MILKY WAY ASTRONOMICAL OBJECTS

(MWAO)

ABSTRACT
This is a website created by Adrian Vazquez for the Napier University, Edinburgh, Scotland. It is created with a Levinux server, python, bootstrap (HTML+CSS+JavaScript) and SQLite3 database.

Vazquez, Adrian
Advanced Web Technologies

Introduction

The web application is created to collect all the known objects discovered in human History within our own galaxy, the Milky Way. The objects are catalogued into different categories:

- Solar systems
  - Planets
    - Satellites
  - Asteroids
  - Comets
- Stars
- Black holes
- Constellations
- Nebulas
- Star clusters

Pages

Objects "/objects"

Objects is the main content page and contains all the objects included in the database. Here only we can see some of the information we stored such as type, name, size, mass, distance and discoverer.

Also at the left in a sidebar we have all the categories that we mentioned before. These categories are link which the user can navigate through, and it will display only the objects inside the category chosen.

Each element is clickable, and once you click in one of them, it goes to "/objects/view/id".

Login "/login"

The main reason to create an admin account for this web applications is because I want two types of users: A visitor which is going to search information about different objects, and an admin user which is going to insert objects, edit and remove them.

If we click on the "glyphicon-user" icon, on the left of the main menu, there will be a dropdown where you can click on login. This link will send you to the login page:

The credentials are:

- User: admin
- Password: secret

Views/Edit

When we select any of the objects displayed before, we can see all the information of that element. Here there are two options:

- Login:
  If we previously login on the website we can see that now the elements are displayed inside a form. We can edit the object chosen and click the button update to make the changes. After clicking "Update", it will go to the same page to check that the element has been edited.
- Not login:

If the user does not login, it will be able to see all the elements, but cannot be modify.

Also, the user can go back using the breadcrumbs to the category of the object or to the main objects page.

About "/about"

About contains small introduction of myself and the project.

Contact "/contact"

Contact page contains information, and links about the author of the web application.

Design

The web application has a very simple structure:

- /
- /about
- /contact
- /login
- /objects
- /objects/category/<category name>
- /objects/category/view/<id of the object>
- /objects/insert

I created a very simple layout for the web application, because the information must be must important, and must be readable, and very accessible. There are four main elements on the website:

- Main navigation
- Admin navigation
- Content
- Footer

These elements are created with Bootstrap and I implemented modifications creating a new CSS file.

Improvements

1. Create a table for the users.
2. Upload images.
3. Improve design to make it more different from basic Bootstrap
4. Improve the objects table, and create many more tables. There is much more information that can be provided about these objects.
5. Reorganize the categories.
6. Documentation.

Personal evaluation

In my opinion the web application contains the minimum requirements. Although there many features that I would like to implement but I had no time to finish.

It has been very hard to understand the communication between the database (SQLite3) and Python, but once I complete the main task for the admin such as display, insert, edit and delete, I could move faster.

However, I think creating this web application has improve my skills to learn new languages and I believe that I will keep learning Python in the future.

Resources:

- Bootstrap: getbootstrap.com
- Stackoverflow: http://stackoverflow.com/
- Teaching materials: http://siwells.github.io/teaching_set09103/
- Flask documentation: flask.pocoo.org/
- WinSCP: https://winscp.net/eng/index.php
- DB Browser for SQLite: http://sqlitebrowser.org/
- SQLite3: https://sqlite.org/