12/16/2022

Web Application Development – Coursework 2 Report

By Chloé Beetschen

Chloé Beetschen
201488921

Purpose:

My website has been designed and coded to be a fully functioning application for the Leeds University Union Women's Association Football Club 's 4th Team.

This includes allowing users to view the team information, any fixtures that are upcoming or past, and the team members in the team. Furthermore, a user will be able to login and signup to the website.

Link:

https://sc21cab.pythonanywhere.com/

Use of 3-Tier Architecture:

I have made use of three-tier architecture in my application. To begin with, the first tier, the presentational tier, includes the aspects of my web application that manage how to web application looks and any user input that needs taking in. Through the use of multiple templates for each page of the website and by using forms on certain pages to take an input from a user, this tier has been fulfilled. In addition, through implementing JavaScript and Css files that link to my templates, the front-end of the application is able to look aesthetically pleasing and have a user interface that is easy for the user to navigate and use.

Secondly, for tier-two, the business logic tier, I have included and made use of a views.py file. Within this file, there is all the functions that will connect the user-input taken in from the front-end (tier-one) layer and be able to either utilise them and act accordingly or transfer the information and data to the back-end layer of the application. Furthermore, this tier is also responsible for performing some low-importance, data validation from the data received from the forms on the website.

Finally, tier-three, the data-access tier, this is where I store my database and models. I have 3 models to incorporate members, fixtures, and fixture information. All tables are linked however, members to fixtures contains a many-to-many relationship however fixture to fixture information makes use of a one-to-one relationship. Any information inputted by the user into the

web application in tier one, will be passed down through the views in tier-two and then further inputted and stored into the database in this final tier.

Request Handling Features:

All my forms contain checks to make sure the form has been filled in and will provide popups to the user when this has not occurred. If a form is filled in correctly, the user's inputted data will be added into the corresponding model and/or directed to the relevant page.

Analysis:

Web Forms:

I have created 5 Web Forms for this application. All forms have a data validation check where they confirm that data has been added to each box/field of the form.

1) LoginForm. This is used to log a user, who is already signed up, into the application. It performs validation checks in the view.py file to ensure that the email entered is valid and that the password matches such member's email.
2) ResetPass. This is used to reset an existing user's password if they have forgotten it. It will prompt the user to enter their name and email, and a new password. It will check that the user exists, and then it will update the password to allow the user to then log in.
3) MemberForm. This form allows a visitor of the webpage to join the team and be able to access more sections of the page than previous was viewable.
    a. In future implementations of this application, I would include a check that the email is a University of Leeds address, as well as only enable someone to sign up if approved by the Captain/Coach
4) ContactUsForm. This form enables the user of the web application to contact the Coach/Captain with any queries they may have.
    a. Something I would change about this is that I would make it functional and have it send via email to the needed parties.
5) FixtureForm. This allows a fixture to be created and presented on the app to every member.

Database:

My database includes 3 models. These models work together to be able to provide a user with information on the members in the team, fixture information and the line-up for each fixture.

1) Member model. This model contains all the information about a member of the team, including a hashed version of their password for security.
2) Fixture model. This contains the main information about a fixture, such as the date, team and whether it is home or away.
3) FixtureInfo model. This model consists of the rest of a fixture's information, including its location and time.

Member and Fixture are linked with a many-to-many relationship, as many fixtures can be played by a member and many members will be in the line-up for one fixture.

Fixture and FixtureInfo are linked through a one-to-one relationship, as one fixture will have one corresponding fixture information section.

Authentication and sessions:

For authentication, I included a login section of my website once the user logs in/signs up they will have access to more of the application. I paired this with the use of sessions to keep the user logged in throughout their use of the application, until they log themselves out or they shut down the page.

Logging:

I have implemented logging of information for any function in my view.py file that will produce a result (eg. Logging in or creating a fixture). The logging information will be stored in a file called logfile.log so that a record is kept of what has happened and debugging can be done more easily if needed.

Deployment:

After following the steps in section 11 of our lecture notes, I deployed an online web application that can be found at the link above.

Styling & Features:

My style choices (colour, font, and text size) were consistent throughout the application to make my website look more professional and flow. Overall, I

believe my website is easy to read and all sections are visible. The pages are all responsive also and support browser resizing.

As an advanced feature, I chose to implemented a carousel of images on the homepage, to welcome in users and visitors of the webpage. To implement this, I used javascript to control which image would be shown when each button is clicked.

Accessibility:

To account for some user's having accessibility issues and to ensure no discrimination occurs against them, I have ensured that all text is always large enough to be read and visible by a user as well as using large icons for buttons so that it is clear where they are, and they can be clicked on easily. I have especially done this with the thought in mind that the web application could be opened on a touch screen device as well as a laptop/computer device.

In addition, to determine the colours I was going to use, I was initially planning on using shades of green (as these are the colours of the team), however after investigating and researching the types of colour blindness [1] I have decided to go with shades of blue. This is down to two reasons, to start, my research found that blue is a colour that the least amount of people with colour blindness suffer from and therefore, only using shades of blue makes a monochromatic and analogous web application, which will reduce the contrast and reduce the difficulty for some users to read it. Furthermore, my colour palette also includes grey and white. The reason for this being that grey and white compliment the colour blue as they do not give a lot of contrast and lighten up the pages more, thus making individual buttons and text stand out more to users with visual impairments.

Security Issues:

Most of the security risks of this application will occur through the login stage, these types of issues include: sql injection, sensitive data exposure and mitigation. To combat this, I have ensured that sensitive information, such as a password is hashed for security. In addition, when resetting your password, you have to enter it twice to confirm it. In addition to battle CSRF I have included a token into my config.py file.

References:

Constant use throughout my application of:

- https://www.w3schools.com/
- https://alt-6100e9398f586.blackboard.com/bbcswebdav/courses/202223_32870_COMP2011/site/

All pictures and media are from LUUWAFC society, as I am a committee member and captain I have access to these and copyright over them.

[1] National Eye Institute. 2019 [Online]. [Accessed 16 December 2022]. Available from: https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases/color-blindness/types-color-blindness