PixlHaven

Scott Davidson
40277878@live.napier.ac.uk
Edinburgh Napier University - Advanced Web Tech (SET09103)

1 Introduction

PixlHaven is a social website about art, pictures and photographs. The website allows users to browse the latest uploads, search for specific keywords, and if they register an account, they can upload new images or edit ones that they've already uploaded, add other users as friends to follow their uploads, keep a list of notable pictures by adding them to their favourites list and add comments to images if they have something they want to say about it. It was written with the Flask framework using python. It utilises HTML and CSS to provide a nice interface for users. Throughout the website, links will direct users to either other users pages, or image pages, to help them navigate the site.

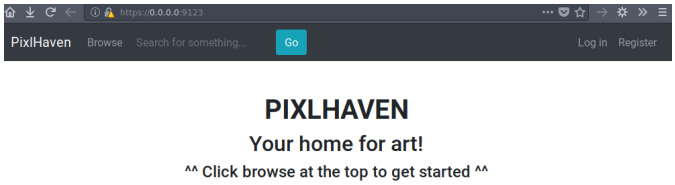

Figure 1: The home page of PixlHaven

2 Design

PixlHaven was designed to be easy to use and effective at providing the user with the content they choose. It uses standard features such as a navigation bar and search functionality, where the rest of the page is utilised for displaying content. The structure of the URL hierarchy has also been designed in a way that is both simple and sensible, even allowing the user to navigate some of the site just using URLs.

Using Jinja, a templating engine for flask apps, PixlHaven is able to dynamically generate page content with information stored in an SQLite3 database, allowing a standard page layout but providing different content based on factors from the users. This allows the website to scale as more content is added to the database. In the absence of information, or if the user attempts to reach a page that is not available, error handling has been implemented, redirecting the user to a page not only showing the error code, but a message that tells the user what the error signifies.

Data has also been structured to minimise redundancy, where the database contains only the information needed, such as usernames or upload dates, and other data, such as file paths for the images, are generated with a combination of these.

Supplying data to the website is mostly secure, as it uses HTTPS to communicate with the user. The amount of sensitive data required is minimal as well, requiring just a username, email and password. Passwords are hashed before being used for any purpose, minimising the chance that they could be cracked. E-mails, on the other hand, are stored as plain text. However, they are only used within the server, and posted from the user, so there is no leakage of information that shouldn't be displayed.

Users can also have a rank, allowing moderators and admins. They have elevated privileges, allowing them to filter content that goes against the rules.

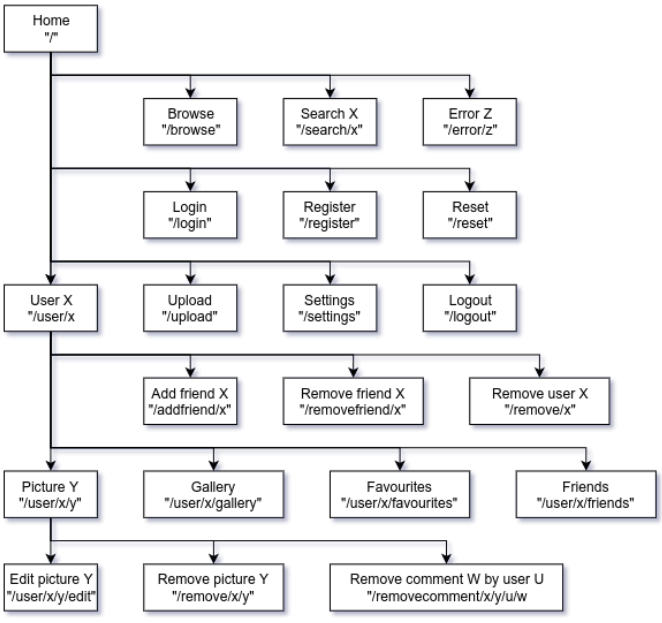

Figure 2: The URL structure

3 Enhancements

PixlHaven has enough features that it can be used as is. However, there are some more modern technologies that it could employ. This might include live updates; when the user is on a browsing page and a new entry is uploaded, it could

get pushed to the users on that page. This would provide a more dynamic experience for users.

Another enhancement could be allowing users to send a private message to other users, or to upload text instead of images. As of right now, the only way to communicate to users is through the comments section of an image's page. This would allow users to feel more connected to their friends or artists they like, and allow artists to connect to their fans, providing a social platform that extends from just sharing pictures.

Searching could include fuzzy search technology, mitigating user error by finding similar words or patterns in text. This would make the search feature more useful and possibly also predict some terms for easier navigation and better user experience.

Styling could also be added, allowing the users to customise how the website looks, for example day and night themes. This would make it more accessible to users who might not like a bright screen at night time, or even users who require a higher contrast for the websites they visit. Styling could also be improved to make elements more consistent across the website.

Rank modification could be improved by allowing admins to change user ranks and other such information directly within the web-app, as currently they must directly connect and modify the database. However, given the nature of such an action, it won't happen frequently, minimising the impact of this missing feature.

Security features such as e-mail validation, password resetting and official HTTPS certificates. Currently, the website uses a temporary certificate for HTTPS, which warns the users when they access the site that it might not be secure. E-mail is currently only used for resetting passwords, although no e-mail is sent out. This would be a feature to add later, when website functionality grows and security is required to be more thorough.

4 Critical Evaluation

PixlHaven has a sensible URL hierarchy that works well when trying to navigate to users pages, for example. This should keep users from getting too confused and allows page content to be independent from page layout.

The friends and favourites features are probably the best, as it allows users to keep track of notable images and see the latest uploads by their friends. Entries on these pages are ordered by date descending, so the latest will always be at the top. Comments are also ordered by their date, so users can see what the latest word is about an image.

Another feature that works well is user rank. Each user has a rank that denotes whether they are a normal user, a moderator, or an admin. This allows content to be filtered, as moderators can remove comments and images that violates rules, where admins can also remove accounts. A missing feature, however, is the ability for the site admin to promote users to moderators, this would save the site admin from having the specifically access and change records in the database.

The simple design of the website allows the user to browse content without getting distracted. However, The design is lacking in a few ways. Pages might not be displayed consistently across browsers and devices, and it could certainly use a polish in terms of making each element fit aesthetically with each other.

One issue that PixlHaven could face is when many users try to post content at the same time. This means the database might be updated differently by different users, leading to inconsistencies.

5 Personal Evaluation

I feel that my performance was good, expanding upon what I applied to the previous work and using what I've learned since to create a robust enough application that can support the functionality required. Some problems faced included working with a database, however, the documentation[1], along with good data planning, meant that such functionality became very modular, making it easy to use once implemented.

Another issue faced was sessions and encryption, as those are integral to the security of users and functionality of user management. Managing to abstract session management made working with users quite trivial, and encryption, once handled as soon as it needs to be, didn't affect the rest of the application, allowing it to be easily implemented.

There are some features that haven't been implemented such as e-mail verification and password resetting through an e-mail, as this would require a few factors that are missing, not least of which an e-mail address for the web-app itself to send them from. There are others that aren't fully implemented, such as admin controls for managing other users, or just haven't had enough time to be polished, such as as the user interface. Given more time I would look into adding and improving these features, even going so far as to learn new web-page styling packages to add a style that would identify the app as a brand.

References

[1] "Sqlite3," *SQLite3 database documentation*.