Software Requirements Specification

for

Shenron

Version 1.1 approved

Prepared by

Amr Khalid 202201502

Abdalrahman Ashraf 202202066

Omar Yasser 202201589

Yusuf Tamer 202201929

Zewail City

9/11/2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
First increment	1/12/2024	Adjust to new changes based on implementation constraints.	1.1

1. Introduction

1.1 Purpose

Shenron is a movie, series and anime database. This is the first version of the product. The system is a platform for users to discover their favorite movies and series as well as new ones and to discuss different topics with other users in forums. This document describes the initial functional and non-functional requirements for the system and provides some graphical descriptions of the system, such as use case diagrams.

1.2 Document Conventions

This document uses a font size of 32 for titles, 18 for main headings, 14 for subheadings, and 12 for normal text. The font used is *Times New Roman.* Keywords are written in italics. No further special formatting was applied in writing this document. Finally, words like movies, programs, TV series, and anime are used interchangeably.

1.3 Intended Audience and Reading Suggestions

This document's primary audiences are developers and the product owner. Developers are the team that will be implementing the project's software. The developers can refer to this document for input on the functional and non-functional requirements. In addition, the document provides information on the system interfaces and the operating environment. Furthermore, this document will be used to validate the project's functionality with the product owner as the product owner has the ultimate authority for which features to be included in the project, and can reject or accept the final product.

For developers, it is suggested to read the following sections: project scope, user classes and characteristics, system features, external interfaces, operating environment, design and implementation, and other nonfunctional requirements in this order of listing. For the product owner, it is suggested to refer to the glossary at the end of the document for technical terms used in the document.

1.4 Project Scope

Shenron is an online movie, series and anime database similar to IMDb and MyAnimeList. *Shenron* will incorporate new features including a Forums page where users can discuss their thoughts about what they have watched, and a new rating system using surveys to accurately and objectively measure a program's rating. As well, *Shenron* will allow users to create lists of their favourite entertainment programs. *Shenron* is envisioned to be the leading website for providing TV program-related information while maintaining an accountable and inclusive community.

1.5 References

User interfaces: Shenron GUI

2. Overall Description

2.1 Product Perspective

Shenron is a new product designed to enhance user experience in finding and tracking their favourite movies/TV shows. Shenron also focuses on delivering personalized recommendations. Third party software will be used for authentication and data retrieval. Shenron's major components are:

- **Frontend**: The user interface for interacting with the system.
- **Backend**: The server-side processing and business logic.
- **Database**: The storage for all persistent data.

2.2 Product Features

- Allow users to explore movies and series to their liking
- Add movies and series to their list
- get recommendations based on their usage
- users can post their opinions and involve in open discussion with each other on forums
- users can rate different movies and series
- report posts that are found offensive, inappropriate, or discriminatory.

2.3 User Classes and Characteristics

Admins: personnels with the highest level of technical background. They will manage and maintain the system and will most likely have the highest level of accessibility.

Forums Moderators: Personnels that will monitor the forums and remove any posts that are found to be inappropriate or discriminatory.

Users: personnels the web application is developed to satisfy their needs. users will use the software to view entertainment programs and related information, review them, and post their opinions through forums.

2.4 Operating Environment

Shenron is designed to be compatible with the latest versions of popular web browsers such as: Google Chrome (version 95 and above), Microsoft Edge (version 95 and above) and Safari (version 14 and above).

For Hosting requirements, Shenron's backend will run on a server that supports Django 5.1 or above as the backend framework, HTML, CSS, and javascript for the frontend and MySQL 8.0 or above for database management and a minimum of 4GB RAM and 2-CPU core to handle high traffic. For hard disk memory, a 10 TB space is required to store the database and to store all the static files, such as images.

2.5 Design and Implementation Constraints

Relying on external API Availability and Consistency as their circumstances may change. Their rates may change or increase with high usage and they may be permanently or temporarily down. The response time of the server is currently very large due to the use of external storage and slow internet speed.

2.6 Assumptions and Dependencies

To update the programs the website depends on a third party API

3. System Features

3.1 Search

3.1.1 Description and Priority

The search allows users to quickly find what they want using the movie's title.

3.1.2 Stimulus/Response Sequences

Stimulus 1:
Action: The user enters a search term in the search bar
System Response: the system displays the movies based on the entering
Stimulus 2:
Action: The user clicks on the movie/tv-show
System Response:The system redirects the user to a new page with more information about the movie/tv-show

3.1.3 Functional Requirements

REQ-1: The search bar accepts a user input as a text and returns relevant results based on the text

REQ-2: The search result will include basic details like title and poster, and will allow the user to click on the movie to be redirected to another page to get more information.

3.2 Add to list

3.2.1 Description and Priority

The add to list function allows users to add a movie/series/anime to their favourite list. The feature is of medium priority

3.2.2 Stimulus/Response Sequences

Stimulus 1:
Action: The user enters a search term in the search bar
System Response: the system displays the movies based on the entering
Stimulus 2:
Action: The user opens the page for a movie/anime/series
System Response: The System will display the page of said program

Stimulus 3:
Action: The user clicks on the Add to Favourites button
System Response: The System will add the program to his favourites list

3.2.3 Functional Requirements

Pre-Condition: The user is logged in

REQ-1: The clicking of the add to favourites button must have the ability to add the program that the user chose to his own favorites list.

3.3 View List

3.3.1 Description and Priority

This feature allows the user to view the items he added to their Favourites list. This feature is of Medium Priority.

3.3.2 Stimulus/Response Sequences

Stimulus 1:
Action: the user selects Favourites from the navigation bar
System Response: the system redirects the user to the selected list

3.3.3 Functional Requirements

Pre-Condition: The user is logged in

REQ-1: The system shall allow the user to view their entertainment programs list when they click on the favourites button from the navigation bar.

3.4 Remove from list

3.4.1 Description and Priority

The remove from list functionality allows the user to remove a program they have previously added to their favourites list using the add to favourites functionality.

3.4.2 Stimulus/Response Sequences

Stimulus 1:
Action: the user selects Favourites from the navigation bar
System Response: the system redirects the user to the selected list

Stimulus 2:
Action: the user hovers over a specific movie
System Response: details related to that movie/program appear with a delete button

Stimulus 3:
Action: the user presses the delete button
System Response: the system removes that movie from the user's favourites list

3.4.3 Functional Requirements

Pre-Condition: The user is logged in

REQ-1: Hovering over a movie in the user's favourites list shall provide the user with more information on that particular movie.

REQ-2: On hovering over a movie in the user's favourites list, a delete button shall pop up.
REQ-3: On pressing the delete button, the system shall remove that program from the user's favourites list.

3.5 Create Forum

3.5.1 Description and Priority

This functionality allows the user to create a forum by providing a title for it.

3.5.2 Stimulus/Response Sequences

Stimulus 1:

Action: the user selects Forums from the navigation bar
System Response: the system redirects the user to the selected page

Stimulus 2:

Action: the user types a title representing the topic that the forum will include discussions about.
System Response: the text written inside the input field are visible to the user

Stimulus 3:

Action: the user presses the Create Topic button.
System Response: the system adds said topic to the Forum topics and displays the updated list of forums with the username of the creator next to the new topic.

3.5.3 Functional Requirements

Pre-Condition: The user is logged in
REQ-1: The user shall be redirected to the forums page on clicking the forums button from the navigation bar
REQ-2: The user shall be able to type a title inside the input field found in the forums page
REQ-3: On clicking the Create Topic button the system shall create a new forum with the title provided by the user.
REQ-4: The system shall display the new list of forums after a new forum has been created.
REQ-5: The system shall display the username of the creator of each forum next to it in the forums list

3.6 Post on Forum

3.6.1 Description and Priority

This feature allows the user to add a post on a specific forum page.

3.6.2 Stimulus/Response Sequences

Stimulus 1:

Action: the user selects Forums from the navigation bar
System Response: the system redirects the user to the selected page

Stimulus 2:
Action: the user selects a forum from the list of forums on the forums page.
System Response: the system displays a new page of the forum along with all posts that were already posted on it.Also, an input field to accept user input.

Stimulus 3
Action: the user provides a title and content in the input field
System Response: the system displays what the user has written in the input field

Stimulus 4:
Action: the user clicks on the Create Post button
System Response: The system displays the updated forum page with the newly added post

3.6.3 Functional Requirements

Pre-Condition: The user is logged in and has opened the forums page.

REQ-1: The system shall allow the user to make a post on the forums page
REQ-2: The system shall display each post creator's username next to the post they have created

3.7 Remove a post

3.7.1 Description and Priority

This feature allows the user to remove a post they have created before.

3.7.2 Stimulus/Response Sequences

Stimulus 1:
Action: the user clicks on the post they have created from the corresponding forum page.
System Response: The system displays the details of that post in a new page with a delete button.

Stimulus 2:
Action: The user clicks on the Delete Post button.
System Response: The system removes the post from the corresponding forum and redirects the user to the main forums page.

3.7.3 Functional Requirements

Pre-Condition: The user is logged in and is at a forum page they had previously created a post on.

REQ-1: The system shall display the details of a post on clicking a post title inside a forum's page.

REQ-2: The system shall display a delete button in the post details page if the user is the creator of that post.
REQ-3: The system shall delete the post on clicking the Delete Post button.
REQ-4: The system shall redirect the user to the main forums page after deleting a post

3.8 Create a comment

3.8.1 Description and Priority

This feature allows the user to comment on a post.

3.8.2 Stimulus/Response Sequences

Stimulus 1:
Action: the user clicks on a post inside a forum page.
System Response: The system displays the details of that post in a new page with a text input field to accept user input.
Stimulus 2:
Action: The user writes content for the comment they want to create.
System Response: The system displays what the user has written inside the input field.
Stimulus 3:
Action: The user clicks on the Create Comment button.
System Response: The system displays the updated post page with the newly created comment

3.8.3 Functional Requirements

Pre-Condition: The user is logged in and is at a forum page.
REQ-1: The system shall display a text input field inside a post details page to accept user input.
REQ-3: The system shall create a post on clicking the Create Comment button.
REQ-4: The system shall refresh the page with newly created comment displayed.
REQ-4: The system shall display the comment creator's username next to each comment

3.9 Report a Post

3.9.1 Description and Priority

This feature allows the user to report a post on the forum's page of a specific program. This feature is of high priority.

3.9.2 Stimulus/Response Sequences

Stimulus 1:

Action: The user clicks on the report button next to the designated post to be reported.
System Response: The system displays a message to ask the user if they are sure they want to report this post.

Stimulus 2:
Action: The user clicks the confirm button.
System Response: The system creates a report of the post with the appropriate information and sends it to the moderator of the forums. In addition, the system displays a success/failure message to the user.

3.9.3 Functional Requirements

Pre-Condition: The user is logged in and has opened a forum page of some program.
REQ-1: The system shall allow the user to report a post on a forum's page.
REQ-2: The system shall ask the user for confirmation before executing the reporting process.
REQ-3: The system shall display the status of the reporting process to the user after confirmation.

3.10 Edit a post

3.10.1 Description and Priority

This feature allows the user to edit a post of their own that they have already composed before. This feature is of medium priority.

3.10.2 Stimulus/Response Sequences

Stimulus 1:
Action: The user clicks on the edit button that is only visible on their own posts.
System Response: The system displays a text-input field for the user.

Stimulus 2:
Action: The user edits their post and clicks on the repost button.
System Response: The system displays the updated forum's page with the edited user post.

3.10.3 Functional Requirements

Pre-Condition: The user is logged in and has opened a forum page of some program that they have already posted on before.
REQ-1: The system shall allow the user to edit their own posts.
REQ-2: The system shall display the updated forum's page with the user-edited post after clicking on the repost button.

3.11 Review

3.11.1 Description and Priority

This feature allows the user to review any program that they have watched using the survey rating system. The feature is of Medium Priority.

3.11.2 Stimulus/Response Sequences

Stimulus 1:
Action: the user selects a movie/anime/series that they have watched
System Response: the system redirects them to the page of that program
Stimulus 2:
Action: the user clicks on the rate button
System Response: the system prompts a survey for the user to fill out
Stimulus 3:
Action: the user submits the rating
System Response: the system prompts a message to the user about the success/failure of the operation

3.11.3 Functional Requirements

Pre-Condition: The user is logged in, The movie/series/anime is in the user's List

REQ-1: The System must open a survey after the user clicks on the Rate button
REQ-2: The System will add the user's rating to the Program's page
REQ-3: The System will measure the updated overall rating in accordance with the new rating
REQ-4: The System will prompt a message to the user about the success/failure of the rating

3.12 Edit Account Information

3.12.1 Description and Priority

This feature allows the user to edit their account information like their username, password, profile picture, etc… . The feature is of High Priority

3.12.2 Stimulus/Response Sequences

Stimulus 1:
Action: the user selects their profile page

System Response: the system redirects them to the page of their profile

Stimulus 2:

Action: the user clicks on the Edit Profile button

System Response: the system prompts an edit menu for the user to choose what they want to change

Stimulus 3:

Action: the user selects one of the options to be edited

System Response: Depends based on the option you chose

Stimulus 4:

Action: the user submits the object edited

System Response: the system prompts a message to the user about the success/failure of the operation

3.12.3 Functional Requirements

Pre-Condition: The user is logged in.

REQ-1: The System must open a list of possible edits after the user clicks on the Edit Profile button

REQ-2: The System will change the current option to the new edited option according to the user's choice

REQ-3: The System will prompt a message to the user about the success/failure of the rating

3.13 View movie details

3.13.1 Description and Priority

This functionality allows the user to view information about a specific movie like: release date, title, revenue, budget, runtime, status, and more.

3.13.2 Stimulus/Response Sequences

Stimulus 1:

Action: The user clicks on a movie/tv-show

System Response: The system redirects the user to a new page with more information about the movie/tv-show

3.13.3 Functional Requirements

REQ-1: The system shall provide the user with the ability to view more details about a movie by clicking on it.

REQ-2: The system should include the movie's poster and backdrop images.

3.14 Movies Page

3.14.1 Description and Priority

This functionality allows the user to view a page of multiple movies.

3.14.2 Stimulus/Response Sequences

Stimulus 1:

Action: The user clicks on the View our Website button on the home page
System Response: The system redirects the user to a new page with a list of movies that they can view.

3.14.3 Functional Requirements

REQ-1: The system shall provide the user with the ability to view a page with multiple movies that can be further viewed with more details on a new page.

REQ-2: The system shall allow the user to access this page through the navigation bar

REQ-3: The system shall display the basic information, such as title duration, poster , and rating for each movie

REQ-4: The system shall display a button underneath each movie poster that when clicked redirects the user to a page with details on that particular movie

4. External Interface Requirements

4.1 User Interfaces

Refer to the references in 1.5 for the GUI.

- **Home page:** includes the home page displaying the website's logo and a button to access the full website.
- **Movies Page:** displays all the movies with their duration and rating in card-like containers with buttons leading to each Movie's details.
- **Movie details:** displays the movie's poster and cover images, a link to the movie trailer, if any, and related details, such as title, runtime, release date, genre, description, rating, etc….
- **Favourites:** The programs the user has added to their favourites list.
- **Forums**: A page containing all the forums.
- **Forum details:** A page dedicated to a single forum displaying its title and all the posts associated with it. Also, a user can add a post on a forum through the page dedicated to it.
- **Post details:** A page displaying the comments and the ability to add comments inside of a post.
- **Login page:** Allows users to access the website.
- **Signup page:** Allow users to create accounts.
- **Search page:** A page that allows users to view the results of their submitted query.

4.2 Hardware Interfaces

- Supported devices: The software supports a variety of devices like: desktops, laptops, tablets, and mobile devices.

- Hardware components: back-end VPS with, initially, a minimum of 4GB RAM and 2 CPU cores to handle requests and implement the software's logic and a database VPS with, initially, a minimum of 10TB for data storage and retrieval.

4.3 Software Interfaces

- **operating systems:** The software is designed to be a web app that supports a variety of operating systems, such as Windows, Linux, and macOS.

- **Frameworks**: Django 5.1 as a back-end REST framework and HTML, CSS, and Javascript as a front-end framework.

- **Third-party APIs:** TMDB API V3 for requesting updates on programs daily, and Google Oauth 2.0 as an authorization framework.

- **Database:** MySQL 8.0 will be used as the DBMS.

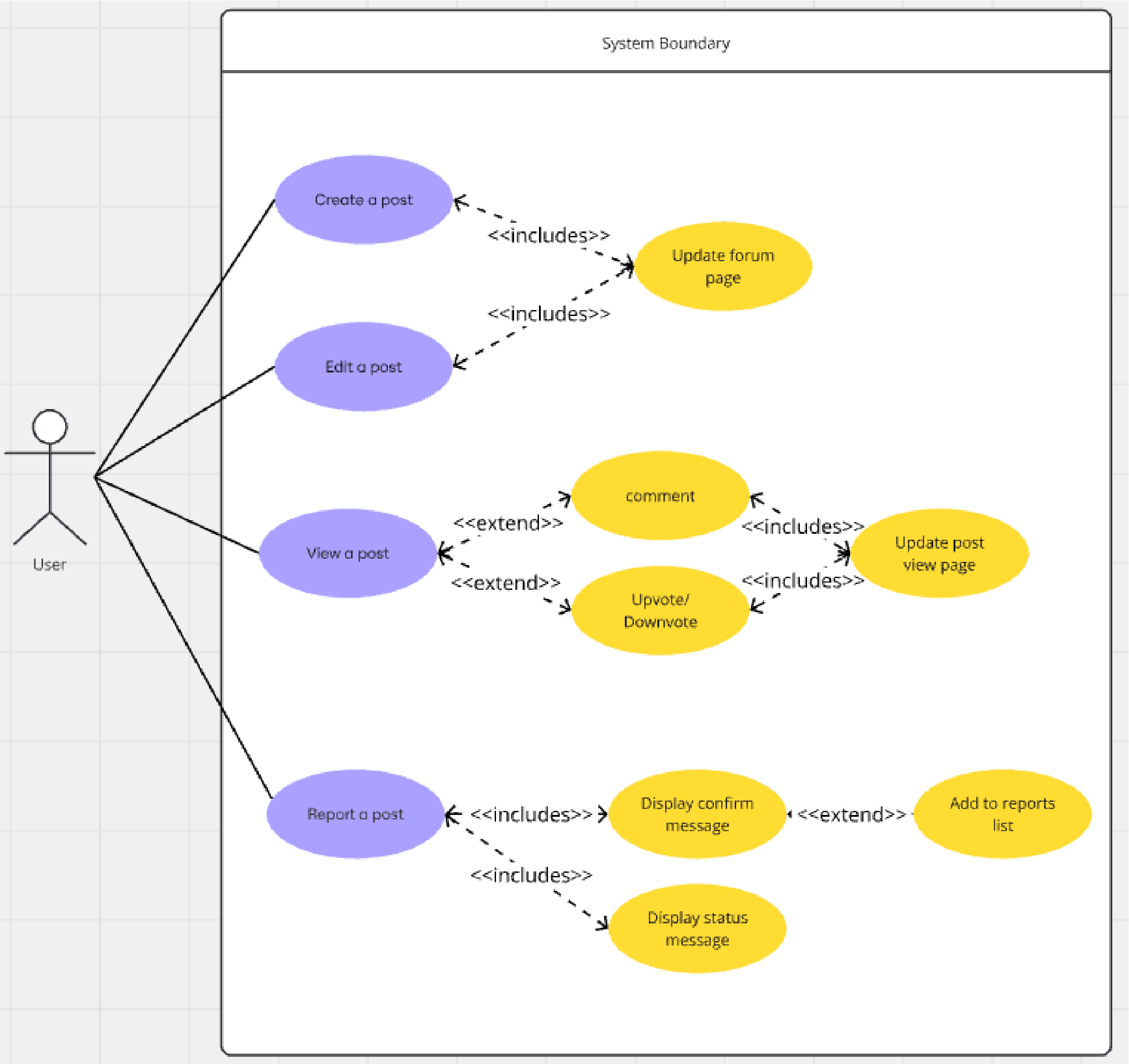
4.4 Communications Interfaces

All requests and their responses will use HTTP, to update files stored in the database SFTP will be used, and for server configurations and updates SSH will be used. Due to the use of Django RESTful API data is expected to be transferred in JSON format. Emails will be sent in case of account registration or password resetting.

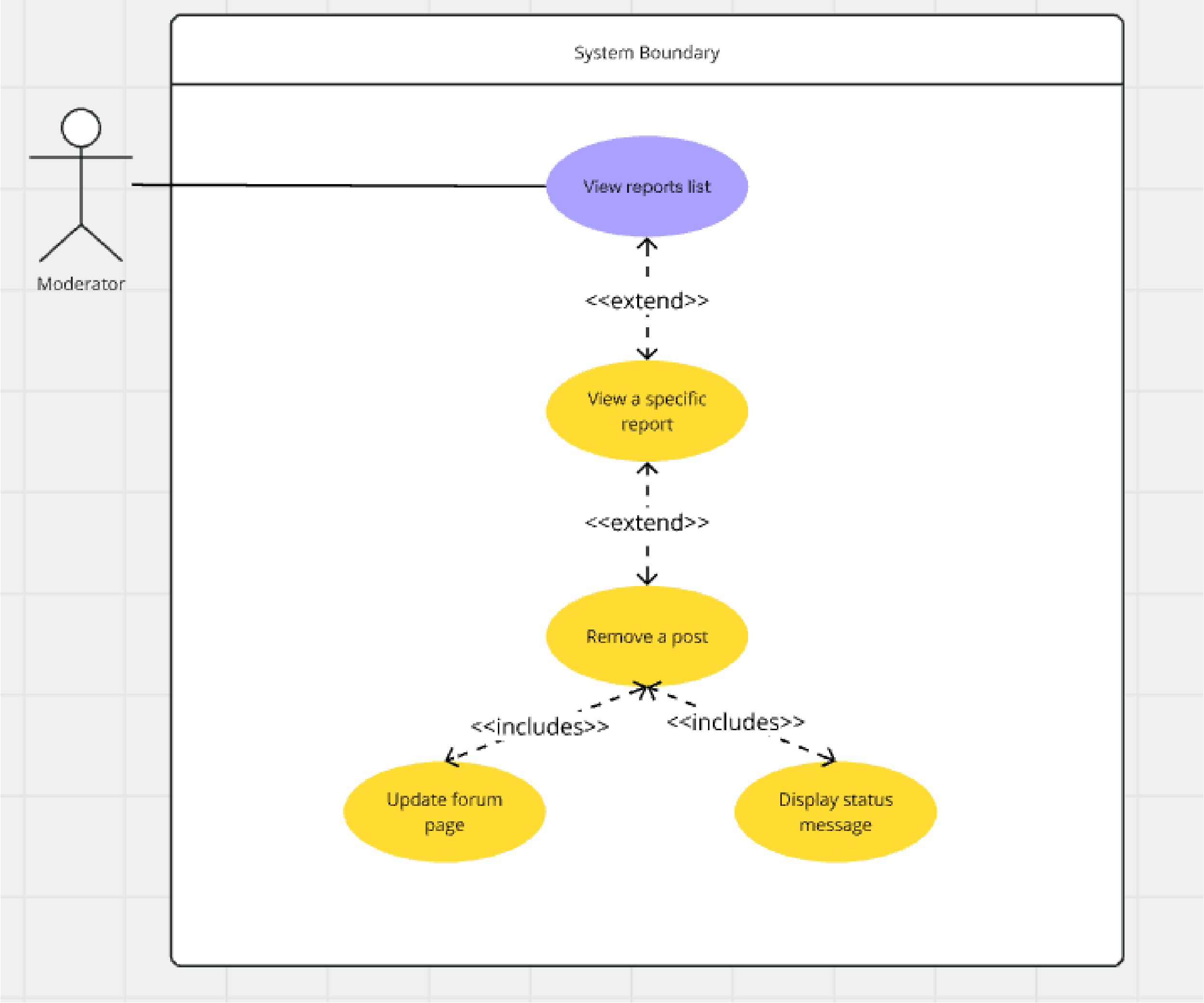
5. System models

5.1 Use Case Diagrams

User's forum functionalities use case diagram.



Moderator's remove post use case diagram



6. Nonfunctional Requirements

6.1 Performance Requirements

The system should respond to user requests within 5 seconds.

6.2 Software Quality Attributes

The system should be compatible with different operating systems to provide accessibility to a wide range of users.

6.3 Usability Requirements

The system's interface should be user-friendly, intuitive to use, and easy to navigate.

7. Other Requirements

The data collection methods must comply with all local data privacy acts/regulations.

Appendix A: Glossary

Web Browser: A software application used to access and view websites. Examples include Google Chrome, Mozilla Firefox, and Safari.

Django: A Python backend web framework used in developing web applications.

React: A JavaScript library for building user interfaces.

MySQL: A relational database management system that is based on Structured Query Language (SQL).

RAM (Random Access Memory): A type of computer memory that can be accessed randomly. It is used to store information related to actively running programs.

CPU (Central Processing Unit): The primary component of a computer that performs most of the processing inside a computer.

API (Application Programming Interface): A tool that enables two different software components to communicate with each other without having to understand how each component implements its functions.

VPS (Virtual Private Server): Allocated set of computing resources that acts, virtually, as a server allowing for applications deployment.

REST (Representational State Transfer): A software architecture that imposes some conditions on how an API should work so that it enables high-performance and reliable communication between the system's software components.

TMDB (The Movie Database) API: A third-party software that allows for retrieving data related to movies, actors, etc….

HTTP (Hypertext Transfer Protocol): A protocol that defines the communication and data exchange on the web between web-browsers and web-servers.

SFTP (Secure File Transfer Protocol): A network protocol that provides secure file access, transfer, and management functionalities over any reliable data stream.

SSH (Secure Shell): A cryptographic network protocol for operating network services securely over an unsecured network.

JSON (JavaScript Object Notation): A lightweight data-interchange format that is easy for humans to read and write, and easy for machines to parse and generate.