|  |
| --- |
| Vanden Broek Nikos |
| Totally Not Warcraft |
| Project Prgrammeren 4 |

|  |
| --- |
| Nikos vanden broek  28-1-2015 |

Inhoud

[Opdrachtomschrijving 2](#_Toc410075909)

[Analyse 3](#_Toc410075910)

[Game UI 3](#_Toc410075911)

[Web UI 5](#_Toc410075912)

[Admin Ui 6](#_Toc410075913)

[Database Diagram 7](#_Toc410075914)

[Problemen 8](#_Toc410075915)

[Analyse 8](#_Toc410075916)

[Hibernate Lazy loading 8](#_Toc410075917)

[Servlet-mapping /\* 9](#_Toc410075918)

[Admin – Ingelogd character 9](#_Toc410075919)

[Besluit 10](#_Toc410075920)

# Opdrachtomschrijving

Mijn project is gebaseerd op een bestaande applicatie genaamd de Armory van Blizzard games. (vb: <http://eu.battle.net/d3/en/profile/LanceDH-2186/hero/40628046>)

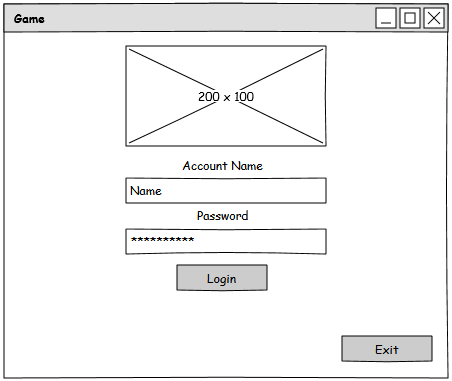
Een speler heeft een account in een game met verschillende characters.

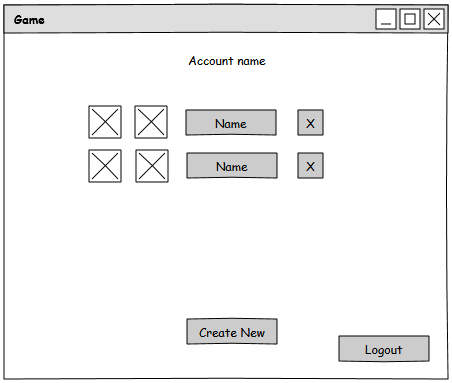
Deze characters kunnen tijdens het spelen items vinden zoals wapens en gebruiken.

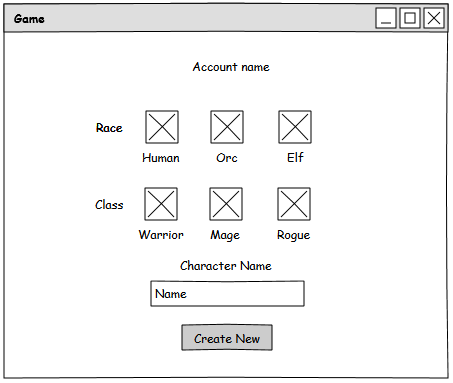
De Armory laat toe om deze characters en hun items te bekijken via het internet zonder te moeten connecteren tot het spel.

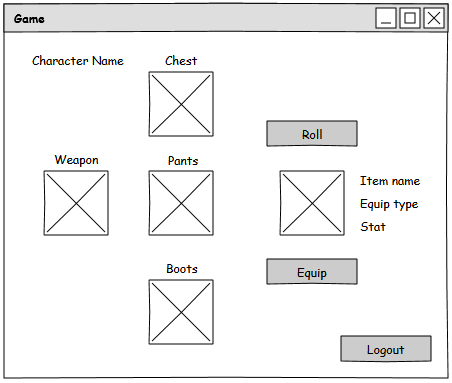
# Analyse

## Game UI

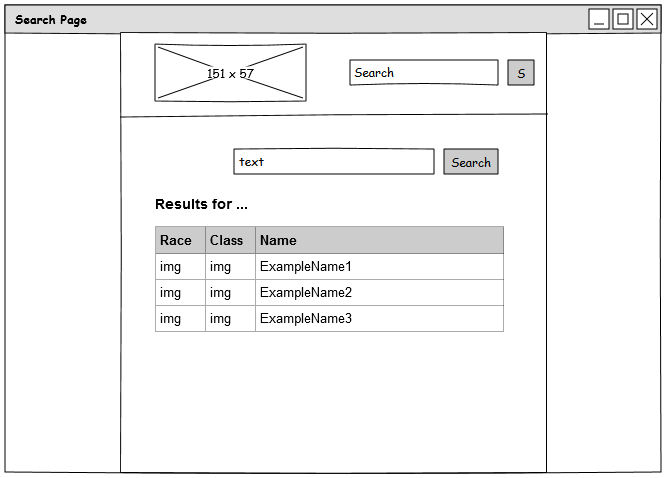


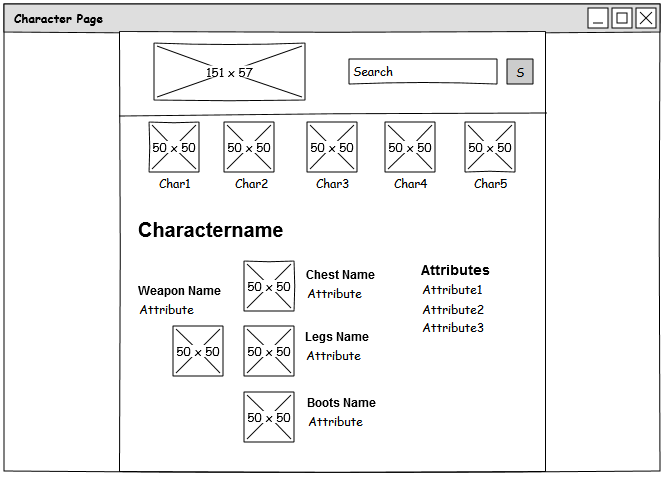




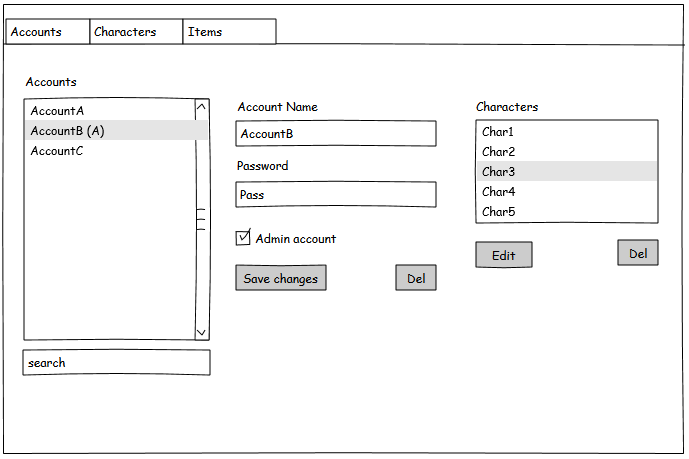


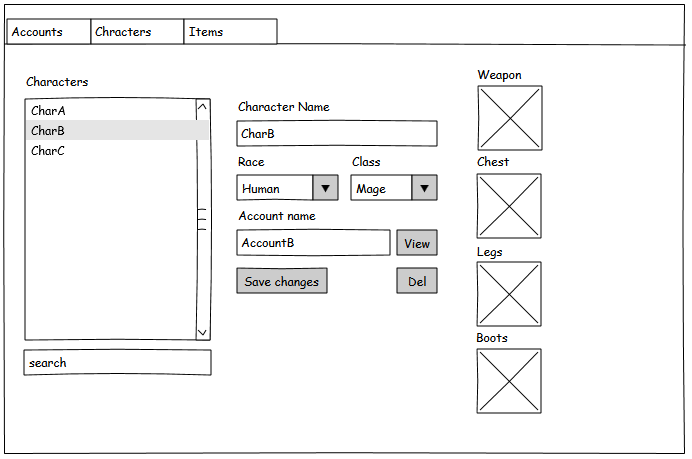
## Web UI



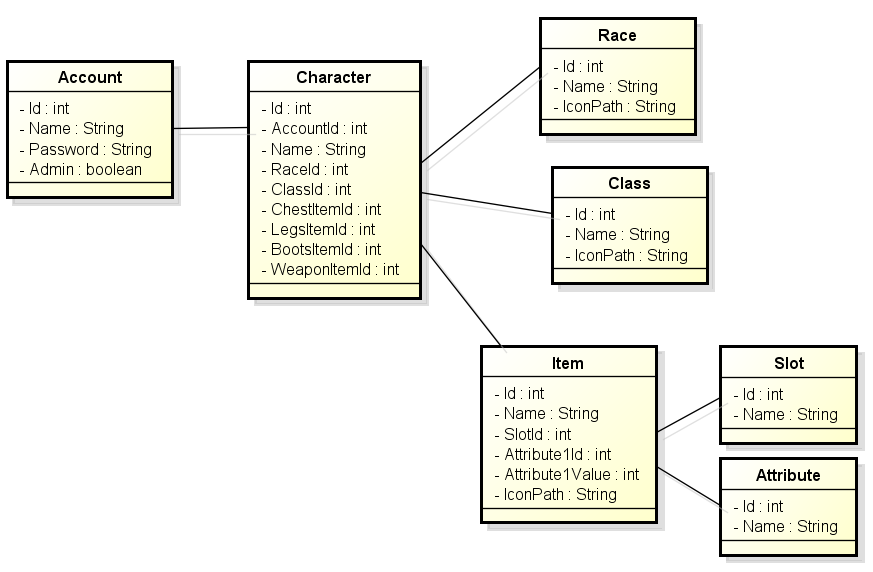


## Admin Ui





## Database Diagram



# Problemen

## Analyse

Een van de grootste problemen die ik ondervond bij dit project was de analyse.

Omdat mijn project gebaseerd is op een bestaande applicatie had ik van begin af al de uitwerking in gedachte.

Hoewel ik nog de kleine details miste had ik al een idee hoe de applicatie eruit zou zien en zou werken.

Dit zorgde ervoor dat niet meer hoefde te brainstormen naar oplossingen.

Hoewel ik al de ideeën al had slaagde ik er niet in om deze neer te pennen en zaten ze al het ware vast in mijn hoofd.

Dit zorgde er uiteindelijk voor de summiere uitwerking van het analytische deel terwijl ik er nog wel in slaagde om het project tot een goed einde te brengen.

## Hibernate Lazy loading

Al vroeg in het project had ik problemen met de Hibernate sessies waarbij er ofwel geen was ofwel meerdere sessies probeerde te openen.

Na enkele veranderingen aan de service klassen werd dit opgelost, maar kwamen er andere problemen aan het licht.

Als data uit de database wordt gehaald die een verwijzing heeft naar een andere tabel, laad Hibernate enkel een verwijzing naar die data.

Als later in de code naar die verwijzing wordt gevraagd probeert Hibernate die data uit de database te halen.

Dit echter op het moment dat er geen sessie open is en ontstaat er dus een foutmelding.

Dit hebt ik opgelost door in de query join en fetch toe te passen.

Die zorgen ervoor dat de data van de verwijzing direct meet worden op gehaald.

Vb:

“From item i left join i.slot”

## Servlet-mapping /\*

Om de verschillende pagina’s van character op te vangen maak ik gebruik van een servlet URL-pattern met een wildcard.

<url-pattern>/ViewCharacter/\*</url-pattern>

Dit zorgt ervoor dat elke URL die /ViewCharacter/ bevat naar deze servlet wordt doorgestuurd.

In deze servlet laad ik de data van het character op en stuur deze door naar een volgende pagina met volgende code:

request.getRequestDispatcher("Character.jsp");

Deze code zorgt er echter voor dat in de URL de naam van die pagina achteraan toegevoegd word wat ervoor zorgt dan de URL nog steeds opgevangen wordt door de URL-mapping en een eindeloze nesting veroorzaakt met een error tot gevolg.

Uiteindelijk bleek de oplossing zeer eenvoudig te zijn en moest er enkel een backslash worden toegevoegd voor de naam dan de pagina.

request.getRequestDispatcher("/Character.jsp");

Dit zorgt ervoor dan de URL naam terug begint vanuit de root en dus niet meer wordt opgevangen terwijl dit ook de weergegeven URL niet verranderd.

## Admin – Ingelogd character

De admin console laat toe om een character te wijzigen of verwijderen.

Dit ken echter voor problemen zorgen als het character om dat moment nog ingelogd is.

Als bijvoorbeeld een ingelogd character verwijderd wordt, zal het bij het uitloggen (update) verwezen worden naar een niet bestaande rij.

Om dit te voorkomen heb ik een extra attribuut moeten toevoegen aan de character tabel van het type boolean.

Afhankelijk of een character in of uit logt wordt deze verzet naar true of false.

Bij het wijzigen of verwijderen van een character wordt die warde eerste uit de database gehaald om na te gaan of het toegelaten is het charater aan te passen.

# Besluit

Desondanks de moeilijke start met de analyse is het resultaat toch goed uitgekomen.

Ik heb tijdens dit project zeker nog nieuwe dingen bij geleerd.

Persoonlijk vind ik dat het project nog iets beter afgewerkt mag worden maar ik ben toch trots op het resultaat.