Tribhuwan University

Institute of Engineering

Thapathali Campus

Subject: Web Application Programming

LAB #3

Lab Report on: Server Side Programming and Django

Submitted By

Name: Srijal Basnet

Roll No: THA080BCT044

Submitted To

Department Of Electronics and Computer Engineering

5th February 2026

OBJECTIVES

To understand the concept and workflow of server-side programming.

To learn the fundamental structure and architecture of the Django framework.

To build a basic CRUD web application using Django.

THEORY

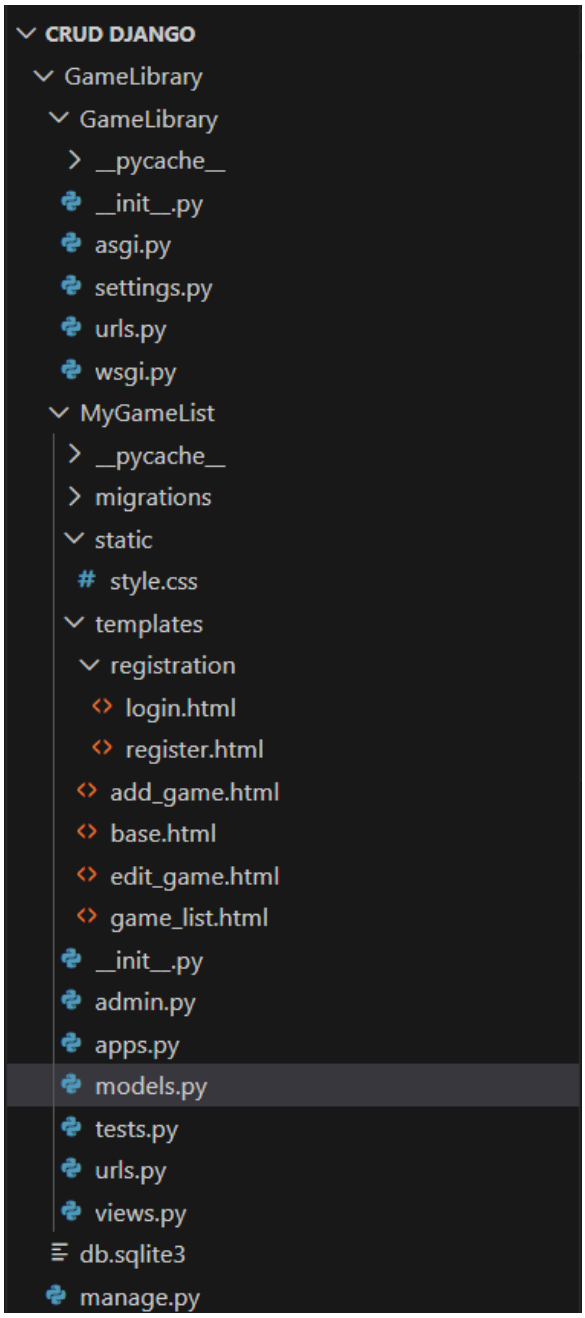
Server-side programming refers to backend development where application logic runs on a web server instead of the user's browser. When a user sends a request through HTTP methods such as GET or POST, the server processes the request, interacts with the database if required, and sends back a response like HTML or JSON. This allows web applications to generate dynamic content based on user input or stored data.

Server-side systems also provide improved security because sensitive logic and database operations remain hidden from the client side. Backend frameworks handle data processing, authentication, and CRUD operations, ensuring structured data management and persistence.

Django is a high-level Python web framework designed to simplify web development by providing built-in tools for routing, database management, authentication, and templating. It follows a structured architecture that separates data models, business logic, and presentation layers, allowing developers to build scalable applications quickly.

In this lab project, I created a simple CRUD Django application called 'My Game List'. The application allows users to add, edit, view, and delete games. Each game entry includes fields such as rating, review, and play status. The application integrates with the RAWG API to retrieve game database information, demonstrating how external APIs can be used alongside Django to enrich application data while maintaining a local database.

Project Folder Structure

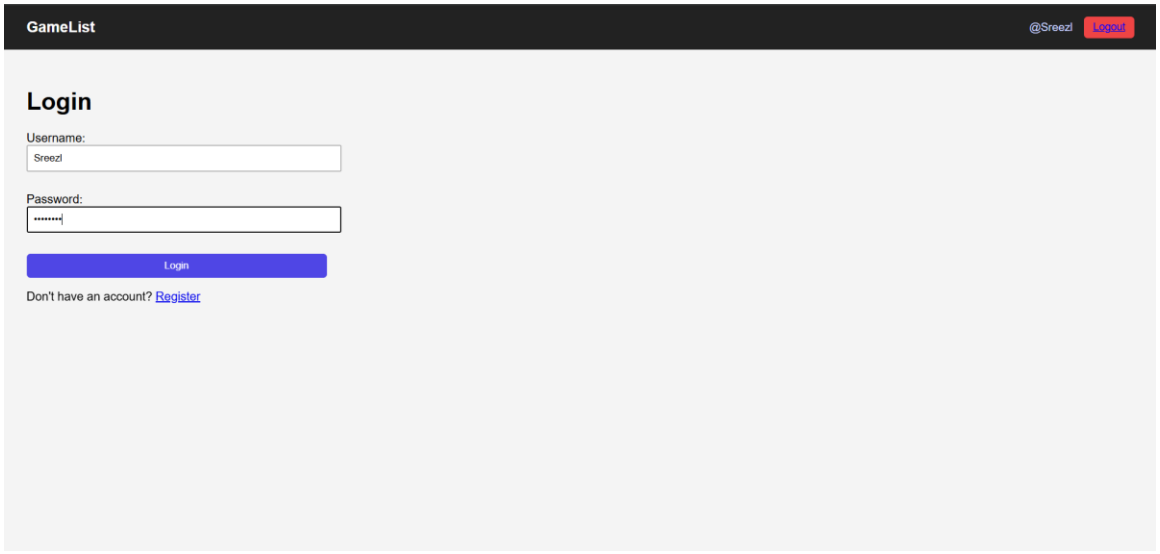


SOURCE CODE

https://github.com/Srijal-Basnet/My-Game-List-App.git

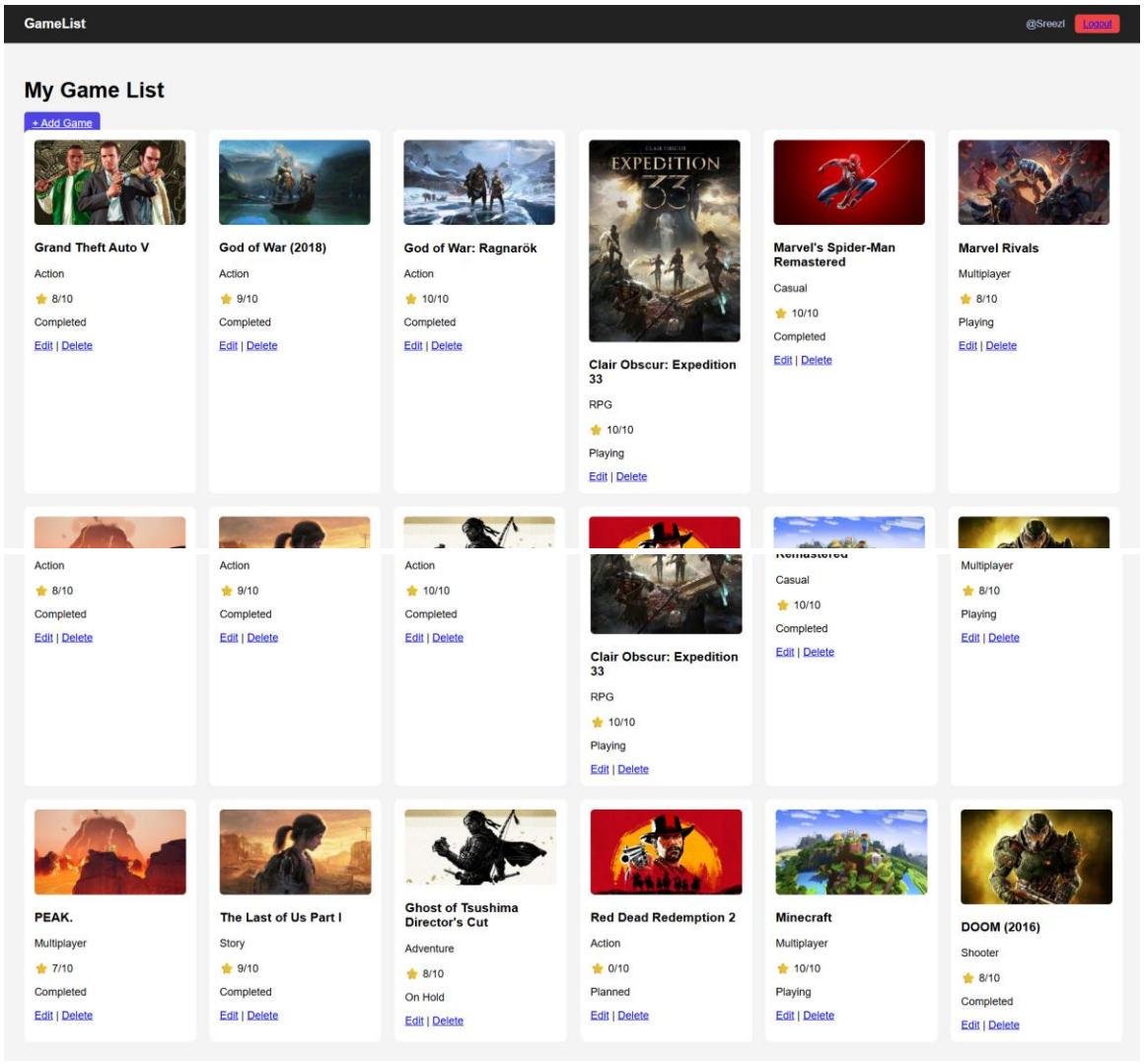
OUTPUT

1. Login page

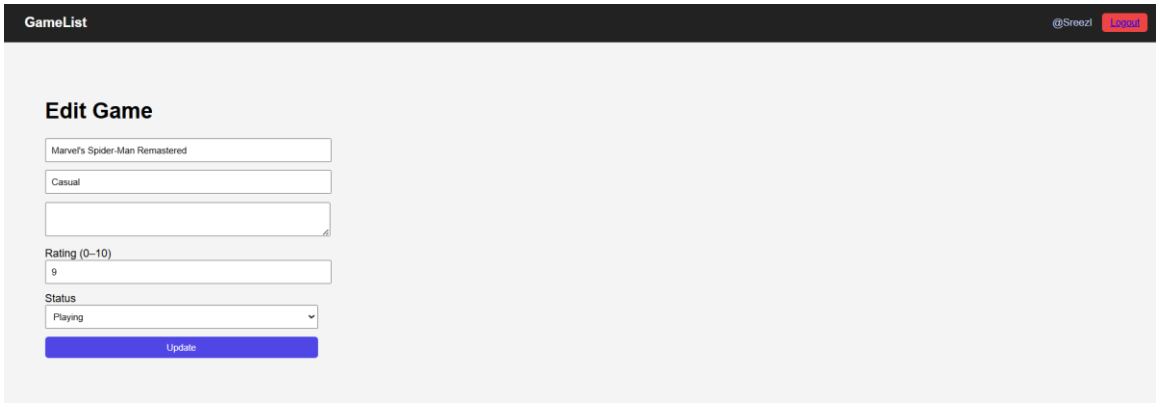


2. Register page

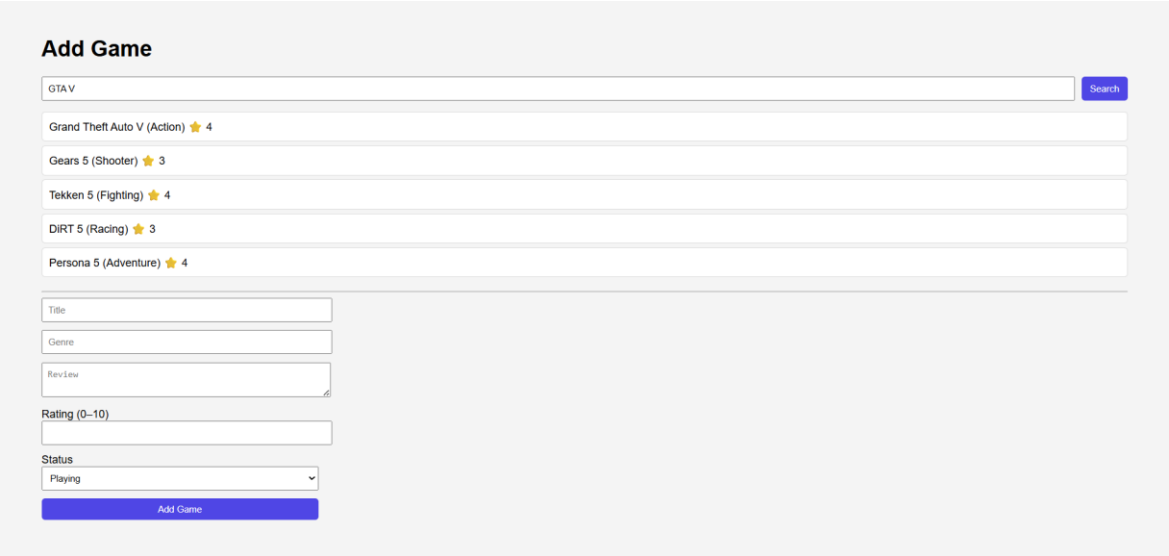
3. Home Page/ Game list



4. Edit Game Page



5. Add Game Page



DISCUSSION AND CONCLUSION

The development of this app demonstrated how server-side programming enables dynamic and interactive web applications. Using Django, the application successfully implemented CRUD functionality for managing game entries while maintaining a clear separation between backend logic and frontend templates. The integration of the RAWG API showed how external services can be used to fetch structured data and enhance user experience without manually maintaining a large dataset.

Through this lab, key concepts such as request-response cycles, database interaction using Django ORM, and API integration were explored. Django's built-in features simplified authentication, routing, and data handling, making development faster and more organized. Overall, the project demonstrates that Django provides a reliable and efficient framework for building secure server-side applications with clean architecture and scalable design.