Coursework Report

Max Millard

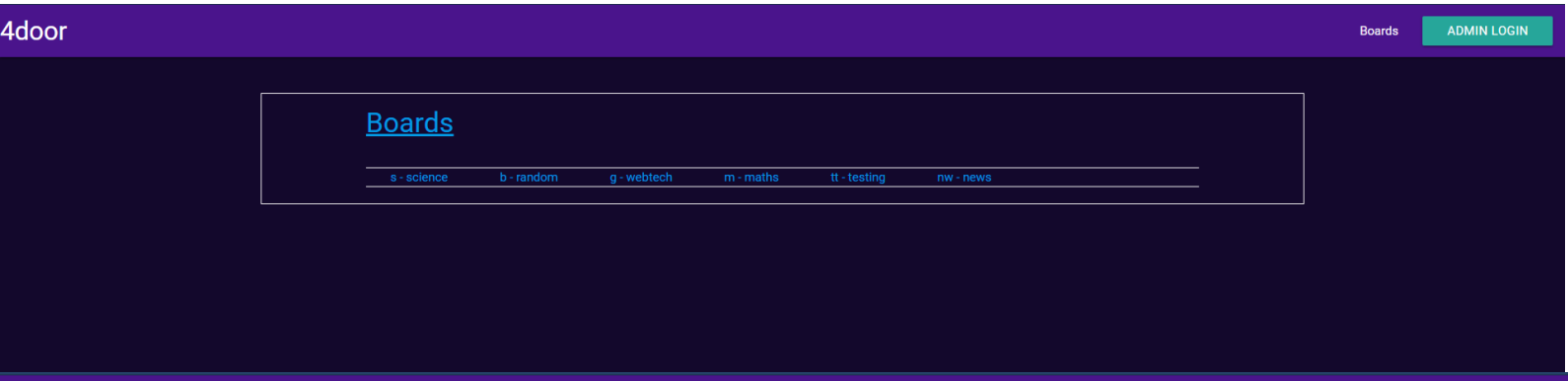
40402623@live.napier.ac.uk

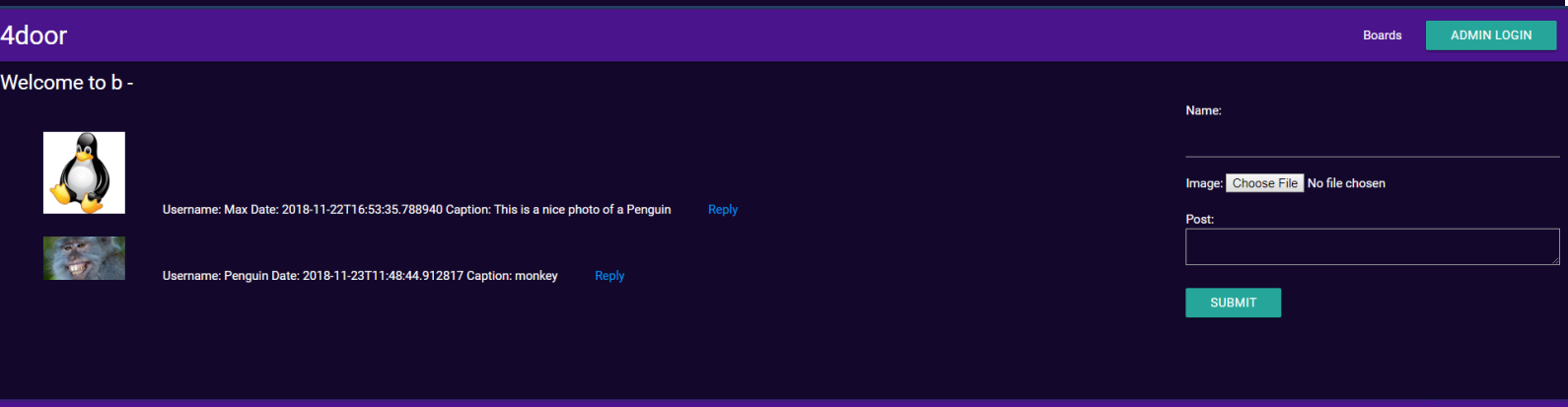
Edinburgh Napier University - Advanced Web technologies CW2 (set09103)

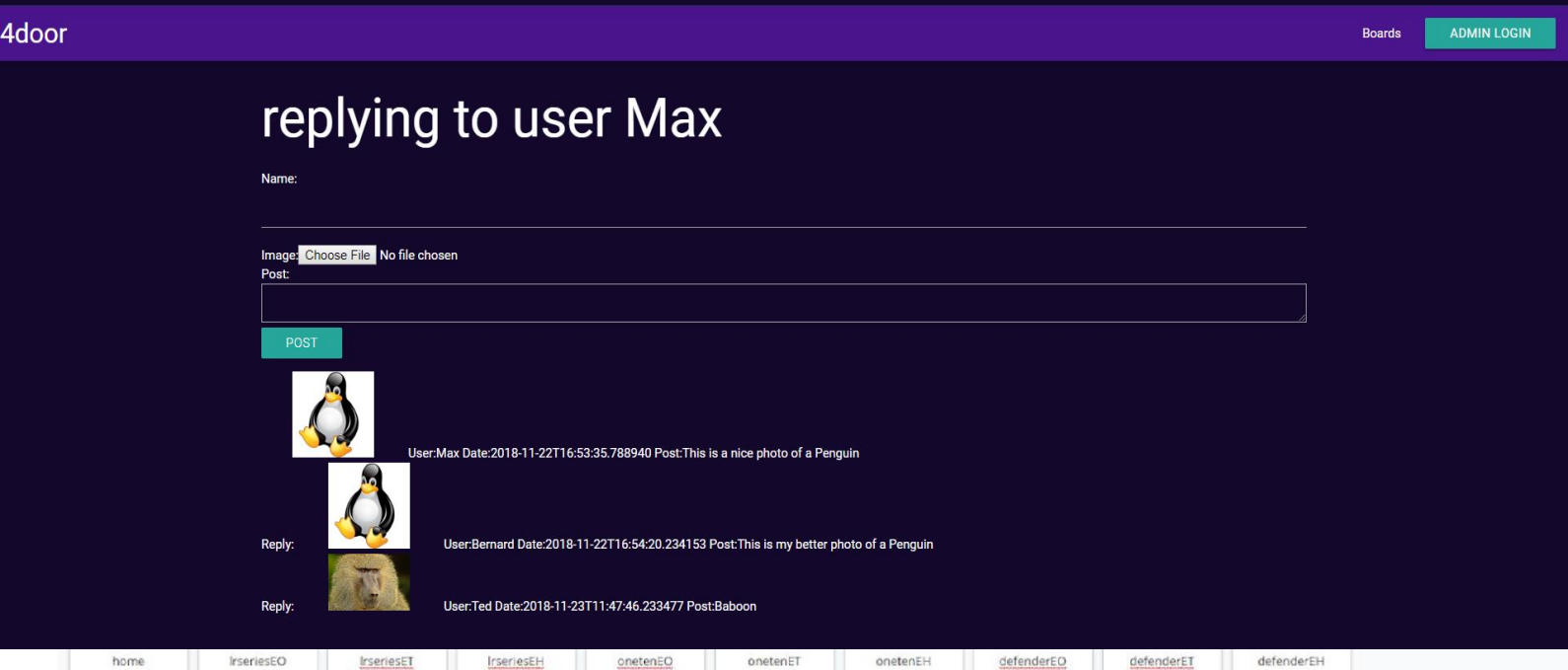
4door, chan/image board clone

Website Introduction

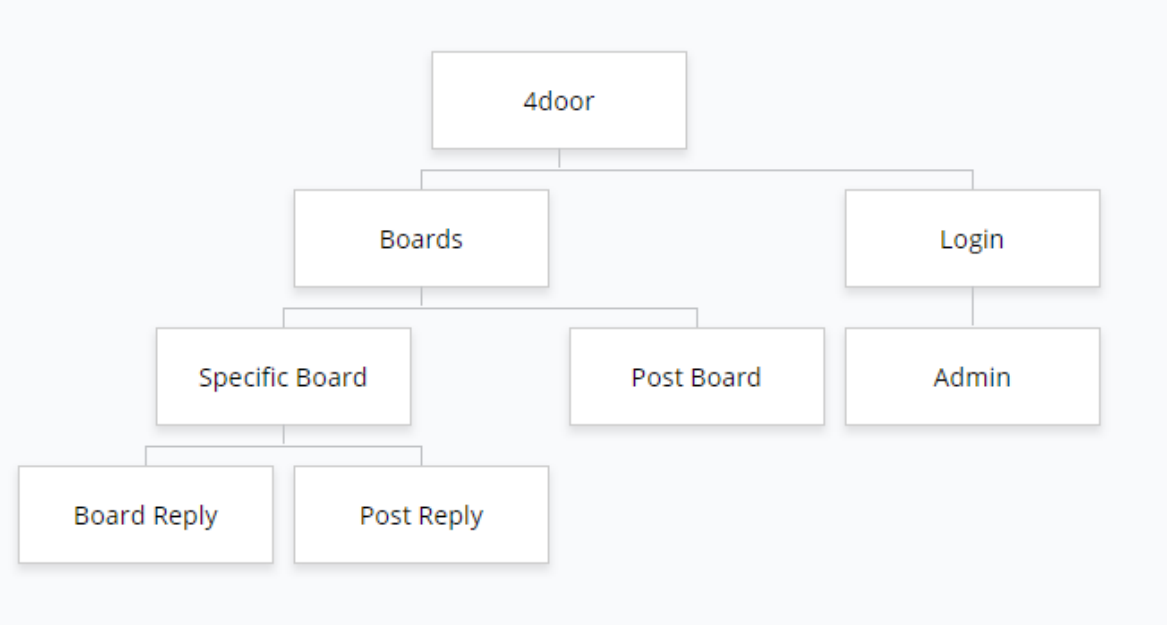
I created an image board that would allow users to add images and post a comment about the image to boards, these boards can be accessed by anyone and can be edited by anyone. In the boards a user can upload an image and comment about it, this will then insert what they have posted to a database, which is then displayed on the page they were on. The main purpose of this website is to share images and ideas with other internet users, something which has proven very successful in the past. The main reason I chose to do this is I believed it to be an interesting path to take and the functionality implemented on this website can hopefully be used in my future projects. Bellow can be seen the home page, then followed by a typical board page, this then follows on to a typical reply page.



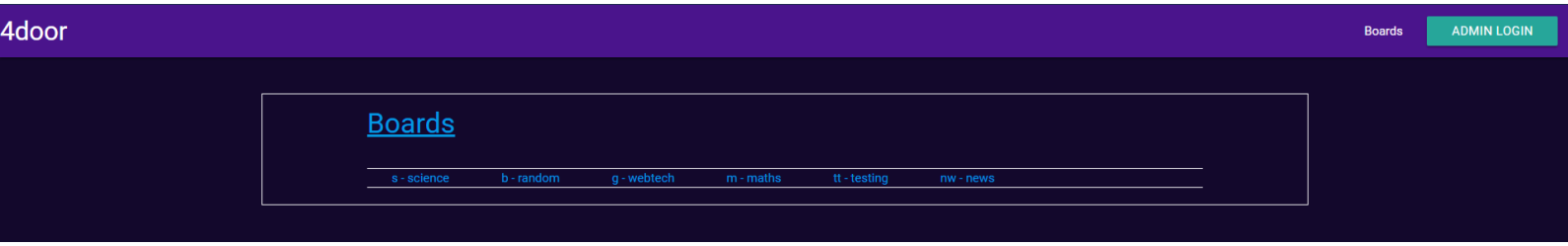




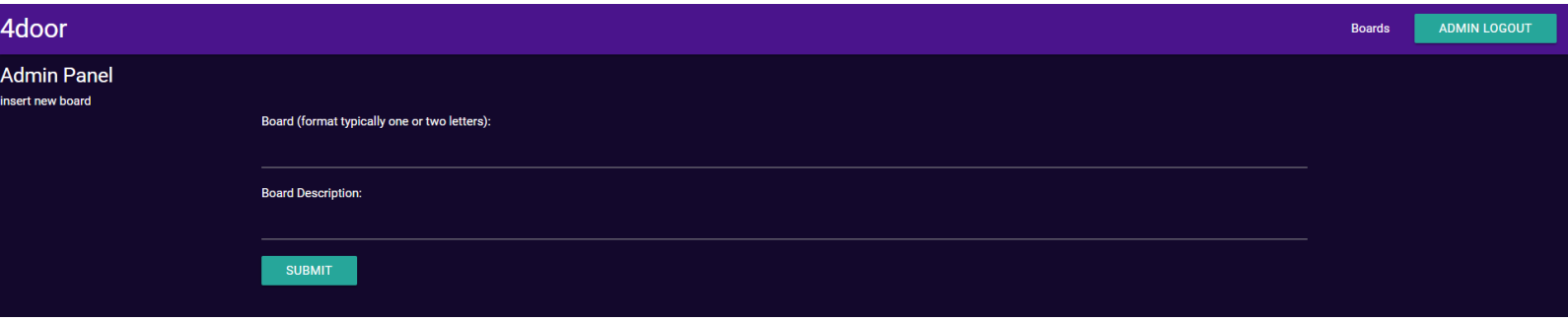
functionality, the website has an efficient folder structure with no excess files or pages doing redundant tasks.
Rather than create a logo for this task I simply used a small code image for the favicon to represent the task. With regards to the sitemap there are only two main branches, admin or boards. Within the admin panel an admin can login and add new boards and in the boards a typical user can post images and messages to the entire board. Past this point a user can also reply to the posts found on the board, this will open up a new page called reply, on reply all the replies made to that post can be found.



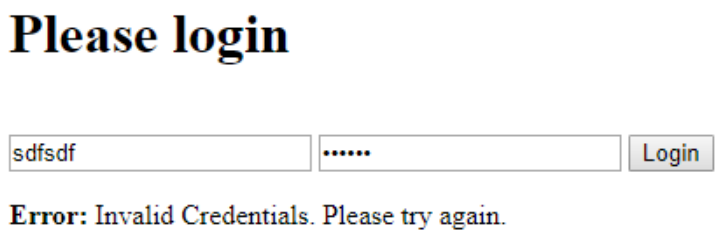
I would implement a purple design style for the website, having decided it would standout and have a unique feel. I went for a colour scheme of #4a148c light purple for headers, and #13082c dark purple for the body, this would be coupled with white text for standard text and #039be5; light blue for board links.



The structure of my webapp is structured in a way that allows the user to easily access the features I have made, whilst allowing admins to safely and securely navigate to a section where they can make changes to the whole app.



When an admin clicks on the admin login button, they will have to enter in a username and password. If they do not enter the username or password correctly, they will receive a message alerting them that the username or password is incorrect. If they enter the username and password in correctly then an admin panel template will be rendered, from this page they are able to make changes to the webapp. The logged in admin can create new boards.

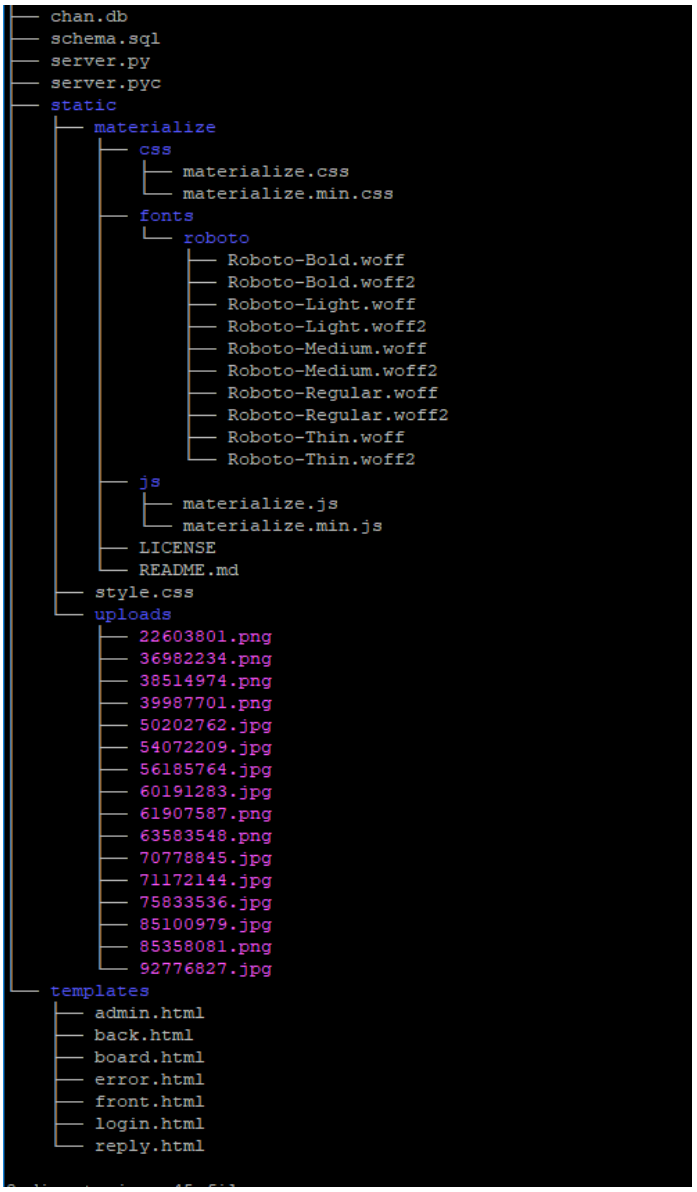


Non-admin users have a different path when using the website they will open the page and will see the list of boards that already exist, any new boards will be displayed here since they are linked to the database. Once a user clicks on one of the boards they will be sent to that board, form this page all the content from the database will be displayed if it matches the boards ID. From this page the user can view all of the current posts from users (name, image, comment, time of post) as well as create there own new post, there is a file uploaded that will only accept: jpg,jpeg,png or giffs. Once they post this, they will be able to see their post on that board.

Adding to what the user can do on the boards page, the user will see a link that says “reply” on each of the posts on that board. If the user clicks the reply link, they will be taken to a reply page for that specific post, from here they can see all the other previous replies to this post as well having the ability to post their own reply, they have the choice to add an image when inserting a reply.

With regards to security I do not believe this site has any glaring issues, the admins log in securely using a username and password that is stored server-side and I do not directly keep user information as the user will add a name every-time they enter a post or a reply, no other personal information is kept, e.g. card information, house address, age, gender, etc. I believe that in the future I may want to add users own pages, for this I would have to implement more security to keep their information secure as well as remembering to adhere to the recently updated data protection act of 2018 http://www.legislation.gov.uk/ukpga/2018/12/contents/enacted.

The full folder structure is represented by the following.



Enhancements

If I had to improve this website, I would improve two main components the admin panel and the styling. The admin panel despite many ten's of hours has a major bug that wont allow the data being pasted through it be sent to my database, this is definitely just an error with my code however I have been unable to fix this problem despite getting similar functions working with no issue. I believe the issues stems from the python script that receives the post from the admin panel, regardless I would like this to be fixed. Adding the enhancements I would make to the admin panel, I would implement

more features rather than just adding boards, I would also like to remove boards, remove posts and remove replies. I feel this would give the admin a lot more functionality to play with.

The styling is less important than admin functionality but still an important consideration when making a webapp that many people might be using for extended periods of time. The main reason for the styling not being up to scratch is on account of personal time constraints rather than the course constraints. Given more time I would like to have made the specific boards and the replies pages look better, I would have done this by created rows and columns for the text and the images so the image sits to the left and text displays to the right in an ordered column. However this proved difficult to quickly do as the whole page already sits inside of a set of rows and columns, meaning upon trying to implement this the whole page ran into difficulties.

Critical evaluation

I built a webapp that allows users to view and post images and comments to one another. I did this by using python flask in combination with sqlite3 for the database. I began by creating the database, the schema for this is found below. This created my database with all of the tables and columns I would need for producing the website.

```
CREATE TABLE posts (
post_id integer PRIMARY KEY,
image_file text,
user text,
date text,
board text NOT NULL,
post_text
);
CREATE TABLE boards (
board_id integer PRIMARY KEY,
board_short_name text NOT NULL,
board_description text NOT NULL
);
CREATE TABLE reply (
reply_id integer PRIMARY KEY,
board text NOT NULL,
reply_image text,
user text,
date text,
post_text text);
CREATE TABLE replies (image_file text, user text, date text, board text, post_text text, replying_to text);
```

I would then go on to create the first set of pages such as server.py and the first templates such as front.html, I would also create the static folder and populate this with the stylesheet, materialize.css and the uploads folder for the images. Once this was done, I progressed to complete the reply page and towards the end I would attempt the admin functionality.

The boards page allows users to select whatever board they wish to use. The boards page contains a list of every board found on the database. Once the user clicks on any of the boards they will be redirected to a page that contains all the posts regarding the specific board they chose. This subpage is generated by selecting all the data from boards as long as the names of the boards match, so if you click on the board ‘random’ you will only see the posts from people posting in ‘random’

I created a form that allows users to post their own content into the database and have it displayed on the specific board.

I also created a reply link, this link allows users to reply to a specific post made by another user as well as reply to their own post. This is done in a similar way to the board posting, but the replies are all posted under each other in a stair style with each reply being moved further right to indicate the order of replies.

I feel that all of the current features work well, the functionality is there however I believe that the typical user experience is not as good as I would like, when a user makes a post or a reply they will be directed to a 'back' page that will redirect them back to the page they were just on. However, to view the post they just made they need to refresh the page, looking back I would like to have not had a redirect at all and instead implemented something that just refreshed the page upon clicking on post.

The adminpanel login works well, I decided I wanted to create this from scratch and I have yet to notice any major flaws in its functionality. If a user enters the data incorrectly, they will be sent back to the login page, if they enter it in correctly, they will have a new html file rendered on the current page, this is the adminpanel.

Once on the adminpanel however I am less happy with the result as the adding new boards does not work, if given more time this would be my main priority to work on as well as adding more features for the admin once logged in.

Personal evaluation

I believe I performed well in creating something unique and overcame many challenges to complete the task. I have to learn how to use SQLite and this was the first time that I could not directly interact with a database via a GUI, this was a strange experience in that I had to make far greater use of SQL. I also had to make use of lengthy python scripts, before starting this course I felt like a knew very little about python or for that matter server-side scripting however now I feel that I have a good start in progressing my understanding of python further.

One major challenge I have to overcome during this course was the image files not being saved the folder that I specified. This was a harrowing experience, spending hours daily trying to fix this one bug that was stopping my app from functioning, I would spend a long time looking at methods online for fixing this issue. In the end nothing worked, so I deleted my image save scripts and re did them from scratch, after about 15 minutes of re-writing the scripts the image saving was working. The images were being sent to my uploads folder with the correct generated name, this was perhaps one of the most rewarding moments in university.

Overall, I feel that I performed well, one thing I have discovered is how quickly time seems to past when I need to fix a bug or are trying to improve functionality. I believe that maybe rather than throwing myself into the deep-end and trying to code everything straight away, I should have spent more time in learning what I was trying to do and develop my own knowledge. Looking back, I am certain that spending more time on research would have actually saved me countless hours of painstaking trail and error.

References

1. Materialize https://materializecss.com/
2. Python https://www.python.org/
3. Python Flask https://www.fullstackpython.com/flask.htm l
4. Vim https://www.vim.org/
5. Linux https://www.linux.org/
6. Wikipedia https://en.wikipedia.org/wiki/Land_Rover_engines
7. Sqlite https://www.sqlite.org/index.html