EE 461L Team Project Report: ChampionsDB

Ben Buhse, Pearse Flood, Conor Flood, Juan Paez

TEAM INFORMATION

Project Name: ChampionsDB
Canvas Group: Team A6
GitHub Repo Link: https://github.com/UT-SWLab/TeamA6.git

Team Member	UT EID	GitHub	Email
Juan Paez	jfp778	juanpaez22	juanfranciscopaez@gmail.com
Ben Buhse	bwb887	bwbuhse	bwbuhse@utexas.edu
Conor Flood	cf26784	cnflood	conorflood@utexas.edu
Pearse Flood	pkf256	pkflood	pearseflood@utexas.edu

MOTIVATION AND USERS

Soccer is the most popular sport in the world. Even so, its popularity continues to grow with momentum, particularly as it begins to seep into the cultures of the US and Canada, which have traditionally rejected the sport. A natural starting place to learn more about the soccer world is becoming invested in the UEFA Champions League (UCL), the world's most prestigious and competitive club soccer competition. Our web application, ChampionsDB, aims to provide a centralized resource for soccer novices and lifelong fans alike to learn about the UCL.

Right now, those looking to learn more about the UCL do not have a centralized resource to do so-- they may look at match scores on ESPN, view individual player information on Wikipedia, and learn about teams through their webpages. Through several REST API data sources, ChampionsDB puts all of this information in one place and presents it in a clean, integrated fashion. We provide matches, players, and teams as three models, each populated with many instances, in order to provide our users with a centralized location for all of this information.

The information we provide can be helpful to anyone interested in the subject, but is particularly aimed at three subgroups of users: soccer novices, sports gamblers, and data geeks. Soccer novices will be able to use this information to familiarize themselves with soccer at the most competitive level. Sports gamblers will be able to learn from trends and predictions in the previous season and apply them to upcoming matches. Lastly, data geeks will be able to visualize lots of information about players, teams, and matches in a single web platform.

REQUIREMENTS

Phase I User Stories

1. As a general user, I should be able to navigate smoothly between the index page, the model pages, and the instance pages so that I can easily digest the presented information and find what I am looking for.
    - Time estimate: 6 hours
    - Actual time: 5 hours
2. As a general visitor, I should be able to view an about page with information about the site including its purpose, intended users, an explanation of the result produced by the

data, the group name and its members, information about each member, links to each data sources, information about how each tool was used, and a link to the GitHub repo.
    - Time estimate: 2 hours
    - Actual time: 3 hours
3. As a general user, I want a coherent home page that clearly presents and summarizes the information available to me at ChampionsDB so I can easily understand my options.
    - Time estimate: 3 hours
    - Actual time: 3 hours
4. As a user wanting to learn more about the UCL, I want to be able to view a grid of all its teams and their respective shields to easily visualize the makeup of the league.
    - Time estimate: 2 hours
    - Actual time: 4 hours
5. As a user wanting to learn more about a specific team, I should be able to view more detailed information like their popular players and recent match results in an instance page for that team.
    - Time estimate: 2 hours
    - Actual time: 4 hours
6. As a soccer fan, I should be able to view a grid of match scores so I can see a variety of results in the same place.
    - Time estimate: 3 hours
    - Actual time: 2 hours
7. As a user curious about previous matches, I should be able to access pages specific to individual matches containing statistics and information about teams and players involved so I can learn more details about individual matches.
    - Time estimate: 2 hours
    - Actual time: 2 hours
8. As a soccer novice, I should be able to view the basic profile information (name, position, nationality) of popular Champions League players in a grid so I can learn more about the most important names in the soccer world.
    - Time estimate: 3 hours
    - Actual time: 3 hours
9. As a more involved soccer fan wanting to learn more about a specific player, I should be able to view a detailed player page containing their strengths and stats so that I can understand that player's style and trends better.
    - Time estimate: 3 hours
    - Actual time: 5 hours

Phase II User Stories (Drafts)

10. As a soccer fan, I want to be able to view the player roster of a given team by clicking on its link from the team listing page to easily know who plays on each team.
11. As a sports gambler, I want to view detailed statistics, such as player rankings and number of goals, for each of the top 100 players in the UCL to holistically assess the abilities of a given player.

DESIGN

TESTING

MODELS AND DATA SOURCES

SOFTWARE TOOLS AND FRAMEWORKS

REFLECTION

Phase 1 Reflection:

After assessing what went well for our team and taking note of the things we struggled with, there are many key takeaways that can help streamline and improve the development process for later phases of the project. Beginning with things our team did well, all members were incredibly receptive to each other's needs which prevented people from falling behind and ensured that everyone stayed on track. Updating and closing issues on Git as they came up also ensured that all members were on the same page in terms of the project's completion status. Additionally, our meetings were organized, focused, and productive, which greatly contributed to our team's ability to finish all parts of Phase 1 by the deadline. Our system for committing changes and updates also proved to be very effective. By issuing pull requests for any file changes and requiring at least one other team member to review them, we were able to avoid merge conflicts and ensure that our master branch was always working. To streamline project development, we also went ahead and implemented a database using mongoDB even though doing so was not required.

Although many things went well for our team, there is always room for improvement. For most tasks, our actual completion times differed, sometimes significantly, from our initial estimates. Now that we have become more in-tune with each other's workflows and productivity levels, we can better estimate completion times in later phases of the project. Additionally, we could have improved how tasks were delegated. Although our team dynamic was good and all members contributed, some team members ended up taking on more responsibility than others. Because of this, task delegation is something we can refine in later phases to make the spread of work more equitable across all team members. Another thing that would have been beneficial was to get more headway on the project during earlier stages of the phase. Majority of our work happened in the week and a half leading up until the phase deadline. Making more progress earlier on would have allowed us to further refine our website instead of saving such changes for later phases. Although not an issue to date, we also should have allocated more time for possible setbacks. If there had been a huge unforeseen bug in our code, we likely would have been pressed for time to address it. Finally, some team members ran into errors due to not routinely pulling the most recent changes before beginning work. Being more cognizant of this would have helped streamline the development process and avoid unnecessary problems. Despite these minor missteps, our team was largely successful in achieving our goals and collaborating to fulfill all the requirements of Phase 1.

To summarize:

Five things that went well:

1. Team members were very receptive to each other's needs
2. Organized, focused, and productive meetings/standups
3. Updated and closed issues on Git as they came up so that team members were on the same page
4. System in place for issuing and reviewing pull requests so that master branch was always working and bug-free
5. Implemented database using mongoDB to streamline project development even though doing so was not a requirement

Five things we struggled with:

1. Accurately estimating time to complete tasks
2. Room for improvement in how tasks were delegated
3. Would have been beneficial to get more headway during the early stages of the phase.
4. Should have allocated more time for unforeseen setbacks
5. Making sure to routinely pull changes before beginning work

REFERENCES

- Cloud Atlas for MongoDB: https://www.youtube.com/watch?v=rE_bJl2GAY8&ab_channel=TechWithTim
- W3Schools Bootstrap tutorial: https://www.w3schools.com/bootstrap/default.asp
- Flask MongoEngine: http://docs.mongoengine.org/projects/flask-mongoengine/en/latest/
- Stack Overflow: https://stackoverflow.com/
    - https://stackoverflow.com/questions/3206344/passing-html-to-template-using-flask-jinja2
    - https://stackoverflow.com/questions/8189702/mongodb-using-an-or-clause-in-mongoengine
    - https://stackoverflow.com/questions/19607851/use-bootstrap-grid-with-variable-number-of-cells
    - Many, many more