Overdrachts-Documentatie

FontysMadeSteam

# Analyse:

# Functionaliteiten vereist:

## **Must have:**

1.1 De applicatie is toegankelijk voor iedereen.  
1.2 De gebruiker heeft een overzicht van geuploade games.  
1.3 Een gebruiker kan de getoonde games filteren op tags.  
1.4 De gebruiker kan individuele games bekijken in de applicatie.   
1.5 De gebruiker kan games downloaden via de applicatie.  
1.6 De gebruiker kan zoeken naar developers.  
1.7 De gebruiker kan zoeken naar games.

## **Should have:**

2.1 De gebruiker kan de games via de applicatie launchen.  
2.2 De applicatie geeft weer welke games gedownload zijn.

|  |  |
| --- | --- |
| **FR-01** De applicatie is toegankelijk voor iedereen. | |
| **Requirement** | **Omschrijving** |
| *K-01.1* | De applicatie is beschikbaar op MAC OS en Windows |

|  |  |
| --- | --- |
| **FR-02** De gebruiker heeft een overzicht van geuploade games. | |
| **Requirement** | **Omschrijving** |
| *K-02.1* | Naam en afbeelding worden weergegeven. |
| *B-02.1* | Dezelfde game komt niet dubbel voor. |

|  |  |
| --- | --- |
| **FR-03** Een gebruiker kan de getoonde games filteren op tags. | |
| **Requirement** | **Omschrijving** |
| *B-03.1* | De gebruiker kan games filteren op max. 5 tags tegelijk. |
| *K-03.1* | Gekozen tags kunnen gereset worden d.m.v. een knop. |
| *K-03.2* | Gekozen tags worden onthouden tot de selectie wordt verwijderd. |

|  |  |
| --- | --- |
| **FR-04** De gebruiker kan individuele games bekijken in de applicatie. | |
| **Requirement** | **Omschrijving** |
| *K-04.1* | Naam, (afbeelding(en), beschrijving, tags, developers, semester, jaar) worden weergegeven bij de gekozen game. |

|  |  |
| --- | --- |
| **FR-05** De gebruiker kan games downloaden via de applicatie. | |
| **Requirement** | **Omschrijving** |
| *B-05.1* | Er is één download link per applicatie. |

|  |  |
| --- | --- |
| **FR-06** De gebruiker kan zoeken naar developers. | |
| **Requirement** | **Omschrijving** |
| *B-06.1* | De gebruiker kan naar 1 developer tegelijk zoeken. |

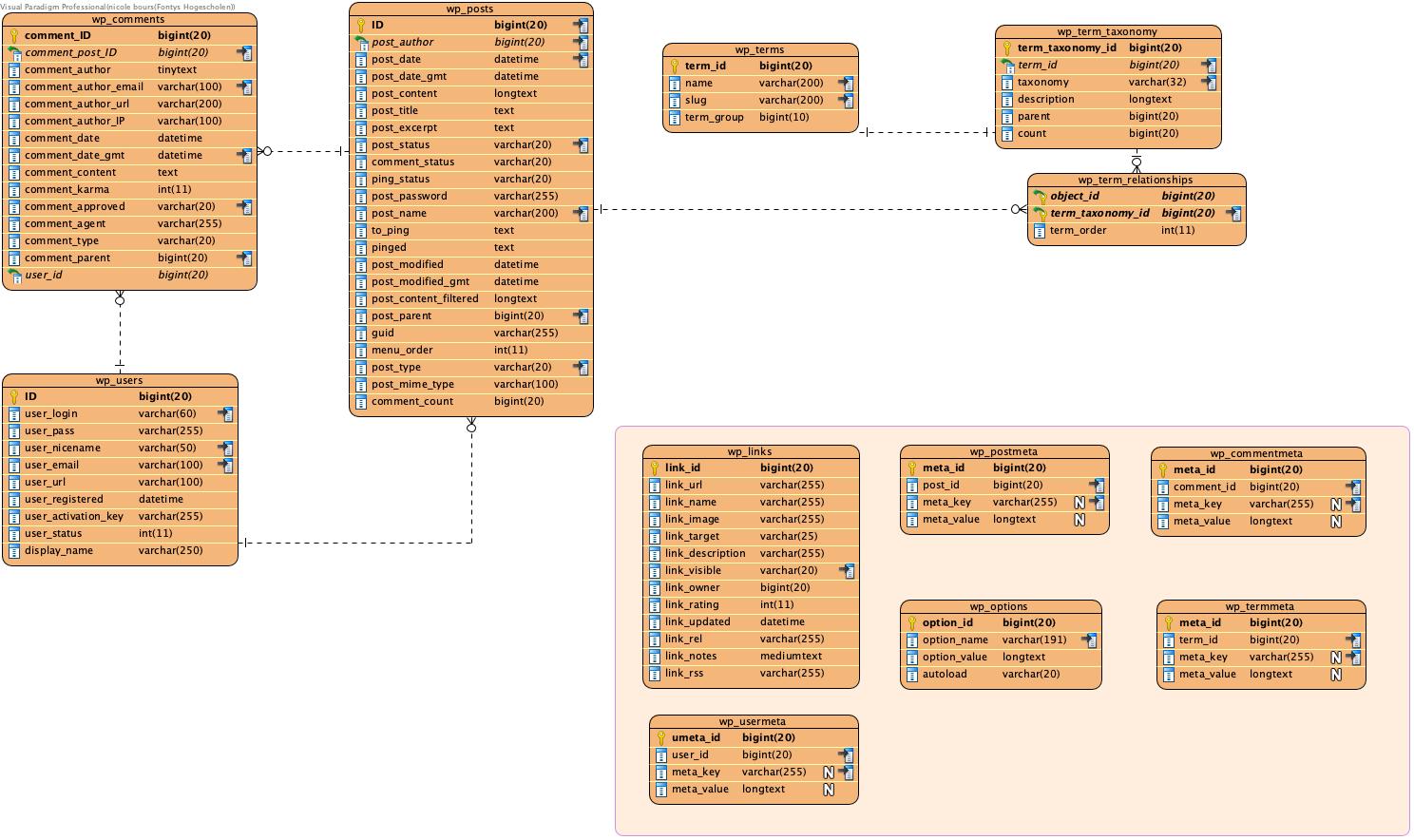
|  |  |
| --- | --- |
| **FR-07** De gebruiker kan zoeken naar games. | |
| **Requirement** | **Omschrijving** |
| *B-07.1* | De gebruiker kan naar 1 game tegelijk zoeken. |

# 

# 

# Huidige Database:

Dit is de database behoorde tot de wordpress site. Van hieruit kunnen de gegevens worden opgehaald.  
Posts: Deze staan in de wp\_posts tabel de bijbehorende comments zijn via wp\_comments op te halen in wp\_terms staan de tags en categorieën.



Queries:   
SELECT ID, post\_title, post\_content FROM wp\_posts

Haalt id, title en content op van een post.

SELECT ID, post\_title, post\_content FROM wp\_posts WHERE ID=?id

Haalt id, title en content van een enkele post op, op basis van id nummer

SELECT wp\_posts.ID , wp\_posts.post\_title , wp\_posts.post\_content From wp\_posts Inner Join wp\_term\_relationships on wp\_posts.ID = wp\_term\_relationships.object\_id inner JOIN wp\_term\_taxonomy on wp\_term\_relationships.term\_taxonomy\_id = wp\_term\_taxonomy.term\_taxonomy\_id INNER JOIN wp\_terms on wp\_terms.term\_id = wp\_term\_taxonomy.term\_id where wp\_terms.name like ?search

Haalt id, title en content op van posts op basis van category/tag name.  
  
Select wp\_terms.name From wp\_terms inner join wp\_term\_taxonomy on wp\_terms.term\_id = wp\_term\_taxonomy.term\_id where wp\_term\_taxonomy.taxonomy = 'tag'

Haalt alle tagnamen op.

Select wp\_terms.name From wp\_terms inner join wp\_term\_taxonomy on wp\_terms.term\_id = wp\_term\_taxonomy.term\_id where wp\_term\_taxonomy.taxonomy = 'category

Haalt alle categorienamen op.

"Select wp\_terms.name From wp\_terms inner join wp\_term\_taxonomy on wp\_terms.term\_id = wp\_term\_taxonomy.term\_id inner join wp\_term\_relationships on wp\_term\_relationships.term\_taxonomy\_id = wp\_term\_taxonomy.term\_taxonomy\_id where wp\_term\_taxonomy.taxonomy = 'category' AND wp\_term\_relationships.object\_id = ?id"

Haalt categorienaam op, op basis van categorie id

Select wp\_terms.name From wp\_terms inner join wp\_term\_taxonomy on wp\_terms.term\_id = wp\_term\_taxonomy.term\_id inner join wp\_term\_relationships on wp\_term\_relationships.term\_taxonomy\_id = wp\_term\_taxonomy.term\_taxonomy\_id where wp\_term\_taxonomy.taxonomy = 'post\_tag' AND wp\_term\_relationships.object\_id = ?id"

Haalt tagnaam op, op basis van tag id

# Regex:

Door Middel van regular expressions worden de gegevens die in de post\_content staat gefilterd en gereed gemaakt om aan de game classes toegevoegd worden.

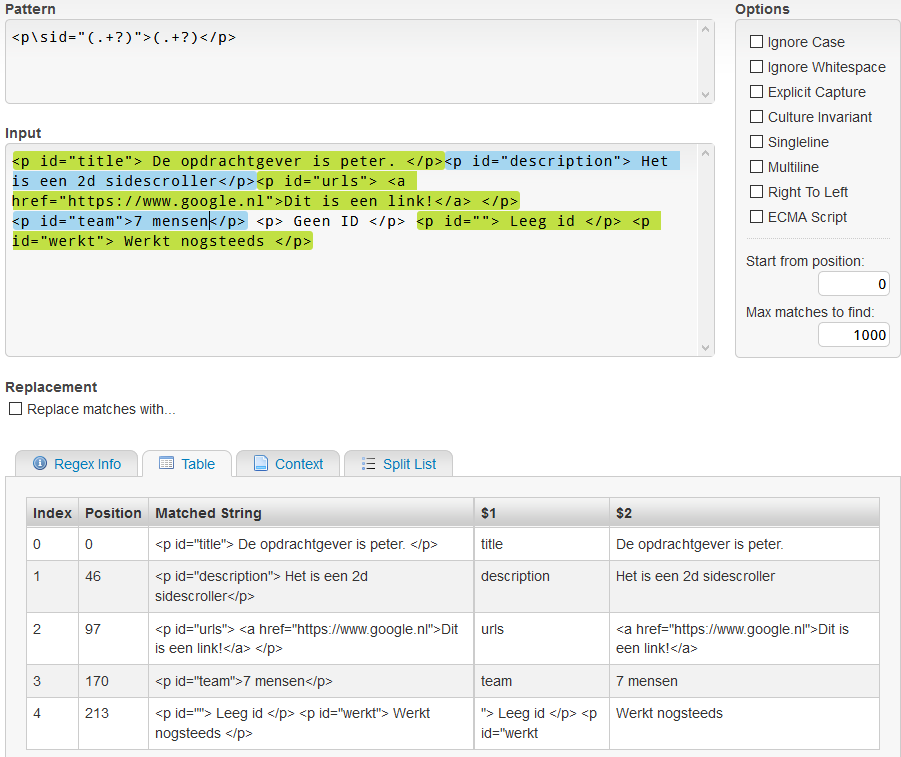
Omdat Wordpress letterlijke html code van een post opslaat in de database kunnen we met een vastgesteld html tag format een patroon opstellen die gebruikt kan worden om specifieke data te extraheren. In dit geval hebben we de paragraph tag in combinatie met een id gebruikt.

De regular expression gebruikt in de code is: <p\sid="”(.+?)"”>(.+?)</p>

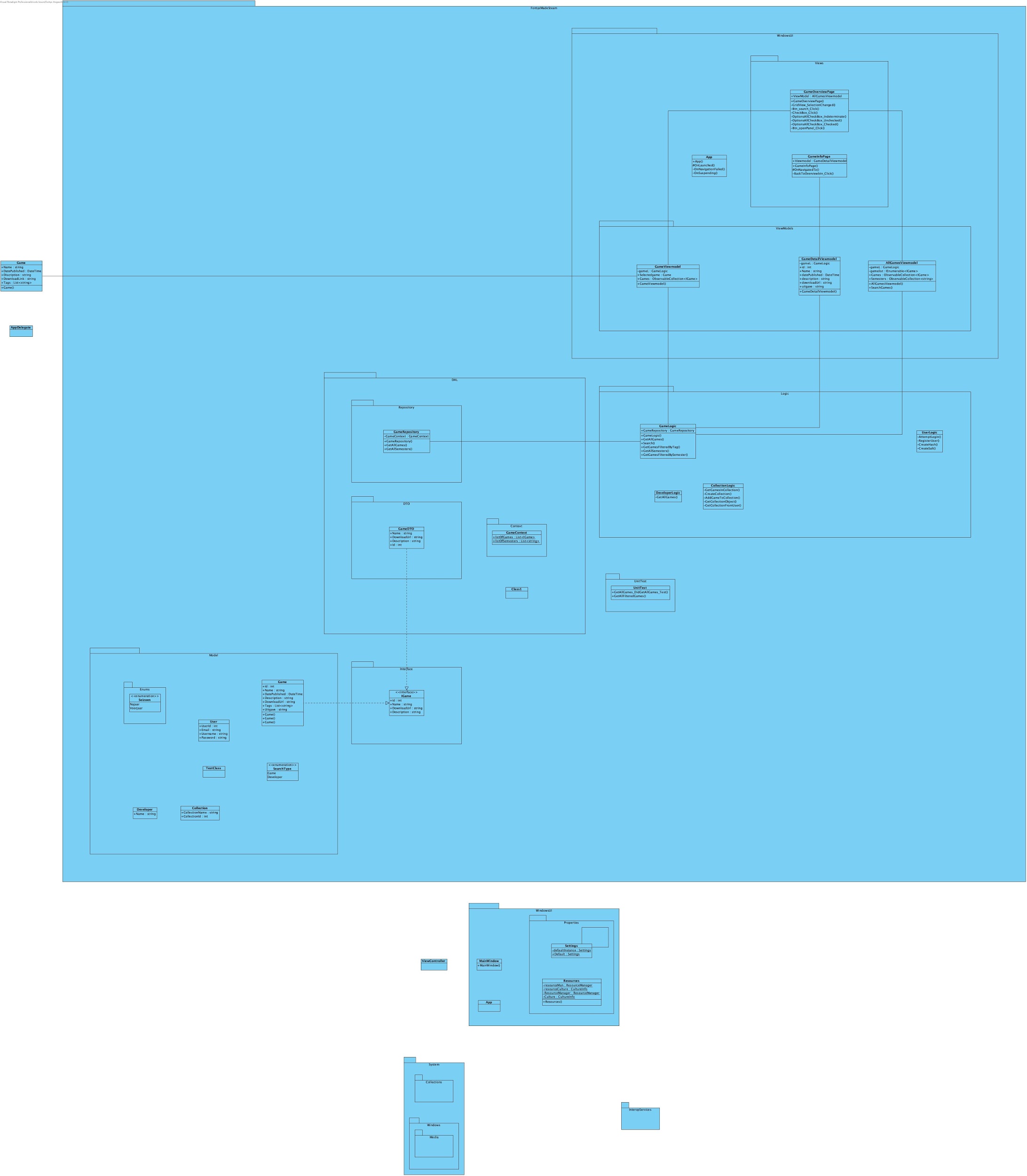
**<p\sid="”(.+?)"”>** zoekt naar het openen van de paragraph tag met id. Hiermee herkent het programma over wat voor content het gaat.  
 **(.+?)** creëert een subgroep die minimaal één karakter verwacht, dit mag elk karakter zijn. Een leeg of geen ID veld zal dus niet meegenomen worden door de regex.

**(.+?)</p>** Zoekt naar alles binnen de paragraph tags. Dit gebruikt de code als daadwerkelijke invulling van de Game pages. Hier wordt de tweede subgroep gemaakt. Vervolgens zoekt de regex de sluiting van de paragraph tag en kunnen de gefilterde groepen opgehaald worden.

Bij aanpassing van de html tag kan dit gemakkelijk veranderd worden. Bijvoorbeeld bij filteren van span tags ipv paragraphs hoeft enkel “<p” en “</p>” veranderd te worden naar “<span” en “</span>”.

De expressie is gemaakt en getest met behulp van de volgende website:  
<http://regexstorm.net/tester>  
Voor meer informatie:  
<https://www.youtube.com/watch?v=sa-TUpSx1JA&t=1826s>  
  
*Hieronder een test van de gebruikte expression:  
****Let op!*** *In de code worden er dubbele aanhalingstekens gebruikt ivm. Syntax.*

# Huidige Klassenstructuur:



# Huidige Solution:

De huidige Solution bestaat uit acht projecten. We gebruiken Class Libraries zodat beide UI projecten dezelfde Logica en Data delen.

|  |  |  |
| --- | --- | --- |
| Naam |  |  |
| FontysMadeSteam.DAL | .Net Standard 2.0 | Class Library |
| FontysMadeSteam.Interface | .Net Standard 2.0 | Class Library |
| FontysMadeSteam.Logic | .Net Standard 2.0 | Class Library |
| FontysMadeSteam.Models | .Net Standard 2.0 | Class Library |
| FontysMadeSteam.WindowsUI | Minimum version: Fall Creators Update | UWP Application |
| FontysMadeSteam.MacUI | Mojave | Xamarin Mac |
| FontysMadeSteam.UnitTests | .Net Core 2.1 | Console Application |
| FontysMadeSteam.DALUnitTest | .Net Core 2.1 | Console Application |

## 

## Communicatie tussen de Lagen

Onderstaande afbeelding laat zien hoe wij de communicatie tussen de lagen in gedachten hadden

