Software Project Engineering

Phase 2 Report

Project leader : Hélène Verhaeghe
GROUP E
AURIAN DE POTTER(GROUP LEADER),
EDDY NDIZERA,
IVAN AHAD,
ARNAUD DETHISE,
LUDOVIC FASTRÉ,
ANTHONY DECHAMPS,
GEOFFROY HUSSON,
JONATHAN LEGAT

INGI2255

1   Introduction

We will start the *Phase 2* report by explaining how the implementation of the requirements planned for this phase is going. We will also describe a new ORM diagram of our database, which is a corrected version of the previous one, as well as the architecture of our software and how it has evolved since the last sprint.

2   Requirements developed in this phase, software description

The requirements below are the ones we planned to develop for the second phase. This time, every requirement has been implemented.

2.1   For this phase

During the previous phase, we decided not to implement one of the requirements (*After a pool has been generated, a staff member can manually reorganize it*). As we had a very busy phase, we came to the conclusion that it was not necessary to implement it because the benefits were not worth it compared to the time it would take us to do it. In this phase however, we made sure to implement it and all the requirements we had intended to develop. Here is the list of them :

— The user can register on a tournament alone and be matched with another player (Arnaud)
— An admin can start or close a tournament (Arnaud)
— The user can leave a comment when registering (Arnaud)
— A staff member can register a personal court as being available for the tournament (Already done in first phase)
— A staff member can access the courts list (Arnaud and Geoffroy)
— Users who registered their own court can see information about the players who will play on those (Ludovic)
— A staff can manually assign a court for each match, or modify it in case of raining (Eddy)
— A staff member can enter or edit match results data (Eddy and Arnaud)

It is important to note that the requirements haven't been implemented in this particular order.

Many other features have also been implemented of course, to improve quality of the current website, such as the creation of a sponsor page, a contact page, a feature that allows a staff member to see and edit all the users information, etc. Moreover, we made sure that the website is *mobile-friendly*, and some other improvements have been brought to the design.

You can see in the list above who implemented which requirement(s) for this phase. The other features were implemented by Ludovic, Aurian, Jonathan, Anthony, and Ar-

naud. Ivan took care of the report and made sure it respected all the criterion listed in the Criterion file. The team in general corrected the ORM diagram and updated the UML one for this phase.

Last sprint, the form for player registration was not working. This issue has now been fixed. Note that the players must choose a tournament to register for, otherwise the registration would be invalid. When this is done, the player will be redirected to the home page. The same mechanism goes for the court registration. For future phases we planned on making it not possible for a player to register if there isn't any available tournament to register for. We also planned on adding a page showing that a registration has successfully been added.

2.2 The software

Here is the live version of our website : http ://sep2015e.herokuapp.com. To get admin access you should add "/admin" in the url. The username and the password are both **admin**. In the last phase our website only worked on our host. For this sprint, we made sure that it works even on the live version.

To get access to the data of our project, you can click on the link to our *Github* repository : https ://github.com/ivanahad/sep2015E

3 ORM diagram, corrected version

In this section, we will only explain the differences between the old version of the ORM and the new one, you can always check the report of Phase 1 to see any further explanation of the diagram.

You can see the diagram in figure 1.

In the previous ORM, we had some inconsistencies, especially with some relations that we put as being unique whereas no unicity was needed. For example, we put that an address can only be used by one user, meaning that a couple living in the same house cannot register to the same address, we also made a mistake in putting that a player can only be assigned to one team, while a player should actually be able to register to several, hence, participating in several tournaments throughout the event.

4 Architecture discussion & UML Diagram

You can see our UML diagram in Figure2. Since the last phase, we changed the **views** classes' structure and we modified the diagram so that it looks simpler and has less overhead.

Our software follows Django's architecture. We gathered all the *html and css* files for the whole website. Before doing so, we had all the HTML files in the repositories of every application of the software. Now, the HTML file of each page extends the file *site_ template* so that we avoid having redundancy. Moreover, this technique is more generic as well.

5 Communication with the client & Feedback

Days before the deadline, we contacted the clients to get some feedback on our website. We asked them to tell us what they felt needed to be changed in our software. However, they were actually unable to give us their opinion about it just yet. We will make sure to take into account the feedback received for the next sprint.

6 Conclusion

After phase 2, the website is starting to reach its final form, as more and more requirements and features start to be implemented into the website. We made sure to correct the mistakes in the ORM diagram and correct some small details that were not working.

Appendices
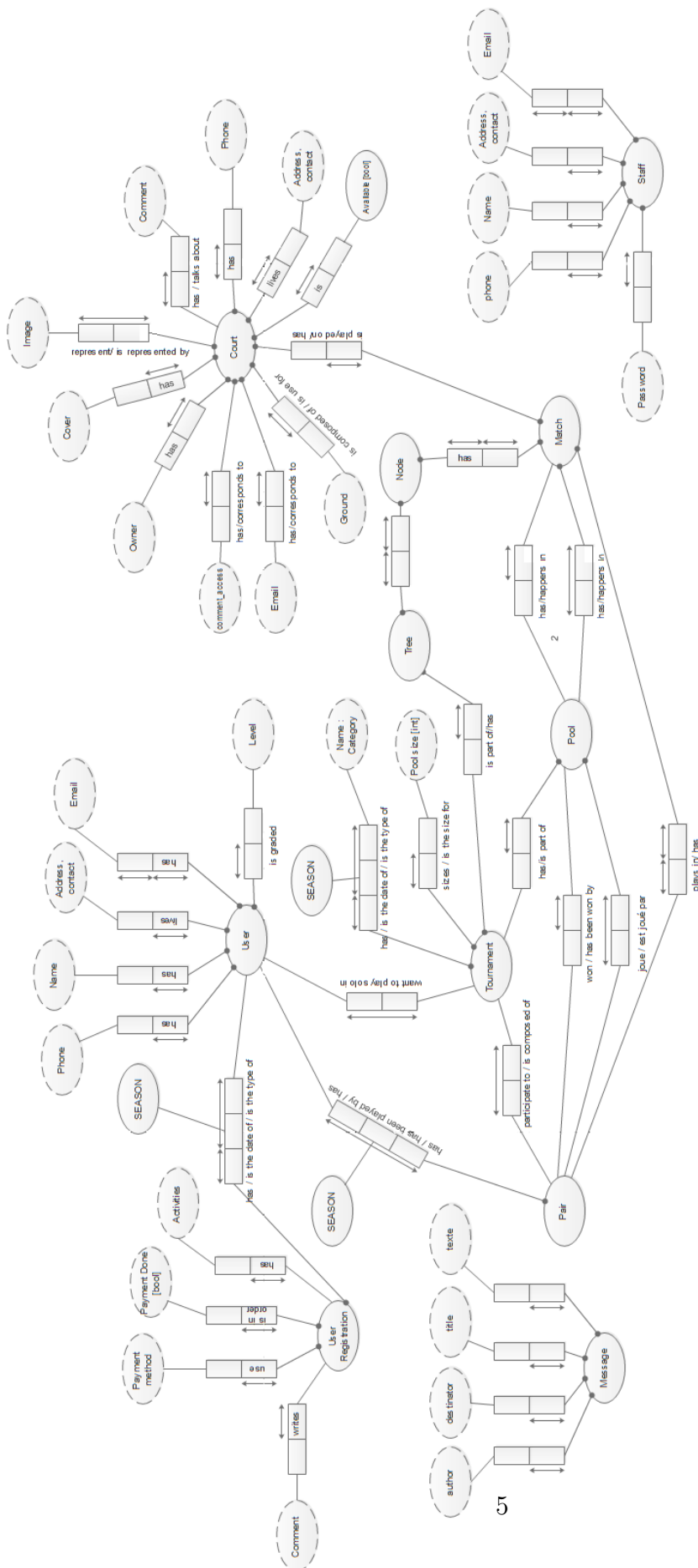
Screenshots of the website
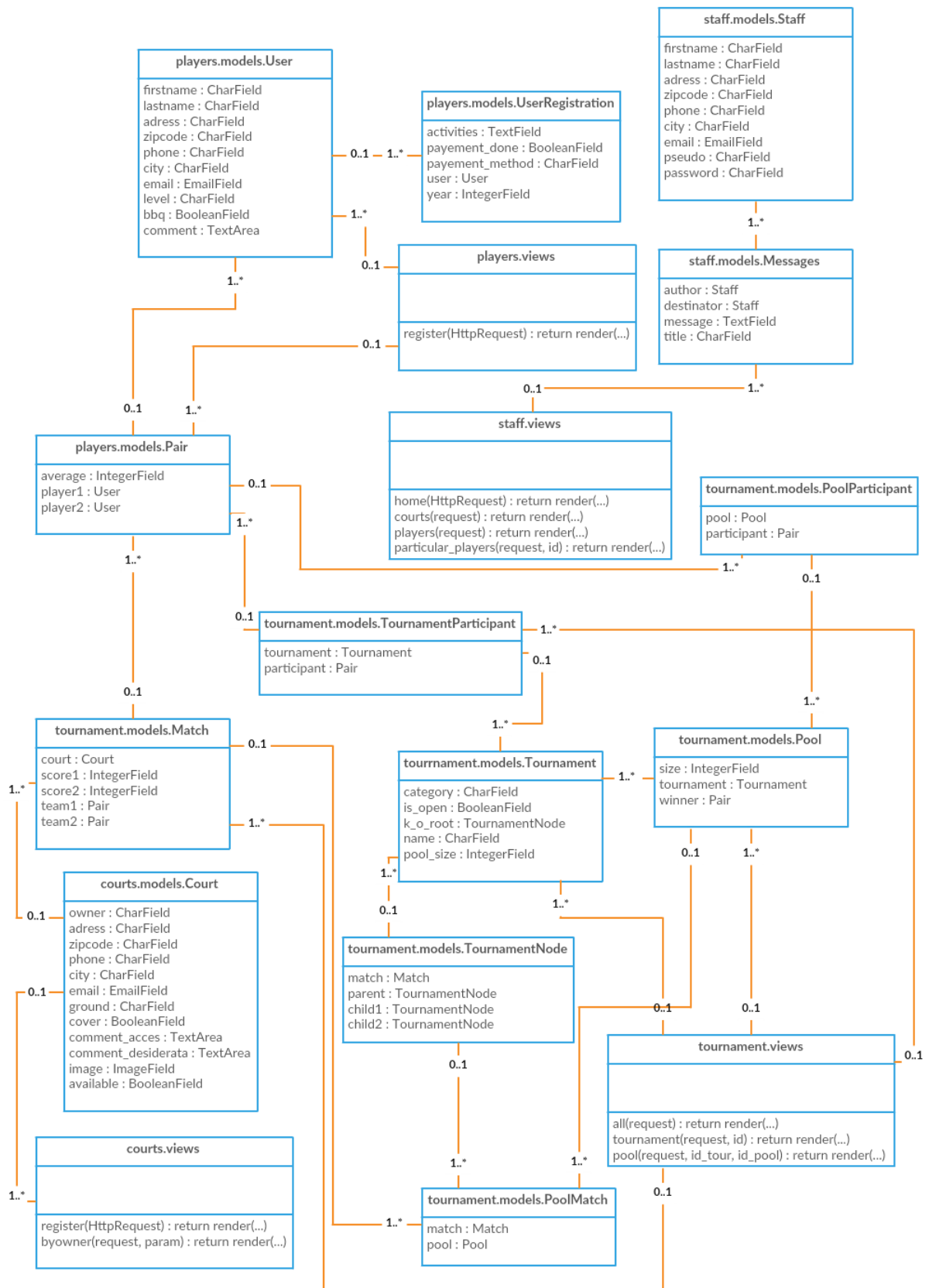
FIGURE 1 – Revamped ORM Diagram

Figure 2 – Revamped UML Diagram

players.models.User

firstname : CharField
lastname : CharField
adress : CharField
zipcode : CharField
phone : CharField
city : CharField
email : EmailField
level : CharField
bbq : BooleanField
comment : TextArea

players.models.UserRegistration

activities : TextField
payement_done : BooleanField
payement_method : CharField
user : User
year : IntegerField

staff.models.Staff

firstname : CharField
lastname : CharField
adress : CharField
zipcode : CharField
phone : CharField
city : CharField
email : EmailField
pseudo : CharField
password : CharField

players.views

register(HttpRequest) : return render(...)

staff.models.Messages

author : Staff
destinator : Staff
message : TextField
title : CharField

players.models.Pair

average : IntegerField
player1 : User
player2 : User

staff.views

home(HttpRequest) : return render(...)
courts(request) : return render(...)
players(request) : return render(...)
particular_players(request, id) : return render(...)

tournament.models.PoolParticipant

pool : Pool
participant : Pair

tournament.models.TournamentParticipant

tournament : Tournament
participant : Pair

tournament.models.Match

court : Court
score1 : IntegerField
score2 : IntegerField
team1 : Pair
team2 : Pair

tourrnament.models.Tournament

category : CharField
is_open : BooleanField
k_o_root : TournamentNode
name : CharField
pool_size : IntegerField

tournament.models.Pool

size : IntegerField
tournament : Tournament
winner : Pair

courts.models.Court

owner : CharField
adress : CharField
zipcode : CharField
phone : CharField
city : CharField
email : EmailField
ground : CharField
cover : BooleanField
comment_acces : TextArea
comment_desiderata : TextArea
image : ImageField
available : BooleanField

tournament.models.TournamentNode

match : Match
parent : TournamentNode
child1 : TournamentNode
child2 : TournamentNode

tournament.views

all(request) : return render(...)
tournament(request, id) : return render(...)
pool(request, id_tour, id_pool) : return render(...)

courts.views

register(HttpRequest) : return render(...)
byowner(request, param) : return render(...)

tournament.models.PoolMatch

match : Match
pool : Pool

0..1   1..*   0..1   1..*   1..*   0..1   1..*   0..1   1..*   0..1   0..1   1..*   1..*   0..1   1..*   0..1   1..*   0..1   1..*   0..1   0..1   1..*   0..1   1..*   1..*   0..1   0..1   1..*   0..1   1..*   0..1   1..*   0..1

FIGURE 3 – Home page

FIGURE 4 – Player(s) registration

FIGURE 5 – Owner court registration

FIGURE 6 – Admin Page

Site administration

Authentication and Authorization		
Groups	✚ Add	✎ Change
Users	✚ Add	✎ Change

Courts		
Courts	✚ Add	✎ Change

Players		
Pairs	✚ Add	✎ Change
User registrations	✚ Add	✎ Change
Users	✚ Add	✎ Change

Staff		
Staffs	✚ Add	✎ Change

Tournament		
Matchs	✚ Add	✎ Change
Pool matchs	✚ Add	✎ Change
Pool participants	✚ Add	✎ Change
Pools	✚ Add	✎ Change
Solo participants	✚ Add	✎ Change
Tournament nodes	✚ Add	✎ Change
Tournament participants	✚ Add	✎ Change
Tournaments	✚ Add	✎ Change

Recent Actions

My Actions
None available

Figure 7 – Staff Page