

The Swords of Game of Thrones

Gregor Bell

40200288@live.napier.ac.uk

Edinburgh Napier University - Advanced Web Technologies (SET09103)

Abstract

The aim of this coursework was to create a web-app that displays a directory of information about a given subject. This report details the core features of the web-app as well as how the system is structured. The report also identifies enhancements that could be made to improve the web-app and evaluates elements of the web-app that work well and elements that don't.

Keywords – SET09103, Advanced, Web, Technologies, Edinburgh, Napier, University, The, Swords, Of, Game, Of, Thrones, Report

1 Introduction

The purpose of the web-app that I have created is to give a directory of the swords featured in the universe of television show "Game of Thrones".

I have implemented various features that I feel supplement this purpose and enable the user to positively interact with the web-app. These features are detailed below:

1.1 List of Swords

The main feature of the web-app was to present the user with a list of the swords featured in "Game of Thrones". I created a list of swords with the aid of the "Game of Thrones" wiki pages [1] and saved it into a JSON file. After loading this JSON file into the web-app, I then rendered the data in an HTML template.

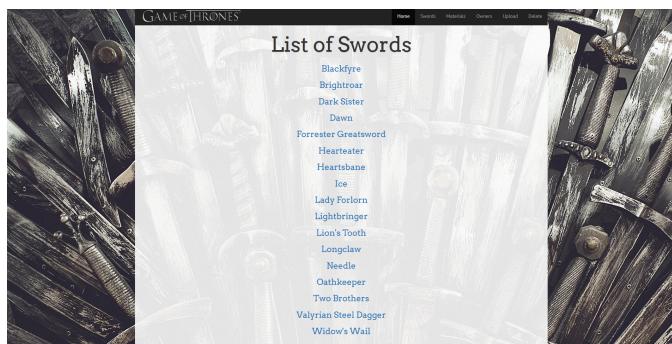


Figure 1: **List of Swords** - The list of swords displayed by the web-app

1.2 Sword Details

A natural extension of listing each sword was to provide significant details for each weapon that could be accessed at the user's behest. This was achieved by adding the following information to each JSON object: the material that the sword was forged with, the owner of the sword and the current status of the sword. This information was then rendered into an HTML template along with an image of the sword. Each image is stored in the images subfolder of the static directory.



Figure 2: **Sword Details** - The information displayed for each sword

1.3 List of Materials

Another feature that I thought would be useful for users interacting with the system was the ability to list the materials used to create each sword. The implementation of this is very similar to creating the list of swords, however, instead of displaying the sword names, it displays a list of the material names. Duplicates have been removed from the list and users can click on the material name to obtain a list of swords that have been forged from the respective material. Clicking on these sword names then takes the user to the relevant sword details page.

1.4 Upload Custom Sword

Users also have the ability to upload custom swords. This is accomplished using a form in an HTML template where the users can enter a name, material, owner and status of a sword. Users can also attach an image of their sword. The web-app then requests this form data using the POST request method and writes the updated sword list to the JSON file. The image is saved to the images folder on the server. Users can then view their sword in the swords list, as they would any of the existing weapons.

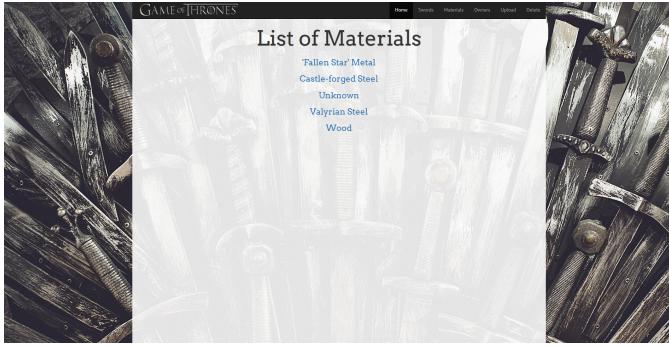


Figure 3: **List of Materials** - The list of materials displayed by the web-app



Figure 4: **Upload** - Form that enables users to upload a custom sword

1.5 Delete Sword

The final core function of the web-app is the sword delete feature. Users can specify the name of the sword that they wish to be deleted and the relevant object will be removed from the JSON file.

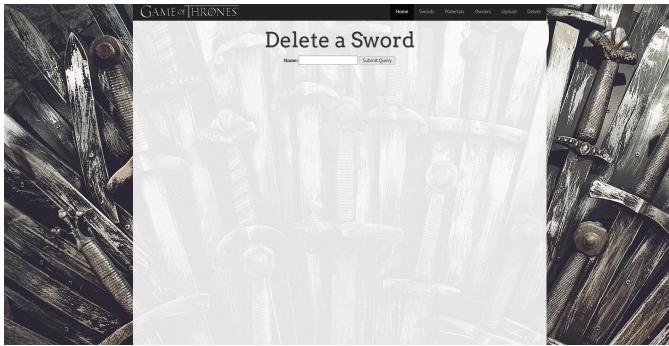


Figure 5: **Delete** - Form that enables users to delete the specified sword

2 Design

The structure of the web-app is illustrated in the figure below:

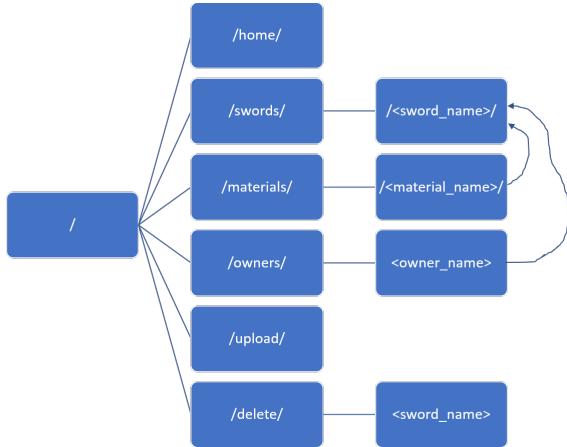


Figure 6: **URL Hierarchy** - The hierarchy of URLs used in the web-app

2.1 /

/ is the root page of the web-app. The only purpose of this page is to redirect the user to /home/

2.2 /home/

/home/ is the home page of the web-app and the initial page that the user will land on when accessing the web site. This page details the function of each button on the navbar and clicking these buttons will forward the user to the relevant page. Clicking the "Game of Thrones" logo in the top-left corner will take the user back to /home/.

2.3 /swords/

/swords/ is the main page of the system. The page gives a list of the swords featured in "Game of Thrones". Each sword can be clicked and the web-app will forward the user to the relevant /swords/"sword_name"/ page, substituting "sword_name" for the name of the sword that the user clicked on.

2.4 /swords/sword_details/

/swords/sword_details/ is an automatically generated page displaying information about each sword. The sword details are stored in a JSON file and fed through to the web-app using the HTML and Jinga2 code below:

```

1 {% for sword in swords %}
2     {% if sword_name == sword["name"] %}
3         <h1 class="text-center">{{ sword["name"] }}</h1>
4         <div class="sword-text">
5             <ul>
6                 <li><p><b>Material:</b> {{ sword["material"] }}</p></li>
7                 <li><p><b>Owner:</b> {{ sword["owner"] }}</p></li>
8                 <li><p><b>Status:</b> {{ sword["status"] }}</p></li>
9             </ul>
  
```

```

10          </div>
11          <div class="text-right sword-image-div">
12              
15      {% endif %}
16      {% endfor %}

```

An image is also fed through to the web-app from the images folder.

2.5 /materials/

/materials/ gives a list of the materials used to forge each sword. Users can then click on the name of a material and the web-app will forward them to /materials/"material_name", where "material_name" is substituted for the name of the material that the user clicked on.

2.6 /materials/material_name

/materials/"material_name" gives a list of the swords forged with the specified material. Users can then click on the sword name to access the relevant /sword/"sword_name" page.

2.7 /owners/

/owners/ is identical to /materials/, the only difference being that instead of a list of materials being displayed, a list of sword owners is displayed.

2.8 /owners/owner_name

/owners/"owner_name"/ delivers a list of swords owned by the specified owner. Each sword can then be clicked on to access the relevant sword's details.

2.9 /upload/

/upload/ presents the user with a form to create a custom sword and append it to the existing JSON data. Users can also upload an image which is saved in the images folder. Once the user clicks the submit button, they are forwarded to /swords/, where they can see their custom sword in the list.

2.10 /delete/

/delete/ provides the user with a form to delete a sword from the system. The user can enter the name of a sword and click submit. The relevant sword is then removed from the JSON file and the user is forwarded to /swords/, where they can see that the selected sword is no longer in the list.

2.11 /delete/sword_name

/delete/"sword_name" serves as a way for the user to bypass the deletion form. Users can specify the sword they wish to be deleted as "sword_name" and it will be removed from the JSON file.

3 Enhancements

There is a number of enhancements that I feel could be implemented on the web-app, both to improve its functionality

and enhance the user's experience.

3.1 Design Improvements

Although I am satisfied with how the overall design of the web-app turned out, certain aspects leave a lot to be desired. A number of the pages, namely the home, upload and delete pages, do not make efficient use of the available space. This results in a lot of blank space that is jarring to look at. I have tried to combat this issue by reducing the opacity of the background texture so that users can see the background image, however making better use of the space available would be a more desirable solution.

Another design improvement that I would like to implement is a dynamic favicon that changes depending on what weapon the user is currently viewing. Currently the favicon does not change and is static throughout the web-app but in the future I would like the favicon to represent the House of the weapon being viewed e.g. if the sword is currently in the possession of House Targaryen, the favicon would display the Targaryen sigil. This is a small change but I feel that it would benefit the user's experience.

3.2 Additional Sword Data

The web-app currently displays minimal information about each sword. This could be improved by adding a description of the weapon as well as detailing the sword's history and lore. Implementing additional sword data is a simple enhancement as all that is required is adding additional fields to each JSON object.

I would also like to potentially embed a 3D model of each sword into the web-app that users could interact with. This would enable the users to obtain a much better grasp of the sword design and dimensions. This feature is much more problematic to implement, however, as my 3D modelling skills are limited and there are no detailed measurements of each sword that could be used to create a model.

3.3 Search Bar and Filtering

I feel that a search bar would be of great benefit to this system. This would enable users to search for a sword directly, rather than having to navigate through the system manually each time. I also feel that it would be beneficial for users to be able to filter their search by material, owner or status. This would mean that if the user did not know the name of the sword that they are looking for but did have some other information about the weapon, they would still be able to utilise the search functionality.

3.4 Account System

An account system would be of benefit to this web-app for a number of reasons. Users would be able to create their own account and use this to declare their favourite weapons and monitor the swords they have created.

Administrator accounts could also be created to securely manage the website data.

4 Critical Evaluation

I am pleased with a number of the web-app's features, however, there is also some aspects that have been implemented to a poorer standard.

4.1 Use of JSON

The feature of the web-app that I am most pleased with is the use of JSON as a storage system. Originally, each sword object was stored in an array contained in the main python file. This worked well but as I added more swords to the array, I quickly realised that this was not an effective long-term solution. Storing data in the main system file looked messy and as the array expanded, it became harder to maintain both the data and the rest of the code in the file.

Storing data in a JSON file is a more scalable solution as it effectively serves as a small database and can be read from and written to with ease.

The issue with the current implementation of JSON in the web-app is that when a sword is uploaded or deleted, the entire JSON file is overwritten. This is demonstrated in the code below:

```
1 with open('static/data/swords.json', 'w') as swords_json:  
2     json.dump(swords, swords_json)
```

This works well with small amounts of data and the information is served to the user quickly. If the JSON file had a large number of objects stored in it, I believe that overwriting the JSON file every time would take a while, resulting in a slower experience for the user. An alternative solution, at least in the case of uploading a new sword, would be to append the new data to the end of the JSON file, rather than re-writing the whole file.

4.2 Uploading and Deleting Swords

I am also pleased with both the upload and delete features of the system. These features provide the user with a more significant method of interacting with the web-app and they are able to see the list of swords update in real-time for instant satisfaction.

The upload and delete methods both re-write the data stored in the JSON file so that changes apply across the site and persist between sessions. An issue with users having the ability to delete data from the underlying JSON file is that there is nothing stopping someone from deleting all the existing data from the file, leaving the website empty. A way to combat this would be to restrict the delete function to administrators.

5 Personal Evaluation

Working on this web-app was an enjoyable and interesting experience. I had never programmed using the Python language so this coursework provided a useful insight into Python as well as the Flask microframework. This web-app was also instrumental in improving my knowledge of the VIM text

editor and VIM is now my default editor for most coding projects.

The main challenge for this web-app was implementing, and subsequently interacting with, the JSON file. I had not utilised JSON files in previous projects so I had to research how to load JSON data into Python. After that, it was a process of trial and error before I was eventually successful.

In future coursework of a similar nature, I would like to make better use of the help available in the practical sessions. I had a number of queries that I resolved myself eventually, but I may have been able to resolve these more quickly if I had requested help in class.

6 Conclusion

References

- [1] Wiki Contributors, "Swords," 2017. [Online; accessed 23-October-2018].