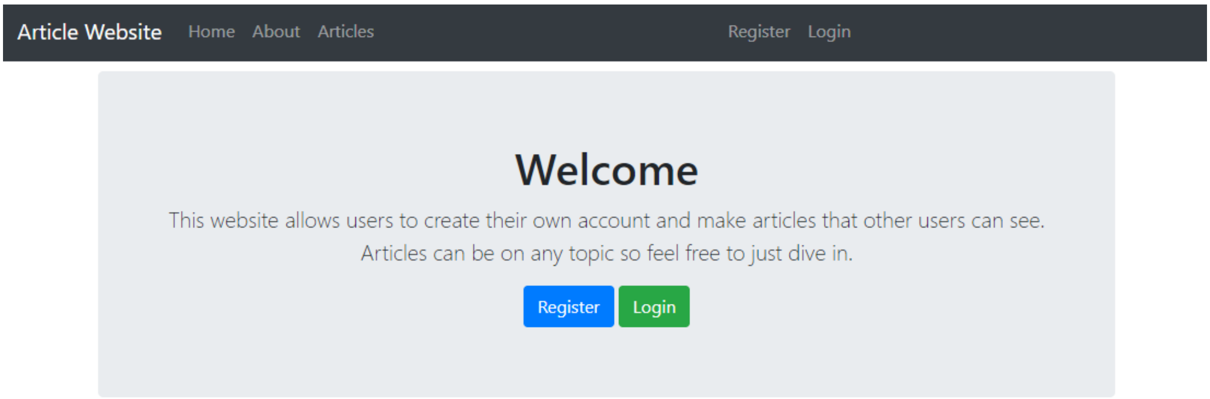
Article Website

Matriculation number: 40284597

Name: Robbie Lewis

Introduction

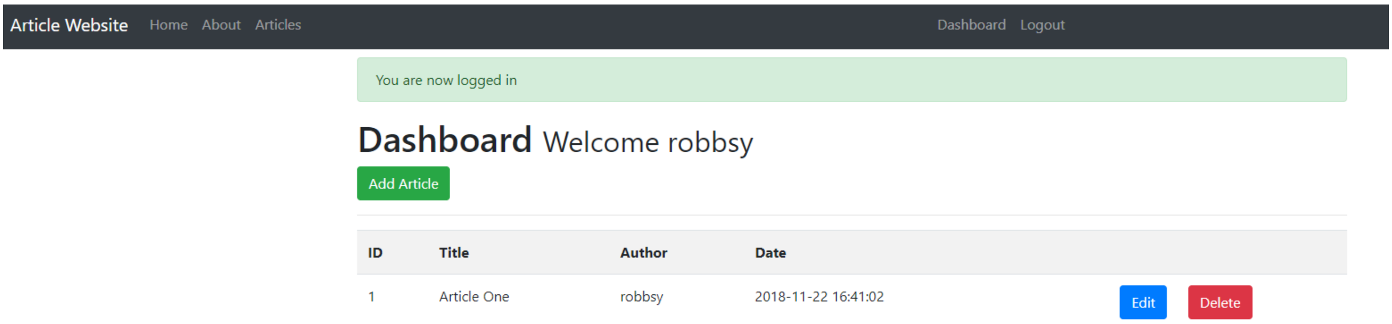
This website application allows users to create their own articles and share them with others. The user can register and then log in which allows them to create an article. Users can create many articles that can then be seen by others. An article can then be edited or deleted by the creator. The articles are created, edited and deleted on the dashboard page which is only made available to the user once they are logged in.



Welcome

This website allows users to create their own account and make articles that other users can see. Articles can be on any topic so feel free to just dive in.

This is the home page of the website. If the user is not logged in then they will be asked to Register or Login before they can create an article. The register and login buttons will not appear if the user is already logged in.



You are now logged in

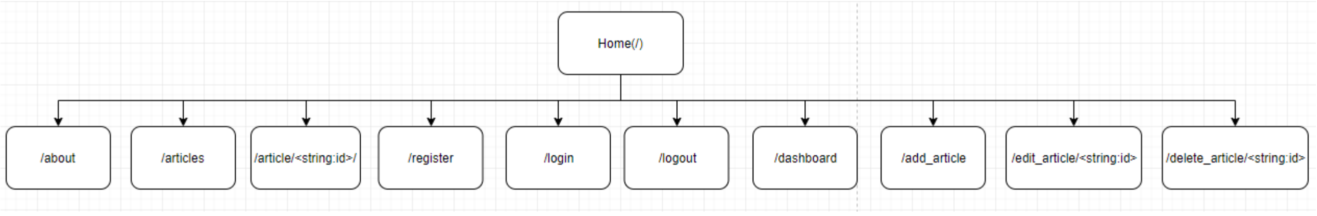
Dashboard Welcome robbsy

ID	Title	Author	Date
1	Article One	robbsy	2018-11-22 16:41:02

This is the dashboard page that the user is redirected to when they login. It allows the user to add an article and then edit or delete a current article.

Design

The website uses a number of roots and classes to navigate through the pages. There is a separate html file for each page and action eg. Add_article, edit_article.[1] The navigation bar and the flashed messages have their own html files too. This makes it easier to keep track of everything and if there are any errors, finding and fixing them becomes a quicker proccess. The dashboard is used to create, edit or delete articles. Articles can then be viewed by going on the articles page where they are all displayed in a list format. Clicking on an article will then take the user to that specific article where it is then displayed on a full page showing the author of the article and the date/time of its creation. This separates the creation and viewing of articles by having them on different pages. The registration proccess adds the users details to a database table called 'users' and then this information is then searched for when a user tries to login[2]. A users password is encrypted using sha256_crypt. This adds to the security of the users account. When a user is logged in, they have access to the dashboard where they can create their own articles. The Articles details are then added to the articles table in the database. When a user wants to edit an article, the information is fetched from the database and displayed in the textboxes. CKEditor is used to give it more functionality. Here is the URL hierarchy.



Enhancements

A feedback page could be useful as it would let the developer know what they could do to improve the website.

Comments could also be a helpful addition to articles. This would allow the authors to see what other users think of their article.

A rating system could allow users to rate their favourite articles and the highest rated articles would then be displayed on the home page.

The URL hierarchy could definitely be improved as every pages URL does not lead on from anothers.

Censoring the text used in articles would help to avoid any offensive language and would make it more user friendly.

A search bar could be added to the articles page so that users could search for a certain article or author.

Articles could be categorised into specific topics to make it easier for users to find any particular subject they wish.

[1] https://getbootstrap.com/docs/4.1/examples/starter-template/

[2] http://flask.pocoo.org/docs/1.0/patterns/sqlite3/

Critical Evaluation

The web-app built is a funcitonal article making website that allows users to easily navigate through the pages. The design is simplistic which makes it much clearer as to what actions can be carried out. The registration and login process is smooth and straight forward and the fact that articles can only be made by users that are logged in is a nice touch. When logged in, the register and login tabs on the navigation bar are then replaced with dashboard and logout. The articles page is good because it shows all the articles in a list format which allows the user to access the article they wish to read.

Personal Evaluation

I feel that my performance on this coursework was at a good level . I learnt a huge amount whilst making this web-app. It was a challenge to get the database working with sqlite3 as there were many links between the tables. Trying to get the users information linked with the article created was definitely a struggle. I ended up using row_factory to display the article details. I found that it is much better to make multiple html pages to store each action that could be carried out. This made it easier to make any changes and find out what was causing a problem. Wtforms became very useful in the registration process as it allowed me to set the variable types and validate the inputs.

References

https://getbootstrap.com/docs/4.1/examples/starter-template/

http://flask.pocoo.org/docs/1.0/patterns/sqlite3/