

Raport Project Databases April

Mathias Beke Bruno Van de Velde Elias Van Langenhove
Alexander Vanhulle Timo Truyts

23 april 2014

Inhoudsopgave

1	Status	2
1.1	Taakverdeling	2
2	Design	3
2.1	UML diagram	3
2.2	API	6
2.3	Parser (GO)	7
3	Database	8
3.1	Schema (ERM-diagram)	8
4	User Interface	8
4.1	Grafieken	8
4.2	Gebruikersgroepen	9
5	User Interface	11
6	Extra Functionaliteit	11
6.1	Wiki	11
6.2	RSS nieuwsfeed	12
6.3	OpenID	12
7	Planning	12
8	Appendix	13
8.1	Queries	13

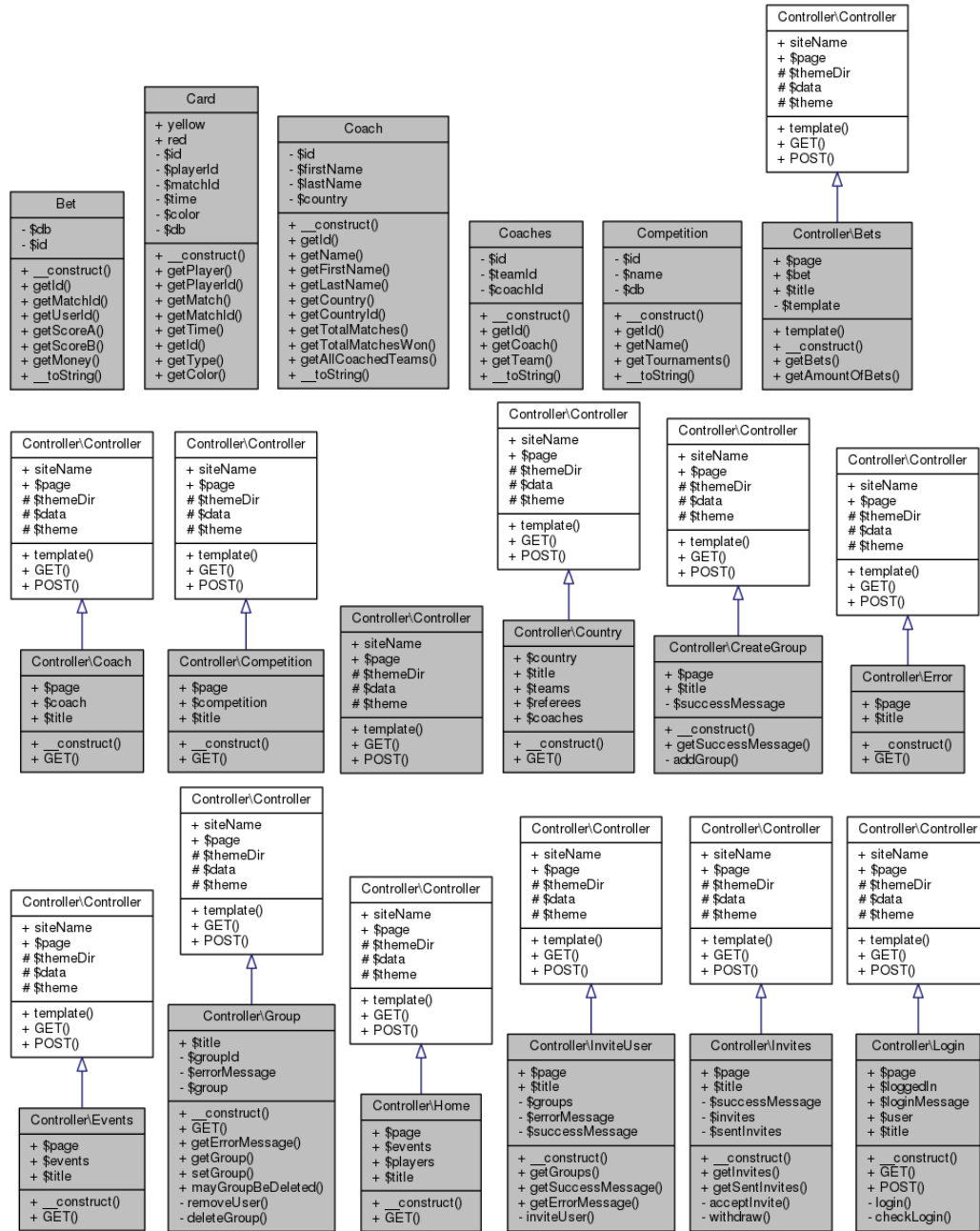
1 Status

1.1 Taakverdeling

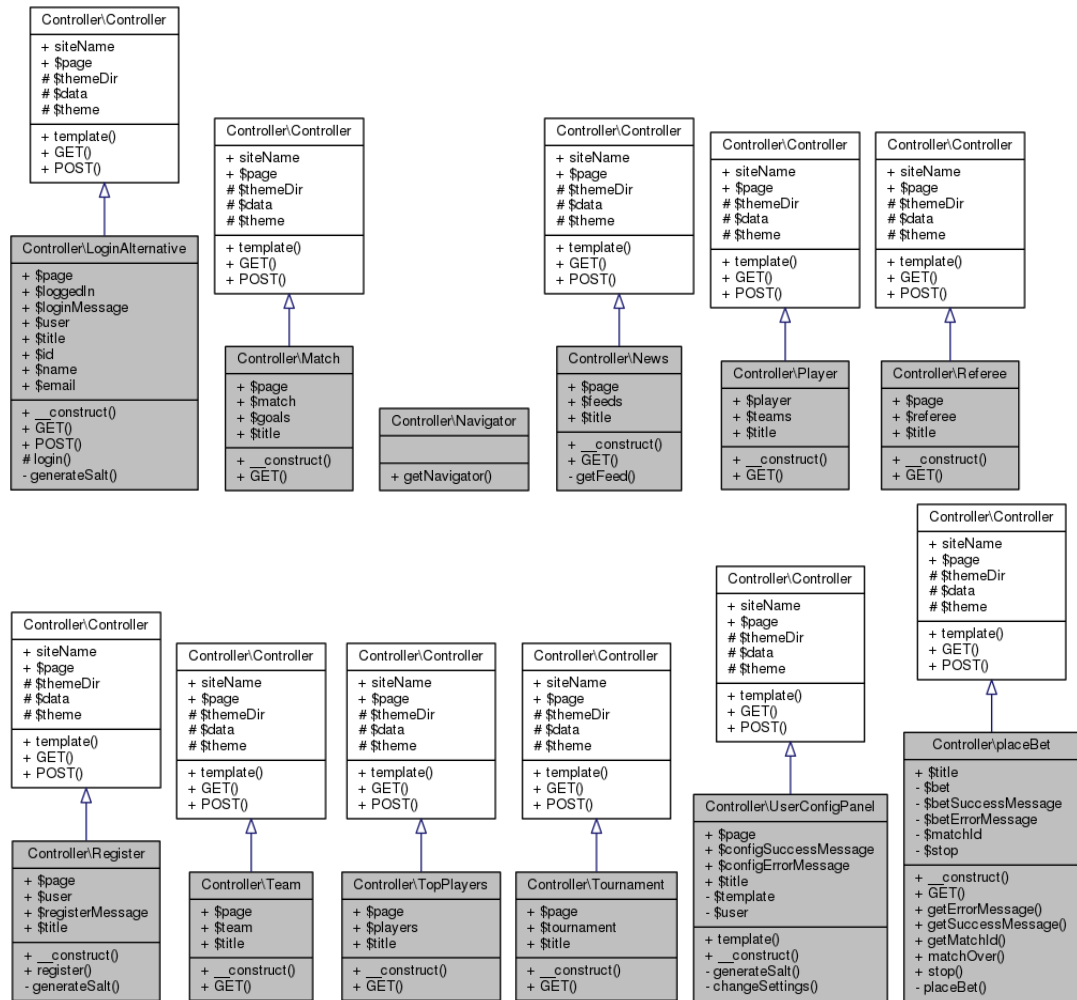
1. *Google/Facebook/OpenID login systeem* Bruno
2. *Wikipedia info toevoegen* Elias
3. *Foto's / Video's van spelers, teams, ...* Elias
4. *User groups* Alexander
5. *Scores van bets, verwerken van weddenschappen* Alexander
6. *Parser* Mathias (en Timo)
7. *Range-based graphs* Timo
8. *Enkele functies toevoegen aan groepen: Invites intrekken, groepen verwijderen, tonen van de bets van alle leden* Alexander
9. *Wikipedia afwerken* Elias
10. *Wikipedia toevoegen aan het geheel* Mathias
11. *Grafieken afwerken* Timo
12. *Parser afwerken* Timo en Mathias
13. *Chatbox* Mathias en Alexander
14. *Login vervolledigen* Bruno

2 Design

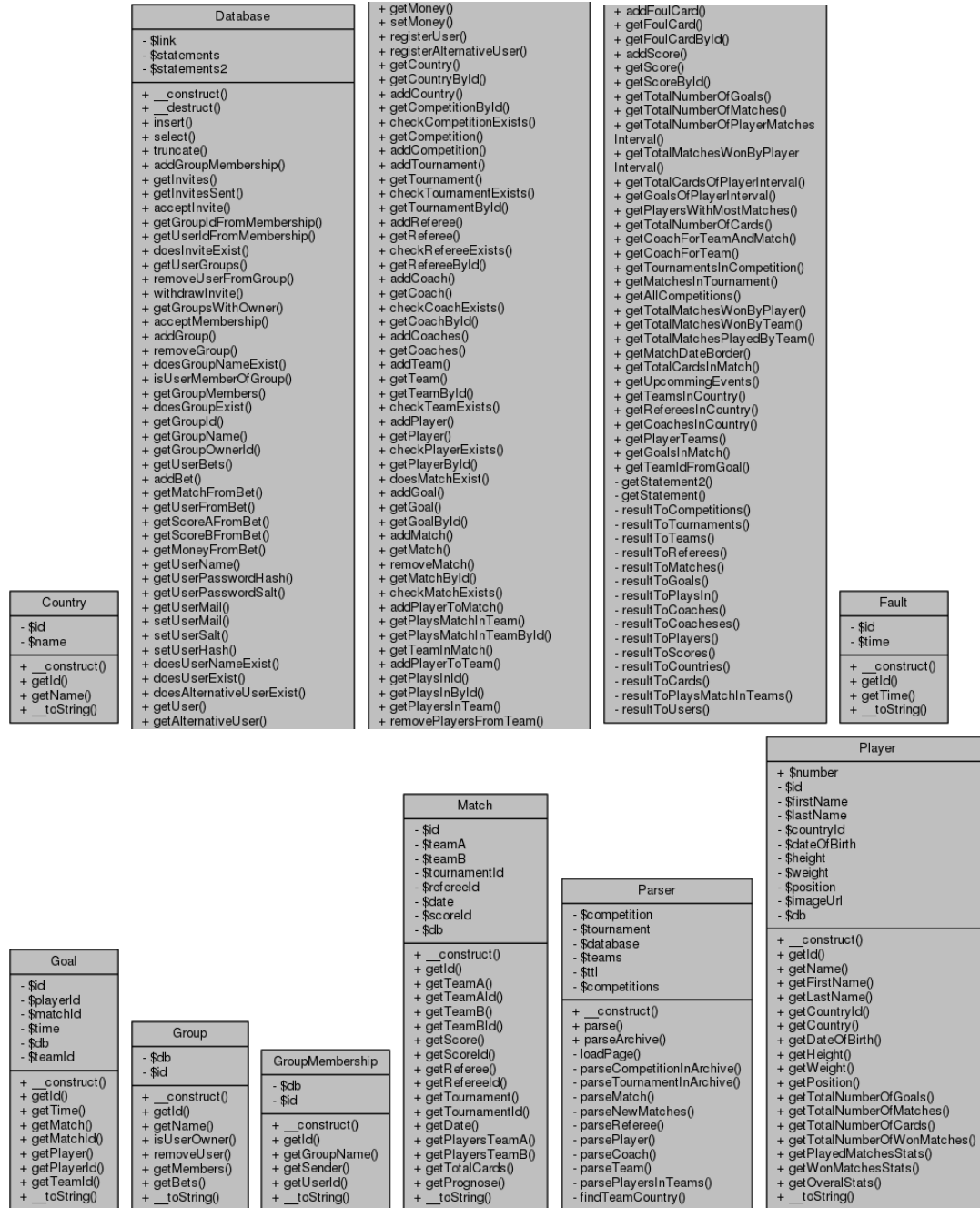
2.1 UML diagram

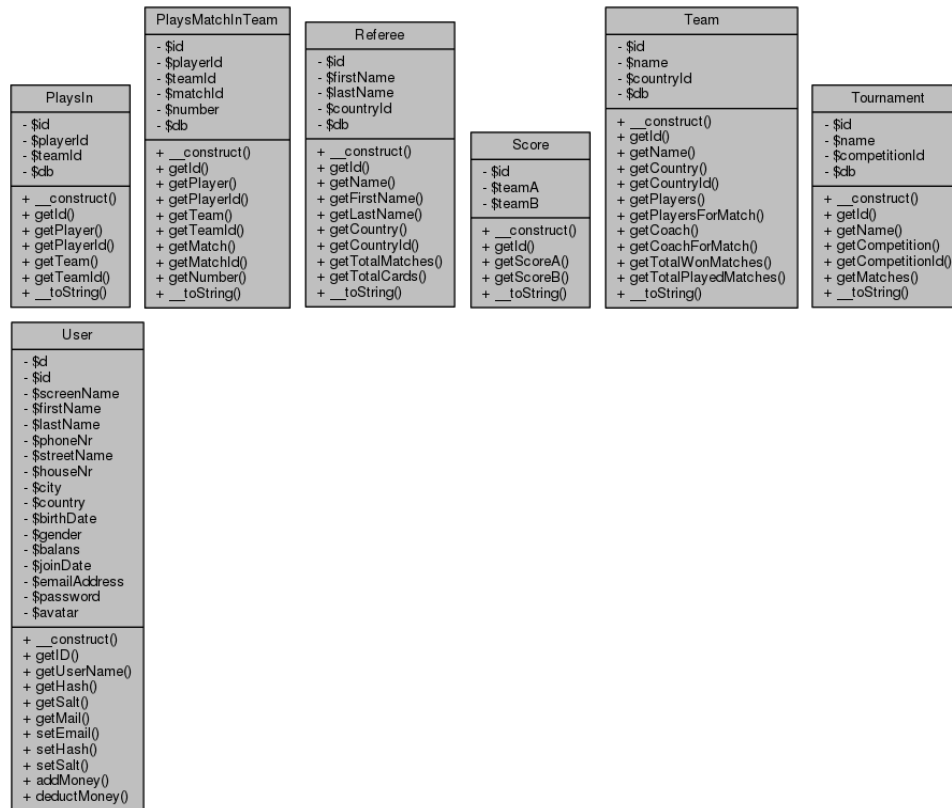


Report Project Databases: April



Report Project Databases: April





2.2 API

Om het mogelijk te maken om via javascript interactief informatie van de server op te halen zijn we begonnen met een api te schrijven.

De api zal antwoorden met json en kan dan eenvoudig in javascript worden gebruikt.

Hier enkele voorbeelden van hoe de api werkt:

```

// Returned informatie over alle spelers
/api/player

// Returned informatie over speler met id 5
/api/player?id=5
  
```

```
// Returned informatie over alle spelers met  
    voornaam Ronaldo  
/api/player?firstname =ronaldo  
  
// Returned informatie over alle spelers gesorteerd  
    op het aantal gespeelde matches  
/api/player?sort=match  
...
```

2.3 Parser (GO)

Bij de aanvang van het project zijn we meteen begonnen met onze parser. We schreven die toen in *PHP* en maakten gebruik van de *PHP Simple HTML DOM Parser*¹. Dit framework bleek zeer traag te zijn, en had geheugenlekken. Ook is *PHP* niet de snelste taal om zulke dingen te doen.

Daarom hebben we besloten op de parser opnieuw te beginnen. Ditmaal hebben we deze in *GO* geschreven. *GO* is een programmeertaal die uitermate geschikt is om concurrent te werken. Recent bedacht, maakt de taal concurrent programmeren veel gemakkelijker. Hierdoor wordt *GO* vaak op servers gebruikt.

De nieuwe versie van de parser is dus in *GO* geschreven en draait vele malen sneller doordat de competities en tournaments concurrent kunnen geparsed worden. De nieuwe parser zal dit archief dan exporteren in een *JSON* bestand, welk we in *PHP* importeren en toevoegen aan de database. (Door met een import/export file te werken, moesten we de database toegang niet herschrijven, en konden we de *PHP* herbruiken.)

Waar *PHP* er uren over doet, handelt *GO* de klus af binnen het half uur af. Een geslaagde verbetering dus.

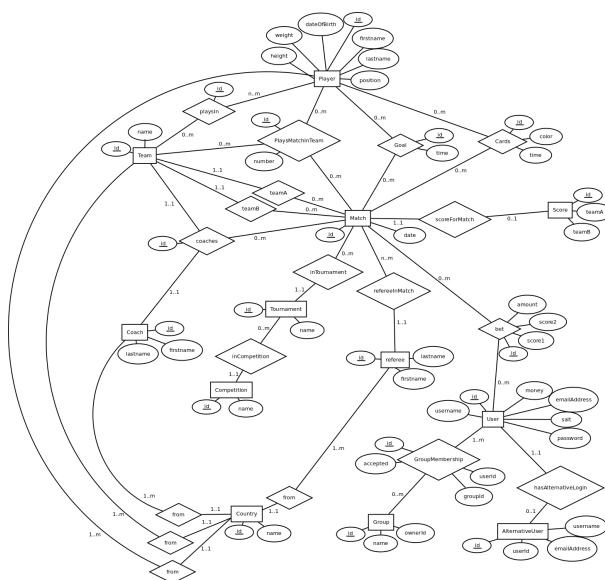
¹<http://simplehtmldom.sourceforge.net>

3 Database

3.1 Schema (ERM-diagram)

De ERM uit onze vorige versie was reeds redelijk compleet waardoor we hier slechts enkele kleine zaken voor nieuwe functionaliteiten dienden toe te voegen.

Hier ging het om de gebruiker-gerelateerde features: het groepen-systeem waardoor een gebruiker lid kan zijn van 0 of meer groepen en het alternatief login systeem waardoor o.a. Google en Facebook accounts nu tevens gebruikt kunnen worden voor het inloggen. Zoals vorige keer voegen we de ERM eveneens toe als bijlage zodat deze apart geopend kan worden (zie bijlagen/ERM2.png).



Figuur 1: ERM schema

4 User Interface

4.1 Grafieken

De framework die we gebruikten bleek niet geavanceerd genoeg om bijvoorbeeld te zoomen, ranges te selecteren, etc.

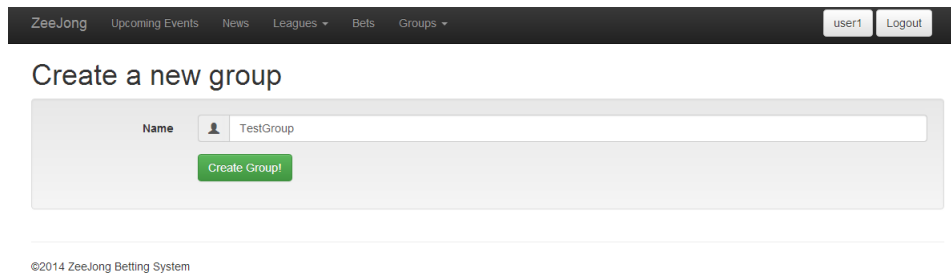
We hebben al veel frameworks geprobeerd maar hebben er nog geen gevonden die volledig voldoet aan onze behoeften.

Het huidige framework is *jqPlot*² waarmee we kunnen inzoomen en selecteren welke informatie er getoond moet worden.

4.2 Gebruikersgroepen

Een heel gebruikersgroepen systeem werd toegevoegd. Dit systeem stelt de gebruiker in staat zelf groepen aan te maken, andere gebruikers uit te nodigen enzovoort. Hieronder een overzicht:

Gebruikers kunnen zelf groepen aanmaken. Hierdoor worden zij de administrator van deze groep en hebben ze volledige rechten.



The screenshot shows a web interface for 'ZeeJong'. The top navigation bar includes links for 'Upcoming Events', 'News', 'Leagues', 'Bets', and 'Groups', along with a user profile 'user1' and a 'Logout' button. The main heading is 'Create a new group'. Below this is a form with a 'Name' label and a text input field containing 'TestGroup'. A green 'Create Group!' button is positioned below the input field. At the bottom of the page, a copyright notice reads '©2014 ZeeJong Betting System'.

Figuur 2: Create Group

Nadat de groep aangemaakt is, kan de beheerder andere gebruikers uitnodigen. Hiervoor is een formulier met een drop-down menu voorzien om tussen de verschillende groepen te kiezen waarvan de gebruiker administrator is.

Elke gebruiker heeft een venster waarop de uitnodigingen zichtbaar zijn die deze gebruiker ontvangen heeft en die de gebruiker verstuurd heeft. Een uitnodiging kan geaccepteerd/ingetrokken worden.

Wanneer een gebruiker een uitnodiging aanvaard heeft, wordt de groep toegevoegd aan het drop-down menu in de header waardoor de gebruiker naar de groep kan navigeren.

²<http://www.jqplot.com>

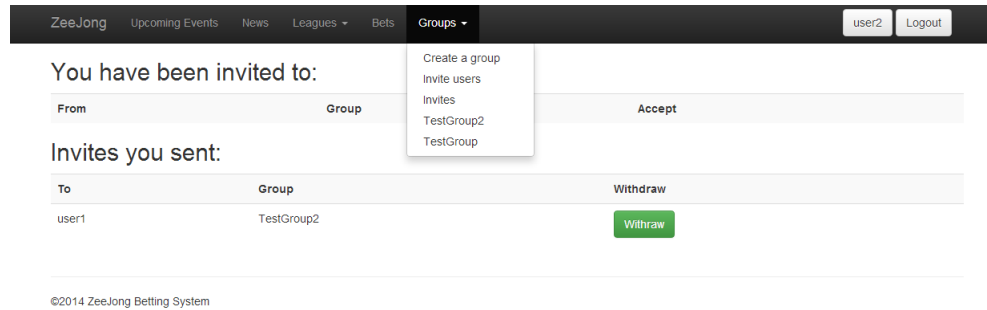
The screenshot shows a web interface for inviting a user to a group. At the top is a dark navigation bar with links: ZeeJong, Upcoming Events, News, Leagues, Bets, and Groups. On the right of the bar are buttons for 'user1' and 'Logout'. The main heading is 'Invite a user to one of your groups'. Below this is a form with two fields: 'User to invite' with a dropdown menu showing 'user2', and 'Group' with a dropdown menu showing 'TestGroup'. A green 'Invite!' button is positioned below the 'Group' dropdown. At the bottom of the page, there is a copyright notice: '©2014 ZeeJong Betting System'.

Figuur 3: Invite User

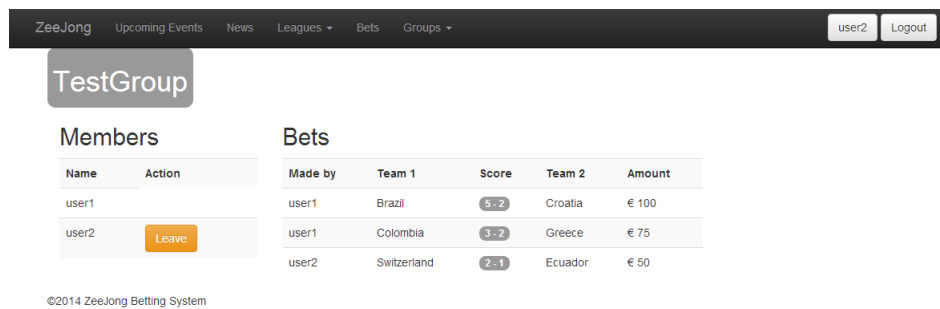
The screenshot shows the 'Invites' page. At the top is the same dark navigation bar as in Figure 3, but the 'user1' button is replaced by 'user2'. The main heading is 'You have been invited to:'. Below this is a table with three columns: 'From', 'Group', and 'Accept'. The table contains one row: 'user1' from 'TestGroup' with a green 'Accept' button. Below the table is the heading 'Invites you sent:'. Below this is another table with three columns: 'To', 'Group', and 'Withdraw'. The table contains one row: 'user1' to 'TestGroup2' with a green 'Withdraw' button. At the bottom of the page, there is a copyright notice: '©2014 ZeeJong Betting System'.

Figuur 4: Invites

In het groepsvenster kan de gebruiker bets van andere spelers van de groep zien. Een gebruiker kan ook uit de groep vertrekken. Indien de gebruiker tevens de administrator is, dan kan hij andere gebruikers uit de groep verwijderen en eventueel de groep zelf opdoeken.



Figuur 5: Invite Accepted



Figuur 6: Group

5 User Interface

6 Extra Functionaliteit

6.1 Wiki

Nu wordt op de speler pagina ook de introductie van de speler op wikipedia getoond. Deze pagina wordt gevonden door eerst te zoeken naar een sectie met als naam de naam van de speler. Indien deze niet gevonden kan worden wordt er een *Wikipedia* search gedaan met als termen de volledige naam van de speler en wordt zo de speler pagina gevonden. Voor het laden van

deze tekst wordt *wikidrain*³ gebruikt, mits enige aanpassingen die we zelf toegepast hebben.

6.2 RSS nieuwsfeed

De RSS nieuwspagina werd uitgebreid met extra feeds. De gebruiker krijgt bovenaan een menuutje, waarin hij een feed kan uitkiezen om het nieuws te lezen. De urls naar de feeds worden bijgehouden in een JSON file, zodat het toevoegen van extra feeds amper werk is.

Intern werkt de RSS nog steeds met *SimplePie*⁴ rss parser.

6.3 OpenID

Naast de normale login is er ook een optie om in te loggen door gebruik te maken van een externe account. Wanneer je naar de alternatieve login pagina gaat, krijg je een keuze uit verschillende services (zoals *Facebook* en *Google*) waarmee je kunt inloggen. Buiten Facebook werken deze allen met *OpenID* en er is uiteindelijk voor de *LightOpenID*⁵ library gekozen omdat deze makkelijk te gebruiken was. *Facebook* is achteraf nog apart toegevoegd aangezien die helemaal geen *OpenID* provider is. Om in te kunnen loggen via *Facebook* moest eerst een app worden aangemaakt op hun site om alles mogelijk te maken. Buiten het feit dat er extra code nodig is die het verschil tussen een *OpenID* en een *Facebook* login checked, is het grote nadeel dat de *Facebook* login enkel op de server kan werken en niet ondertussen op de localhost getest kan worden. Maar dit heeft geen gevolgen voor de gebruikers van de site.

7 Planning

De verdere planning focust zich vooral op kleine verbeteringen. We hopen dat we geen structurele veranderingen meer moeten doorvoeren.

³<https://github.com/abreksa4/wikidrain>

⁴<http://simplepie.org>

⁵<https://code.google.com/p/lightopenid/>

8 Appendix

8.1 Queries

Accepteren van invite Deze query past de groepslidmaatschappen van de user aan door een nieuwe groep toe te voegen.

```
UPDATE GroupMembership  
SET accepted = ?  
[WHERE id = ?];
```

User verwijderen uit groep Deze query verwijdert een gegeven user uit een gegeven groep.

```
DELETE  
FROM 'GroupMembership'  
[WHERE id = ? AND  
groupId = ?];
```

groep verwijderen Deze query verwijdert een gegeven groep.

```
DELETE  
FROM 'Group'  
[WHERE id = ?];
```

Invite intrekken Deze query trekt een gegeven verzonden invite weer in.

```
DELETE  
FROM 'GroupMembership'  
[WHERE id = ? AND  
accepted = False];
```

Invite accepteren Deze query accepteert een invite van een gegeven user voor een gegeven groep.

```
UPDATE 'GroupMembership'
SET accepted = True
[WHERE id = ? AND
groupId = ?];
```

Uitnodigen verzonden door gebruiker Deze query vraagt de uitnodigingen op die verzonden zijn door een gegeven user.

```
SELECT GroupMembership.id
FROM GroupMembership
[INNER JOIN 'Group'
ON GroupMembership.groupId=Group.id]
[WHERE Group.ownerId=? AND
GroupMembership.accepted=0]
```

Gewonnen wedstrijden van speler binnen interval Deze query vraagt de hoeveelheid matches waarbij de gegeven speler voor het winnende team speelde binnen een gegeven tijdspanne op.

```
SELECT COUNT(*)
FROM 'PlaysMatchInTeam'
[JOIN 'Match' ON 'Match'.id = matchId]
[JOIN 'Score' ON 'Score'.id = scoreId]
[WHERE playerId = ? AND
'Match'.date > ? AND
'Match'.date < ? AND
((teamId = 'Match'.teamA AND 'Score'.teamA > '
Score'.teamB) OR
(teamId = 'Match'.teamB AND 'Score'.teamB > '
Score'.teamA))];
```