Introduction 1.

2. Scope

2.1Product Perspective

System Interface 2.1.1

Functions 2.1.2

2.1.3Quality Assurance

2.1.4Design and Implementation

2.2User Classes and Characteristics

2.3Operating Environment

2.4Assumptions and Dependencies

Functional Requirements3.

3.1User Account Management

3.2Movie Management

Movie Display 3.3

Search 3.4

Non-Functional Requirements

4.1User Interface

4.2Security

4.3Performance

Conclusion

SRS Document for Simple Internet Movies Database

Introduction

The Simple Internet Movies Database is a web application for movie enthusiasts. The application allows users to view information about movies, including movie title, description, release year, actors, and images. Users can also leave comments and ratings for movies. The application has an admin interface that allows the admin to add, edit, and delete movies in the database.

Scope

The scope of this document is to describe the requirements of the Simple Internet Movies Database web application. The web application will be built using Python programming language and Flask framework. SQLite will be used for database storage.

Product Perspective2.1

The Simple Internet Movies Database web application is intended for public use. The application will contain 10 movies initially, with the option to add more movies. The application will have the following features:

2.1.1 System Interface

The application will have the following user interface:

index.html - the home page that contains the main movies page, including movie pictures

movie\_info.html - a web page containing the movie information

add\_profile.html - a web page for adding new movie

alter\_movie.html - a web page for editing an existing movie

Index.html- a web page containing all the movie

Login .html –the browser write his email and password

Sign\_in\_html-to register in the website

On each movie page, there is information related to the movie (movie title, description of the movie, actors and author, year of release) and under met, there is the possibility for those who read the movie information to put a comment. The comment will be visible to all other users

.

The glossary for the movie website project includes:

|  |  |
| --- | --- |
| A database is a collection of organized data that can be accessed, managed, and updated easily. | Database |
| Browser can write his name and write a review and rating | review |
| Is a document that outlines the features, functionalities, and specifications of a software system to be developed.. | SRS- |
| Is a lightweight and self-contained relational database management system (RDBMS) that allows users to store, manage, and retrieve structured data in a file-based database.. | SQLite |
| Is a lightweight and flexible Python web framework that allows developers to build web applications quickly and easily.. | Flask |
| The 10 movies image appear, and the button that leads to the add movie page appears | Home page admin |
| The 10 movie image appear | Home page (user) |
| A page showing a specific movie, and the admin can delete or change the movie information | Movie info(admin) |
| A page showing a specific movie and his information | Movie info(user) |
| In order to enter the site and watch movies, the browser must write the email and password | Sign in |
| In order for the browser to use the movie site, he must have an account on the site, and if he does not have it, he must fill in his details on log in page. | Log in |

2.1.2Functions

The web application will have the following functions:

View movie information

Add new movie

Edit existing movie

Delete existing movie

Leave comments and ratings for movies

2.1.3Quality Assurance

The web application will be tested thoroughly to ensure that it meets the following quality standards:

The web application should be easy to use and navigate

The web application should be visually appealing and responsive

The web application should be bug-free and work efficiently

The web application should be secure and protect user data

The web application should be scalable to accommodate future growth

Design and Implementation 2.1.4

The web application will be written in Python programming language using Flask framework. SQLite will be used for database storage. The application will be designed with a responsive user interface that is easy to navigate and visually appealing.

2.2 User Classes and Characteristics

The web application is intended for movie enthusiasts who want to view information about movies and leave comments and ratings for movies.

Operating Environment 2.3

The web application will be compatible with standard computers (PCs) and run on web browsers. The web application will require an internet connection to download the Python file.

Assumptions and Dependencies 2.4

The web application assumes that users have basic computer skills and are familiar with using web browsers. The web application depends on Python programming language, Flask framework, and SQLite for database storage.

Functional Requirements:

User Account Management:

The system should allow users to create an account by entering their email and password.

The system should allow users to log in to their account.

The system should allow users to view their account information.

Movie Management:

The system should allow the admin to add a new movie to the site.

The system should allow the admin to delete a specific movie from the site.

The system should allow the admin to edit the information of a specific movie on the site.

Movie Display:

The system should display 10 movie images on the main page.

The system should display movie information (title, description, actors, release year) on a separate page.

The system should display comments written by users on the movie information page.

Search:

The system should allow users to search for a specific movie by entering the movie title in a search bar.

Non-Functional Requirements:

User Interface:

The user interface should be user-friendly and easy to use.

The user interface should be visually appealing and attractive.

The user interface should be responsive and adaptable to different screen sizes.

Security:

The system should have secure user account management.

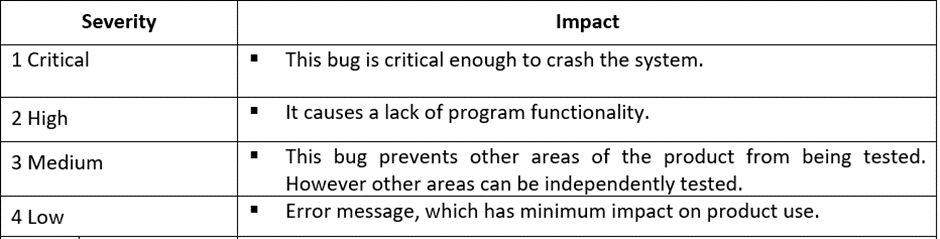
The system should protect user information from unauthorized access.

The system should protect user information from data loss.

Performance:

The system should load quickly and respond to user actions without delay.

The system should be able to handle a large number of users accessing the site simultaneously



Conclusion:

The Simple Internet Movies DataBase is a website designed to allow users to view information about movies and leave comments. The website contains 10 movies, and users can add, delete or edit information for each movie. The website also allows users to search for specific movies. The user interface is designed to be easy to use and visually appealing. Security and performance are important considerations for the site, with measures in place to protect user information and ensure fast loading times

Authors and acknowledgment:-

Marjan Ikteelat

-License

The software is free under GPL (General Public License) The introduction.