Personalised Adaptive RSS Reader and Recommender

ICS2208 Intelligent Interfaces 1

University of Malta

Jenny Attard 16696G

Kristina Catania 85196M

Edward Fleri Soler 247696M

Contents

Introduction

With the advent of digital media and increased communications, accessing news and information at the click of a button has become the norm. Recent years have seen all popular journalism and news outlets shifting focus towards the web, reaching customers around the world at a fraction of the cost, while maintaining a steady revenue from advertisements and subscriptions. With so many news outlets to chose from, users may find it hard to find and tap into content which is most suitable and entertaining to them. This is where RSS Feeds, like our own, come in.

In this report, we shall present our teams RSS Feed site [1] and walk through the project procedure. Our intelligent site is designed to act as a central news hub, where users may access news content most suitable to their needs. Over time the system learns and adapts to the characteristics and preferences of each individual user, providing a better and more personal user experience. The main focus of this project is to explore adaptivity and recommendation features of such a site, and design a fully functional specimen.

We shall start by outlining the aims and objectives of this project, followed by the design and planning process which the team has undergone. Following, we shall describe the procedure involved in creating such a system, including the problems encountered along the way. We shall then end with a discussion of the functionality and usability of this site, together with the possible improvements that this project may benefit from in future implementations.

System Specifications and Descriptions

The purpose of this assignment is to develop an RSS reader and recommender. Here multiple news sources are combined to produce this diverse website which includes various categories, hitting the interests of many potential users. Adaptivity is the main focus in this assignment so as to make the users experience a simple and enjoyable one. Keeping in mind that the website is to be made for each individual, various features such as deleting and adding new links shall also be available. Adaptivity comes into play with the help of priorities given to each article as well as each link. This allows the system to keep track of any current preferences the user seems to be browsing through and then reference back to this when recommending its own articles.

System Design

It is a key feature that this type of webpage will be as simple as possible but adequate for all type of users as well as very efficient. For this reason the group decided that a user will be greeted with both the login as well as the registration page to make it easier to use especially for a first time user. In the latter this person will be asked to input basic information such as a name and a surname so that the system can build information and

collect data about the end-user. In this case JavaScript is used to help catch user errors such as a mismatch in the entered password.

In the meantime while the user is inputting these details the system will fetch both the default links and categories which have to be present at the start of the login page together with the information entered beforehand. Moreover the welcoming page is essential to this assignment since it involves the recommended articles which at startup will be the interested articles of the users' selection inputs. It is important to note that these articles are the ones having the highest priority.

A feature implemented on this webpage is that a user can view articles in one of two ways either by selecting a category or from a side menu that will show the articles of a specific link such as Times of Malta. Another feature to mention is that a user can both delete a link from the side menu and even enter a new one. For both actions performed the webpage will adapt to such changes and will remove the articles of a specific news link or fetch articles of the new link respectively.

Having such an option is of utmost importance since the end-user will have a personalised webpage with editing features that can be used for customisation purposes.

While traversing through the article links a user can view the article by clicking on the content leading to the actual article in the news link, this does not mean that a user will have to go back in the rss webpage to keep browsing as the article will be opened in a new tab. Although this is sometimes taken for-granted this action is quite helpful for such users because they can keep traversing the actual rss page and also open new articles without losing track of where they where.

To make this website even more user friendly and intelligent, articles which were clicked on will be set to watched, to distinguish the difference between unwatched and watched articles. This content will have a lower opacity in the latter case.

When all the needed actions have been performed the person using this website can log out and log back in at any time. The priorities of the articles are constantly changing every time an action is performed by the user this means that the recommended page will have the most interested articles which are adequate for the user.

Such a priority also works when a person is logged out meaning that the next time a person logs back into the account and inputs a new link this priority will have a higher priority than the rest of the links.

System Implementation

As was mentioned above, the site consists of two main web pages; a login/registration page, and the main RSS Feed page. We shall now walk through and understand the various components which form these two pages, and discuss how the site was brought together to form a working specimen.

- Login/Account Creation

Our site enforces subscription by means of a free account to access our services. Attempts to access the feed page without creating a new account or logging in are prevented. Therefore, the first main team focus was on the login and creating page. Since PHP was found to be the most suitable language for back-end handling, sessions were implemented to store access credentials. Users are only allowed to access the RSS Feed if their session certificate is set. Upon the correct entry of credentials in the login page, the user's session certificate is set, and they are redirected to the main page.

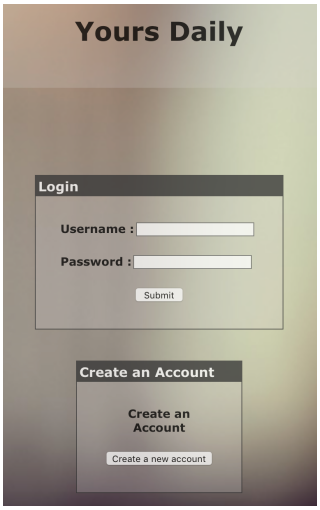

Fig 1.1: Login page

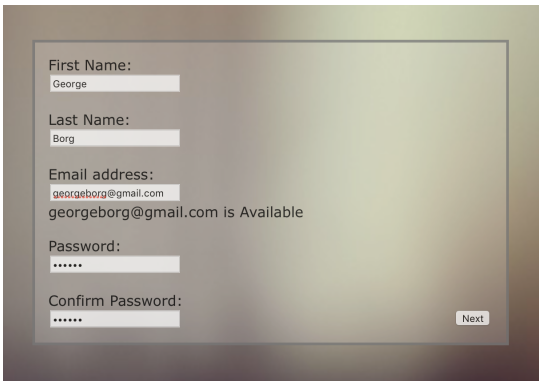

Fig 1.2: Register page

Alternatively, if a user decides to create a new account, they are asked to fill in a form with details such as name, surname, username and password. JavaScript is implemented to enforce input validation, ensuring that valid email addresses and passwords are entered. With all validation tests complete, a recommendation form is displayed, where the user may select the category of news in which they are most interested. A list of preferred categories may be selected and submitted to be handled by the system. We shall discuss the implications and uses of this explicit preference identification by the user at a later stage. The user is then redirected to the main RSS Feed page.

- Main page

The main RSS Feed page consists of a list of news articles for reading, a navigation and category section; whereby the user may adjust the articles which are displayed, and an RSS addition form, which allows the user to manually add their own feeds, over and above the default feeds.

As was mentioned, each user is assigned 14 news feeds by default, split up into their respective categories. Each individual feed is therefore assigned a category, together with a name, link (to the feed itself) and priority. The name and priority are specific to each individual user, while the link itself is not, as multiple users may access the same feeds. We shall touch on the user priority at a later stage.

- Article output

Outputting of articles from the respective feeds on screen is the most intensive process that this site handles. The feeds to be used are first extracted from the database based on a selectivity criteria (such as category,

name or priority). These links are stored in a PHP array, and are then passed on for further processing. A loop is used to cycle through each of these links, and a request is used to download the XML source code. This source code is temporarily saved, so as to undergo parsing for data extraction.

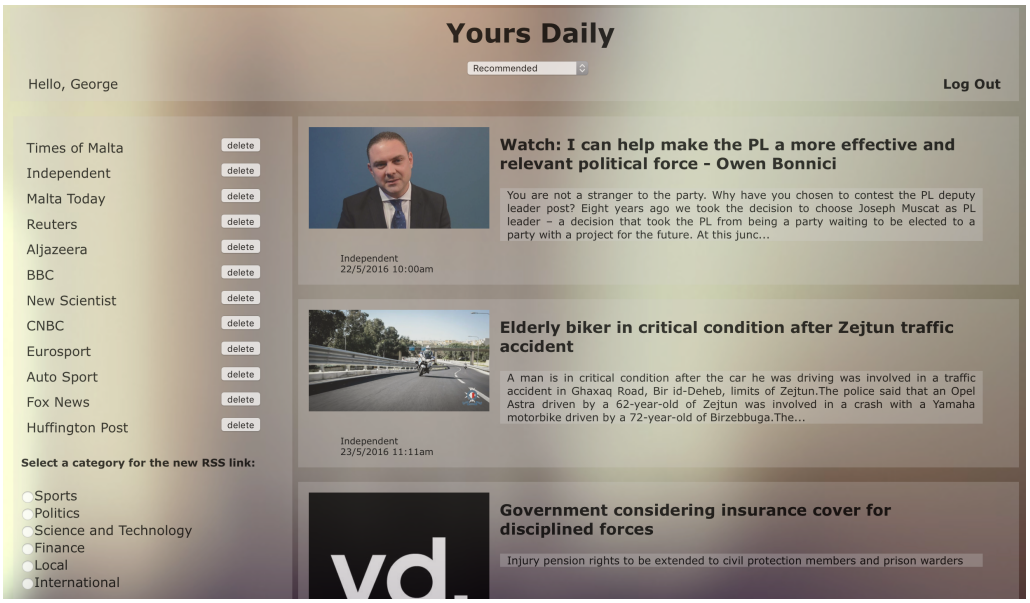

Fig 1.3: Main page with articles

Specific XML tags such as <pubDate> and <description> are sought out for their respective data to be extracted. Research has allowed us to conclude, that although no strict regulations are enforced when it comes to the presentation of an RSS Feed, most news outlets follow similar patterns, with the most common pattern being chosen for implementation in our system. Data such as title, article link, image (where applicable), description, publish date and news source are extracted and saved in variables. Certain data, such as article titles, are extracted directly from the XML tags, however others such as article images, require further parsing of content. Therefore, the extraction of image data was treated on a feed by feed basis, whereby the parsing system was hardcoded to look for image sources in certain areas for certain feeds. Once all this data is extracted, boxes are used to output the data in an orderly fashion. Placement and styling was also applied to these outputs so as to make them more appealing and easy to understand.

The above mentioned procedure was implemented for each feed, and for each article in the respective feed, resulting in a list of articles being output on screen in an orderly fashion. Whenever a new set of articles is desired to be viewed (by altering the category or visiting an individual feed), the above mentioned PHP script is called through Ajax, loading the articles dynamically to provide a better user experience.

- Priority

The adjustability and intelligence of this system resides in the manner and order in which articles are selected and output on screen. It was decided that a priority based system, ranging from 0 (lowest priority) to 1000 (highest priority), is the most suitable for a project of this type, whereby each feed and individual article are assigned a priority, and the content displayed is ordered by this assigned priority. This priority is altered according to the manner in which the user acts, predicting suitable content for them to read. Over a moderate period of time, the system is able to correctly suggest and give preference to articles which are most interesting to the user.

- Feed Priority

Each news feed is assigned a priority upon creation. By default, ordinary feeds are assigned a mid-range priority of 500, giving each feed an unbiased chance of being output. The explicit category interest, specified by the user on registration, is then used to alter this priority. If a specific category is selected as being of interest to the user, the feeds forming that priority are boosted with a 50 point boost. This boost gives these articles preference over the others, as they are most likely of more interest to the user. Manually added feeds also start at a priority of 550, as they are automatically considered to be of interest to the user.

When a category is selected to be viewed from the drop down menu, each feed forming part of that category is given a boost of 3 points, as it is assumed that the user is generally interested in feeds of such a category. Furthermore, if an article link is clicked on, redirecting the user to the article, the feed from which that article originates is given a boost of 15 points, as it is assumed that the user might find further content from this feed of interest. The article is also marked as being ‘watched’ by lowering the opacity and therefore indicating to the user that the current news article has been previously accessed.

- Scheduled Priority

Following testing and observations, it was found that a mechanism was needed to keep the articles from diverging apart from each other over time, resulting in complete preference of one or two news feeds.

A scheduled script was put in place to run every day in the early hours of the morning. This script traverses the database and decrements the priority of each feed. For this problem, it was found that a percentage decrease is more suitable than an integer decrease, as this helps with convergence. Articles with priorities between 500 and 1000 are decreased by 2% each day, resulting in a decrease of between 10 and 20 points. Meanwhile, articles with priorities between 0 and 499 are decreased by 1%, resulting in a decrease of between 1 and 5 points every day. This means that higher priority feeds are decreased at a faster rate, ensuring that, without constant user preference for a single feed, no one feed should overshadow the others. This also means that over a large period of time, if no user interaction occurs, the priorities are more or less levelled out.

- Article priorities

The final and most important piece of the priority system is the article priority. When being output on-screen, articles are ordered by priority, bringing the most relevant articles to the current user to the top, while pushing down the remaining ‘less important’ ones.

During the article output phase, each article inherits the priority of their parent feed. Therefore, if the Times of Malta feed currently has a priority of 635 then, initially, all of its articles will inherit the 635 priority. Next, the time of publish each individual article is taken into account. The priority of each article is decreased by 5% for each hour that the article is old. This is done by comparing the current hour with the hour listed in the <pubDate> tag, and adjusting the priority according. Therefore, if a Times of Malta article is 5 hours old, its priority is reduced to 490, with a further reduction of 5% for each hour that passes after that. Such a rapid decrease rate was selected due to the fast rate at which news portals publish articles, deeming day old articles practically outdated. It is useful to note that articles are available on our site until they are removed from the

host news' portal, and therefore low priority articles will always be accessible, but will be at the bottom of a page.

This inheritance and reduction therefore incorporates article date into the priority calculation, giving preference to fresh articles. Articles are then ordered by means of preference before being output on screen, with the top articles being most suitable and fresh for the current user. The above procedure allows us to create a recommendation page, which outputs the best articles from all categories, with these articles being personalised for the current user.

- Customisation

Users have a variety of controls at their disposals to adjust and perfect their user experience on the site. On the left hand side of the page, a list of currently subscribed news feeds is available. Clicking on one of these feed titles displays news articles from that specific feed only. Furthermore, feeds can be added or deleted, allowing the user to manage what news to view. Deletion of links is possible by simply clicking on the delete button near the feed title on the side. This removes the feed record from the user's database entry and prevents articles from that feed from showing any more.

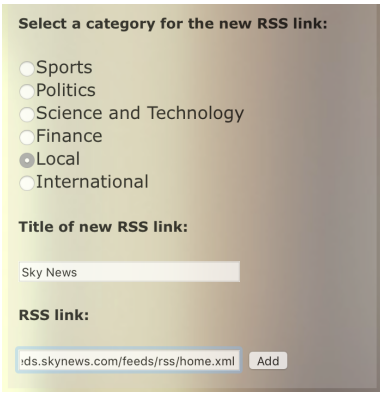

Fig 1.4: Adding link

Addition of feeds is possible by means of the addition form on the left hand side of the main page. The user may assign a feed title and a category together with a URL link to the desired RSS Feed. This feed is then included in the users profile, with an initial priority of 550 as discussed above.

- Deployment

Following testing and modifications, the site was deployed on the Microsoft Azure Web Service platform. A web application was created and the HTML, PHP, CSS and JavaScript files were uploaded to the platform. A MySQL database was also created on the platform and linked within the scripts by means of the provided access credentials. An Azure Scheduler was also created to run the mentioned priority schedule scripts. This scheduler was set to run every day at 3:30am. This was found to be a highly useful feature as it allowed

Problems Encountered

Below are a list of issues encountered throughout the assignments implementation;

Each website required some parsing to be done in order to extract the articles and any images if present. Our system is made up of 14 different news links. 4 of these feeds had considerably different XML code to the others and therefore their own parser was created. When it came to creating their own parser issues where encountered since code was differing between one article and another within the same xml feed. By trial and error the main problems were found and these were then hard coded so as to have correct articles displayed.

One of the main issues which resulted in the team creating a unique parser was that the images were found in the description tag and another being that <p> tags were opened and not closed properly and therefore confused the system, resulting in multiple articles displaying as one. With respect to the images, this issue was found in the Times of Malta XML. Within the description section an image link is included where images were available. This resulted in such images being displayed in the description section which is not the layout we were going for. The parser was then generated so that when reaching the description tag images links were extracted and dealt with first and then the description could be handled later with no issues.

In order to keep the articles in a correct order the publication date was used. When it came to 'TechNews' no publication was found in the XML code. This defeated the purpose of having articles in date order and resulted in the system not being able to sort such articles. For this reason this link was removed.

With regards to the layout of this website, the adjustments were coded using percentages which allowed the site to be scaled accordingly depending on the screen size. Although testing was done using two different screen sizes, when it came to testing on other screen sizes, issues were encountered such as overlapping of tags. Time was allocated in order to try solve this issue however no better solution was found.

With regards to the priorities used, these were also changed mid way through implementation. The values used during incrementation of priorities were found to be too large and articles were not being displayed as desired. In order to solve this, the priorities were re-evaluated and new values were set which displayed better results.

Future Work

Like in any other assignment or project there are an infinite amount of features which could be added but due to time constraints the team was restricted from implement more features. An important feature which would have been implemented was the use of storing user data to recommend better and more accurate articles. This is because the webpage recommendations currently working depends on the users' clicks on the RSS website.

Since a problem arose when the webpage was tested on other monitor screens it was found that rather than working with percentages and considering the window size, a referenced link stated that using 'Em' might have given better results [2].

Regarding design, a choice could have been available to toggle between different design templates. Different design plates could have included different background, font styles as well as article layout.

Last but not least such a website would be essential to work on all browsers if possible and although it was tested on Chrome and all the features worked well, while testing it on Safari not all features were compatible.

Conclusions

Due to the current on-the-go lifestyle, such a website is beneficial since everything can be accessed in just one site. It has been shown how users can personalise this site according to their preferences as well as interests. This implies having everything in one website makes it easier and more efficient for users to keep up to date with any news. Furthermore with various references to different sites, users may also compare news articles between different links with just one click.

In conclusion working on this assignment helped the team better understand the process of adaptivity and the work required to achieve great results which will be of great use in the future. The functionality of the proposed system has been shown, with all the specifications met. Further improvements to the site could see a more adaptive, personalised and intelligent system be developed.

References

[1] Website link: http://rssfeedejk.azurewebsites.net/

[2] Em for page scalability: https://www.boutell.com/newfaq/creating/anyresolution.html

- Image used for background: https://www.google.com.mt/search?q=blurred%20background&espv=2&biw=1366&bih=667&tbm=isch&imgil=vApzFc-6CPUJUM%253A%253BrH26NhrxsCO9mM%253Bhttp%25253A%25252F%25252Fwallpart.com%25252Fposter%25252Fblurred-image-7588-978101456&source=iu&pf=m&fir=vApzFc-6CPUJUM%253A%252CrH26NhrxsCO9mM%252C_&usg=__fDdaXZaBlOK3rVw80g_GOZrdTT8%3D&ved=0ahUKEwj87c-4iuvMAhUKHxoKHavrCVoQyjcIMw&ei=0EhAV_zFMIq-aKvXp9AF#imgrc=6dVNTtVI25-rTM%3A
- Image used as default picture for articles when no image was found: http://www.canelandcentral.com.au/-/media/retail/au/_shared/retailer-logos/yd.ashx