

# Realisatie

Na een aantal problemen gehad te hebben tijdens de eerste milestones, zijn we tegen het einde toch in een sterk tempo beginnen vorderen. Helaas zijn we niet geraakt waar we wilden, maar we vinden wel dat wat we bereikt hebben al een grote prestatie op zich is. Doordat we in tijdsnood kwamen zijn een aantal van de technische elementen maar deels uitgewerkt (unit-tests en websockets)

Wel zijn we trots op de code die we geschreven hebben, het verbinden van de TMDB api bijvoorbeeld aan onze backend hebben we uitgewerkt zodat we series kunnen volgen, terwijl het ophalen van series slechts 1 request blijft. De authenticatie hebben we volledig zelf uitgewerkt, dit was een hele hoop werk maar werkt zeer goed en veilig. We hebben ook een volledig werkende feed die via sockets werkt, waarin er meldingen komen voor het volgen van vrienden en series. We hebben daarnaast ook zelf zoekfunctionaliteit uitgewerkt, die ook samen met de TMDB api werkt.

## Struikelblokken

Het grootste struikelblok was zeker dat we onszelf hebben overschat. De hoeveelheid werk die we moesten verrichten was te klein ingeschat en daardoor is niet alles volledig uitgewerkt geraakt. We kunnen gemakkelijk nog een extra milestone toevoegen met nog evenveel taken om dit project volledig af te krijgen.

We hadden zeer veel problemen om Angular te kunnen builden, en dan linken in de html met Systemjs. Om dit op te lossen hebben we geprobeerd over te stappen naar Webpack, maar dit bleek ook niet volledig te werken. Na veel zoeken en testen hebben we dan uiteindelijk Angular-CLI gevonden en dit werkte volledig vanaf het begin.

We waren al begonnen aan de backend voordat we met Express gewerkt hadden in de les, dit zorgde ervoor dat we veel code hadden die moeilijk leesbaar en moeilijk onderhoudbaar was. We hebben de code dan herschreven naar Express om dit op te lossen.

Authenticatie was een zeer groot en lastig onderdeel, met volledig nieuwe packages (Passport) die we nog niet kenden. Hierdoor duurde het een tijd voordat dit geïmplementeerd geraakte en moesten enkele taken op zich laten wachten.