Adam Le Bay

Adam.lb@hotmail.com

WEB TECHNOLOGIES SET09103 cw2 REPORT

A.L.B WEBSITE REPORT SET09103

Contents

[An introduction describing the web-app, 2](#_Toc436602107)

[Login page 3](#_Toc436602108)

[Feed page 3](#_Toc436602109)

[Register page 4](#_Toc436602110)

[Design and Website architecture 4](#_Toc436602111)

[Enhancements to add 5](#_Toc436602112)

[Evaluation of the web-app 5](#_Toc436602113)

[Personal Evaluation 6](#_Toc436602114)

[Resources used and list of references 6](#_Toc436602115)

[Icons and Wallpaper 6](#_Toc436602116)

[Learning Ressources 6](#_Toc436602117)

## An introduction describing the web-app,

This Python Flask web app has been developed for the advance web technologies module at Napier University.

A.L.B aka Alligators, Lions and Baboons is a social media platform where everyone has the ability to post on a blog that then goes online. Each user has an account which he can create, login and logout easily. From this account, any user can post new stories about their adventures and experiences during trips around the world.

The website in designed for a smooth and easy interaction. It features a user friendly design and follows some aspects of the ISO standardisation rules ( Web Content Accessibility Guidelines) .

Every user can upload a profile picture that will then appear on its profile and its post (public and private).

The A.L.B website proposes content for unlogged in users and specific content to logged in users. You must be logged in to access these features otherwise user will be directly redirected at the login page where he can register if needed.

From the main Feed page, available for unlogged in users, is displayed all the most recent posts from all users of ALB. Each post specifies a title, a location and a time. This data could in future be used to regroup and order posts depending on requirements.

The web-app uses a SQLite database with 2 tables, one for users and one for posts. They each have an auto incrementing primary key and a set of text columns.

### Login page



### Feed page



### Register page



## Design and Website architecture

The design of a website in its architecture or visual aspect is a critical element that will determine it success and usability. As many website online, the A.L.B website uses a architectural design ready for a potential large number of users. The code is ran on a Levinux os which is usually used for large servers. The python runs on a local server created using Pipulate, and is reached through a windows internet explorer.

After using a text file as a database in my previous coursework, I opted for a SQLite database initialised in my python script. In Scheme.sql I created 2 SQL tables:

* User (idPK,username,password,)
* Post (idPK,Userid,title,post,location,time)

My database is then stored in a VAR folder under data.db.

The files needed for this web app are stored through 4 directories:

* The main ALB-website directory where the python launcher and my schema.sql is stored
* The static directory (for profile pic uploads and background images + style.css )
* The VAR directory for the database (data. DB)
* The template directory where all web pages in html are stored.

The CSS for this web-app is well optimised and most page uses the same body. The overall aesthetics have been aimed in order to attract a hipster type of community that would enjoy blogging and reviewing their recent trips.

The website is initialised by running welcome.py and the database is reinitialised in init\_db.py.

I used Jinja and Django to extend templates and display data in the html through for loops.

For my profile picture saving, I created a new folder in static that automatically renames the picture to the username and saves it as a .jpg .IT is then displayed later on specific user page + feed.

Finally, I implemented a password encryption system for security and data privacy.

## Enhancements to add

A.L.B is still on a primitive state, and many features could be added to the social media.

A first feature that the web app lacks is a profile page that can be heavily personalised and that can be viewed by other users of ALB (if users are friends)

To make it more interactive, a comment section should be added at the end of each post so that each user has feedback on its post. This way interactive discussions can appear.

A specific feature that I was unable to incorporate because of lack of time is a ranking system for each post and user. Every time a post gets up voted, his rate counter in the SQL post and user table increments of one. This way, users can then be ranked by their popularity among others.

Using a ranking system would be great for displaying most popular posts, and in this way propose quality/popular/funny content to new users.

As you may have noticed the goal of this website is to add as many features possible for the best user experience. The aim would be to be as complete as any recognised social media (Facebook, Trip advisor, Twitter…) Most of social media hit a peak when a chat and notification option is added. The chat and messaging development is quite complex in terms of programming but the final result is an increase of users coming back (checking for any new notifications or messages).

## Evaluation of the web-app

My website works well and is glitch free. User is constantly reoriented to a specific page depending on its login (session). Unfortunately the web app has flaws in its content that seriously limits the user. Unlike Facebook there are not multiple pages in order to view other user s content (home/profile page/friend and info Page…) A.L.B only has the main “feed” page where all data is displayed.

The huge flaw of this system is in the increase of users and the overpopulation of posts in the “feed” page. Unorganised and unspecific content for a user isn’t what makes him want to come back.

The web app also doesn’t give a possibility to a user to change his profile picture. This feature is essential and indispensable for a social media. Using the current system where the profile picture is stored in a PIC folder under its user id.

To conclude this app is a beta version of what could become a successful website. Unfortunately it lack essential features with which the user can interact.

## Personal Evaluation

After using Python flask and css in my previous website, I was quite familiar with most of the technical aspects of this coursework.

Nonetheless I did face the challenge of creating a database in SQL, this took me a lot of beug fixing and was very frustrating. Although SQL is an easy enough language it was a lot harder to implement/initialise in with python.

Unfortunately I have not had the chance to try other languages (eg. JAVA) that could have been implemented for better styling (menu overflow, interactive search).

I also learned a lot about scalability in css. My website is know a lot more centred and looks more professional then the previous.

Overall, I am quite pleased with the aesthetics of my web-app although I think it lacks personalisation features and “addictive” content.

A.L.B has a real potential if it manages to include smoothly a ranking system that ranks users by most popular. People could then compare and compete in a same Country/city/school to be the number 1.

## Resources used and list of references

### Icons and Wallpaper

<http://cdn.mysitemyway.com/etc-mysitemyway/icons/legacy-previews/icons-256/magic-marker-icons-animals/114677-magic-marker-icon-animals-animal-bird6-sc44.png>

<http://ma-magazinephoto.com/wp-content/uploads/2015/03/LANDES-SAVANE-LANDAISE-1>

http://cdn.desktopwallpapers4.me/wallpapers/digital-art/1920x1080/2/16182-world-map-1920x1080-digital-art-wallpaper.jpg

### Learning Ressources

<https://flask-wtf.readthedocs.org/en/v0.8.3/>

<http://stackoverflow.com/questions/9023488/build-error-with-variables-and-url-for-in-flask>

<http://flask.pocoo.org/snippets/54/>

<http://stackoverflow.com/questions/29629537/django-print-sqlite-db-data-in-template>

http://www.tutorialspoint.com/sqlite/sqlite\_python.htm

<http://stackoverflow.com/questions/452859/inserting-multiple-rows-in-a-single-sql-query>

<http://www.w3schools.com/sql/sql_datatypes.asp>

<http://www.techonthenet.com/sqlite/primary_keys.php>

<http://stackoverflow.com/questions/415511/how-to-get-current-time-in-python>

<http://stackoverflow.com/questions/3559559/how-to-delete-a-character-from-a-string-using-python>

<http://moodle.napier.ac.uk/pluginfile.php/955615/mod_resource/content/10/workbook.pdf>

<http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=58625>

http://flask.pocoo.org/docs/0.10/patterns/fileuploads/

ps:Really enjoyed this course, hope to work with you next year.