User Requirements Specification (URS) for RSS Hamster

Date	**:**	**16/05/2024**
Version	**:**	**2.0**
State	**:**	**Finished**
Author	**:**	**Nuno Dias**

Version history

Version	Date	Author(s)	Changes	State
1.0	04-04-2024	Nuno Dias	First Draft	Finished
2.0	16-05-2024	Nuno Dias	Expanded on original doc Isolated Use Cases to other doc Corrected grammar	Finished

Glossary

RSS	Really Simple Syndication
Algorithm	Coded Instructions that create a desired effect

Introduction

The User Requirements Specifications (URS) outlines the functional and non-functional requirements for the development of a project, an RSS Feed aggregator and viewer. These will serve to guide the project s that both the clients requirements and expectations are met.

The project itself will be a tool to allow individuals to centralised their news, entertainment and other website notifications so they don't need to visit thirty different websites to keep up with the web-o-verse.

The project will include a WinForms application, mainly for use by Admins to help them manage and moderate the content, as well as a WebApp to be deployed to the Web for any user to use. These requirements will cover both Applications in one way or another.

Functional Requirements

FR.1 Feeds

FR. 1.1. The system will allow administrators to create, view, update, and delete Feeds from the database (CRUD).

FR. 1.2. By selecting a Feed a User will be able to view its comments, ratings and information that might help him decide if he is interested in following the feed or not. These include, but are not limited to, categories, popularity and similarity to content the user enjoys (determined by an algorithm).

FR. 1.3. There will be a record of all removed feeds.

FR.2 Users

FR. 2.1. The system will allow users to register, view their profile, update information, and delete their account (CRUD).

FR. 2.2. Users will be able to follow Feeds they want to keep up with and add Feeds at their discretion.

FR. 2.3. Users will have a simple Profile they can view and edit

FR.3 Admins

FR. 3.1. Admins will be able to create and delete other admins accounts (CRUD).

FR. 3.2. Admins will be able to view and manage all Feeds and their details.

FR. 3.3. Admins will be able to view all Users and manage them.

FR. 3.4. Admins will be able to hide or remove certain feeds.

FR.4 Search and Filters

FR. 4.1. Users will be able to search all feeds

FR. 4.2. Users will be able to select a subset of Feeds based on filters such as, but not limited to, categories, algorithmic recommendations, ratings and popularity

FR.5 Algorithms

FR. 5.1. Users will be able receive a recommendation of Feeds based on their past activities

FR. 5.2. There will be multiple Algorithms to select from and therefore lists of recommendations for the user to browse through.

FR.6 Reporting and Analytics

FR. 6.1. The system will generate analytical reports for admins.

FR. 6.2. Reports will include feed popularity, overall user activity, banned users and more.

FR. 6.3. The system will support export capabilities for sharing reports in various formats (e.g. PDF, JSON).

Non-Functional Requirements

NFR.1 User Friendly

NFR.1. The system will have an intuitive and user-friendly interface to facilitate ease of use and navigation for administrators and users.
NFR.2. Features such as search filters, sorting options, and contextual help will be provided to make the experience more intuitive.

NFR.2 Scalability

NFR.1. The system architecture will be designed to accommodate future growth and scalability requirements, including increased data volume and user concurrency.
NFR.2. Modular and extensible code will be utilised to support future updates and swapping of modules with minimal effect on the rest of the code.

NFR.3 Reliability

NFR.1. Through error handling
NFR.2. Multiple data validation check at multiple points through the program

NFR.4 Security

NFR.1. The system will implement role-based access control to control sensitive information and functionalities based on user roles. This will help the App restrict unauthorised access and ensure that users will only have access to the corresponding resources.
NFR.2. The system will implement user authentication mechanisms (e.g., username/password, multi-factor authentication). This will help to verify the identity of users and add another layer of security to the system.

NFR.5 Performance

NFR.1. The system shall be capable of handling concurrent user interactions and database transactions without significant performance loss.
NFR.2. Response times for common operations (e.g., record retrieval, profile updates) should be within acceptable limits.