

Test Cases for Movie Recommendation System

September 29, 2024

Introduction

This document provides a comprehensive list of 100 test cases designed to verify the functionality of a movie recommendation system built using Flask, natural language processing models, and various data processing tools.

Test Cases

1. User Interface Tests (20)

- Home Page Loading
- Search Bar Presence on Home Page
- Search Bar Functionality for Valid Input
- Search Bar Functionality for Invalid Input
- Movie Suggestions Display on Home Page
- Movie Details Page Loading
- Movie Poster Display on Details Page
- Movie Overview Display on Details Page
- User Navigation to Movie Details from Suggestions
- Breadcrumb Navigation between Pages
- Form Field Presence in Movie Submission Form
- Submit Movie Form with Valid Data
- Submit Movie Form with Invalid Data
- Empty Form Submission Handling
- Button Clickability (e.g., Submit, Recommend)
- Input Field Placeholder Text
- Dynamic Loading of Recommendations
- Scroll Behavior on Recommendation Page
- Form Validation Errors Display
- Responsiveness of Website on Mobile Devices

2. Recommendation System Logic Tests (20)

- TF-IDF Matrix Generation for Valid Data
- TF-IDF Matrix Generation for Missing Overviews
- Recommendation Generation for Valid Movie Title
- No Recommendation Generation for Invalid Title
- Genre-Specific Recommendations for Action Movies
- Genre-Specific Recommendations for Romance Movies
- Similarity-Based Recommendations for Popular Movies
- Popularity-Based Recommendation Prioritization
- Correct Recommendations for Multiple Genres
- Recommendation for Highly Rated Movies
- Movie Title Case Insensitivity in Recommendations
- Duplicate Movie Title Handling in Recommendation
- Correct Recommendation Ranking
- Handling of Empty Metadata in Recommendations
- Recommendations for Recent Releases
- Recommendations for Older Movies
- Edge Case: Recommendation for Obscure Movies
- Ensuring Diverse Recommendations for Same Genre
- Recommendation System Response Time for Large Dataset
- Invalid Recommendation Generation Attempt Handling

3. Sentiment Analysis and Reviews (15)

- Positive Review Classification for Valid Input
- Negative Review Classification for Valid Input
- Sentiment Classification for Neutral Reviews
- Empty Review Input Handling
- Review Submission Form Validation
- Multiple Reviews Sentiment Classification
- Review Analysis with Special Characters
- Handling of Extremely Long Reviews
- Reviews Containing Non-English Characters
- Review Sentiment Analysis Response Time
- Handling of Mixed Positive/Negative Reviews
- Review Sentiment Accuracy for Popular Movies
- Review Sentiment Analysis for Short Texts
- Invalid Input (Non-Text) Handling in Sentiment Model
- Multiple Language Review Handling in Sentiment Model

4. Form Validation and Input Handling (10)

- Empty Input for Search Bar Handling
- Invalid Input for Search Bar Handling
- Empty Fields in Movie Submission Form
- Invalid Email Input in Movie Submission Form
- Invalid Data Types in Movie Submission Form
- Malformed Data in Recommendation Request
- Handling of Special Characters in Movie Title Input
- Missing Data in Recommendation Request (e.g., Missing Genres)
- Form Reset Functionality for Movie Submission
- Validation Error Messages for Each Invalid Input Type

5. Error Handling and Logging (10)

- Error Logging for Missing Data in Movie Details
- Error Logging for Invalid Model Inputs
- Logging of Invalid URL Access Attempts
- Logging of Errors in Recommendation Generation
- Logging of Errors in TF-IDF Matrix Generation
- Logging of API Request Failures (e.g., IMDB)
- Error Handling for Missing Files (e.g., Pickle Models)
- Handling of Unexpected Input Types (e.g., Non-String Titles)
- Handling of Invalid JSON Requests
- Graceful Recovery from Flask Server Errors

6. External Data Integration Tests (10)

- IMDB Data Retrieval for Valid Movie ID
- Handling of Invalid Movie ID in IMDB Request
- Parsing of IMDB Data for Valid Movie
- Parsing of IMDB Data for Missing Data Fields
- Successful Handling of Missing IMDB Data (e.g., No Reviews)
- API Request Failure Handling for IMDB
- Handling of Timeouts During IMDB Data Retrieval
- Logging of Errors During API Request
- Error Recovery on Failed API Request Retry
- Handling of Invalid Responses from External APIs

7. Performance and Scalability Tests (10)

- Performance Under Heavy Load of Requests
- Response Time for Large Movie Dataset
- Latency in Recommendation Generation for Large Datasets
- Scalability with Increasing Number of Movie Reviews
- Sentiment Analysis Model Speed with Large Input Volume
- Memory Usage for Large Recommendation Requests
- System Load Testing with Concurrent Users
- Response Time for Simultaneous Multiple Search Queries
- Response Time for Simultaneous Multiple Recommendations
- Stress Testing of TF-IDF Matrix Generation

8. Security Tests (15)

- SQL Injection Protection in Movie Search
- XSS Protection on User-Submitted Reviews
- Input Validation for User-Submitted Data (E.g., Movie Titles)
- Protection Against CSRF (Cross-Site Request Forgery)
- User Authentication for Restricted Pages
- Data Encryption for Sensitive User Information
- Rate Limiting for API Requests
- CAPTCHA Implementation for Movie Submission
- Handling of Authentication Failures
- Validation for OAuth Tokens (if applicable)
- Prevention of Account Enumeration Attacks
- Prevention of Brute Force Attacks on Login Page
- Securing Flask Session Data
- HTTPS Enforcement for Data Transmission
- Logging of Suspicious User Activity