Web Development 3 2023
Project Specification

1. Overall Project Outline

Develop a Web Application for a topic of your choice using Single Web Page Application (SPA) architecture. The application must use adhere to the architecture shown in Section 3.

The application should:

- Use RESTful web services
- Have CRUD actions for one or more entities
- Have CRUD actions for users/admins
- Use the following features in the UI (based on Twitter bootstrap)
  - Glyphicons
  - Progress bars
  - Cards
  - Dialog boxes/Modal
- Use DataTables
- Be fully responsive (small, medium and large devices).

The application must be fully tested and some test automation is expected.

2. Purpose of Application

The site will be based around some category of products.

You decide on the type of products. It can be cars, books, films, laptops, tourist activities etc.

The product information will be stored in the database.

The product should have at least 8-10 properties (attributes) that can be stored.

You will also have a table of users or admins.

The UI will provide “admin dashboard” functionality. Some examples are given for the UI, but more information will be given later. Some of the functionality that we will add in the UI are

- Login/logout
- CRUD actions on the entities
- CRUD actions on the users
- Graphs

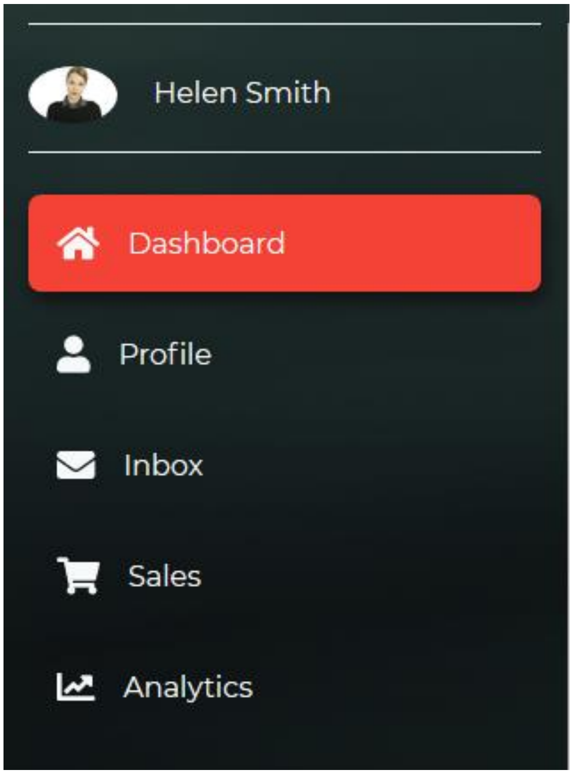


Figure 1: Navbar

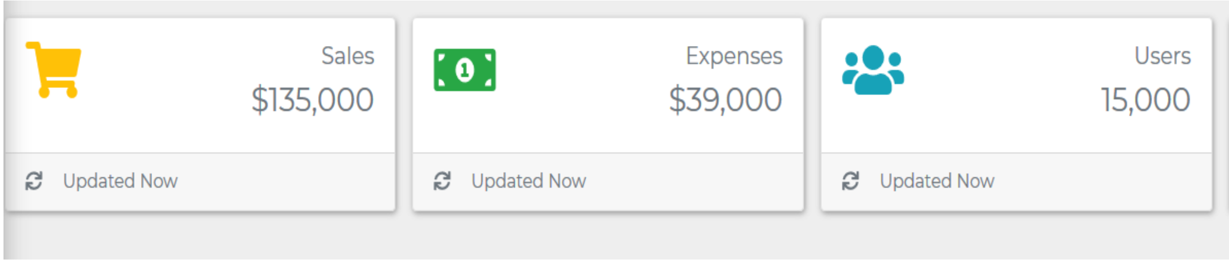
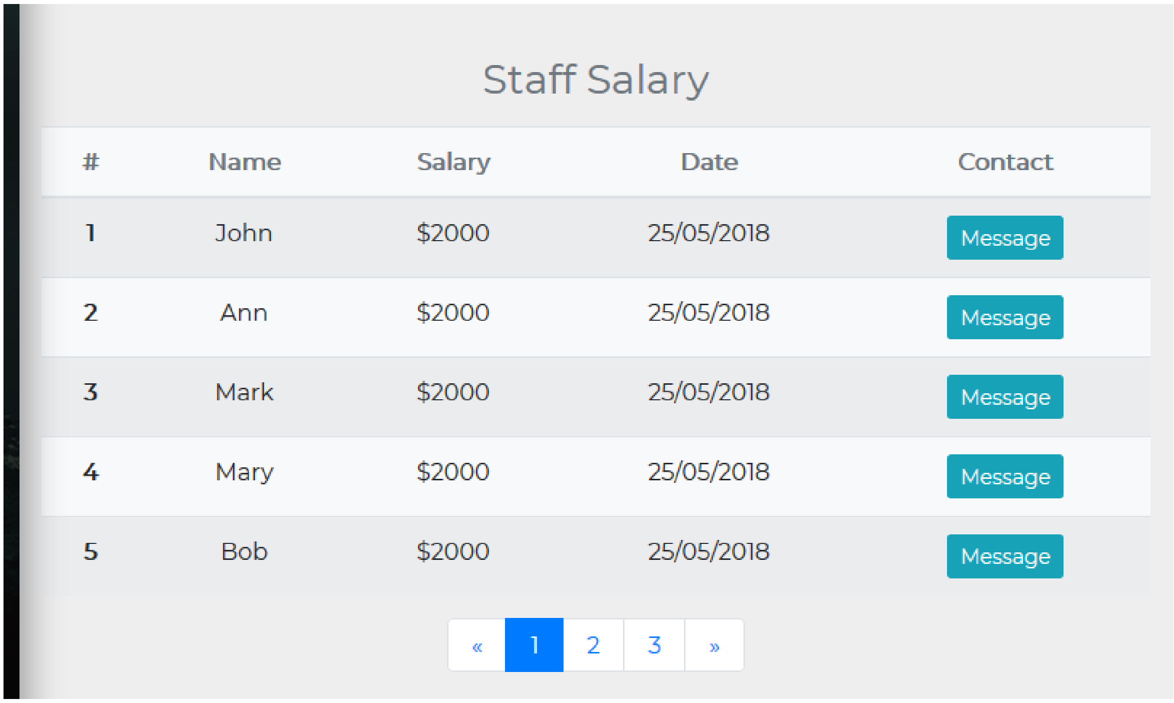


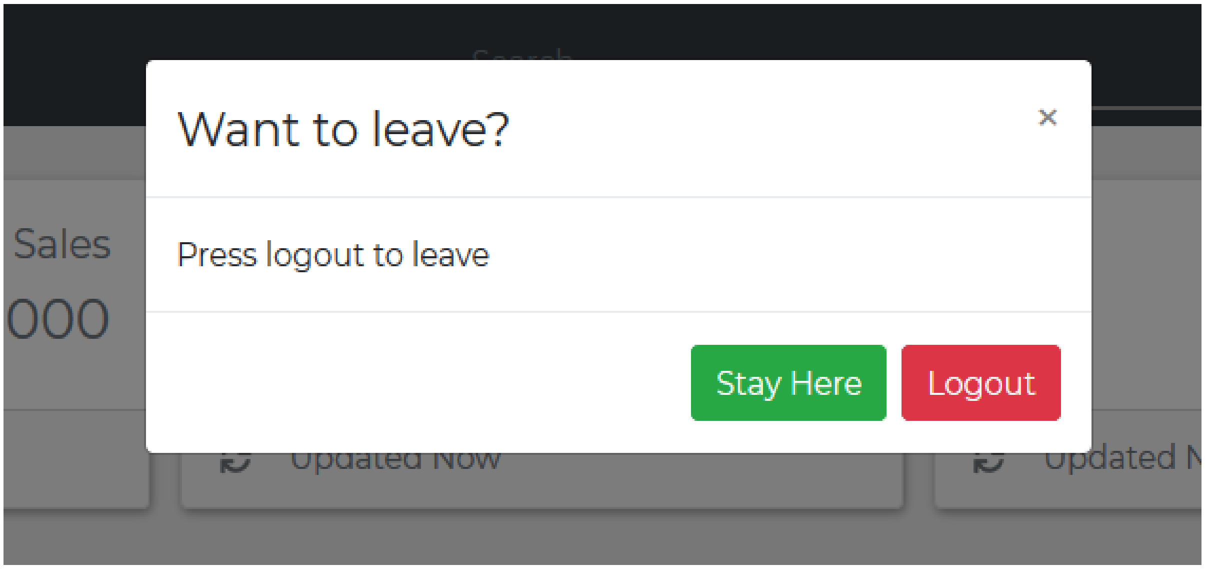
Figure 2: Cards



Staff Salary

#	Name	Salary	Date	Contact
1	John	$2000	25/05/2018	Message
2	Ann	$2000	25/05/2018	Message
3	Mark	$2000	25/05/2018	Message
4	Mary	$2000	25/05/2018	Message
5	Bob	$2000	25/05/2018	Message

Figure 3: Tables with pagination



Want to leave?

Press logout to leave

Figure 4: Login/Logout

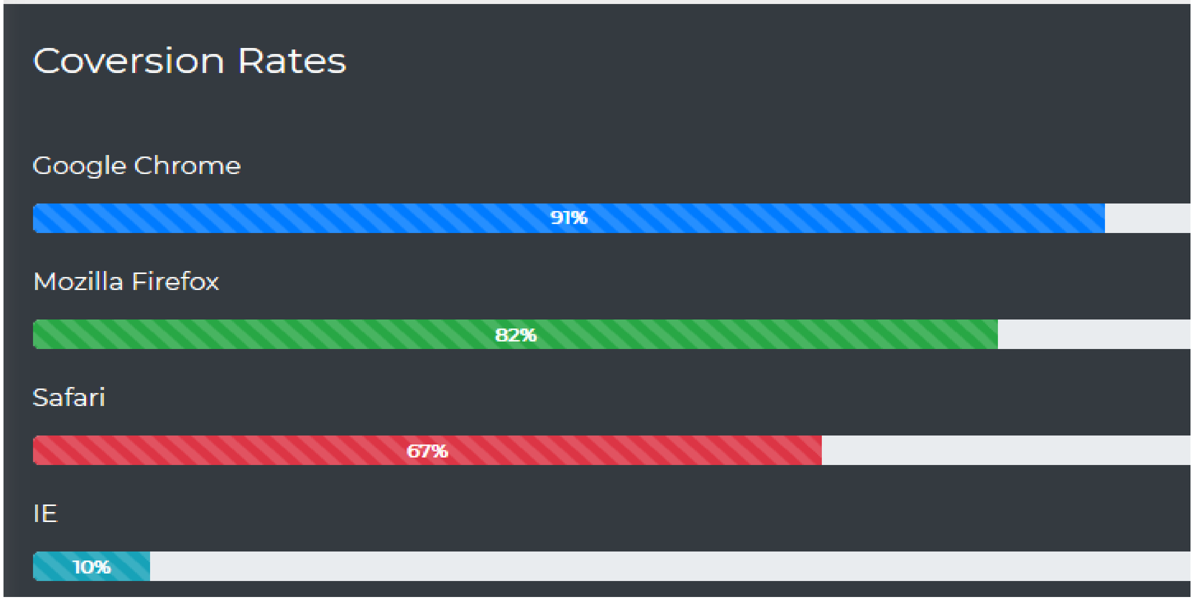
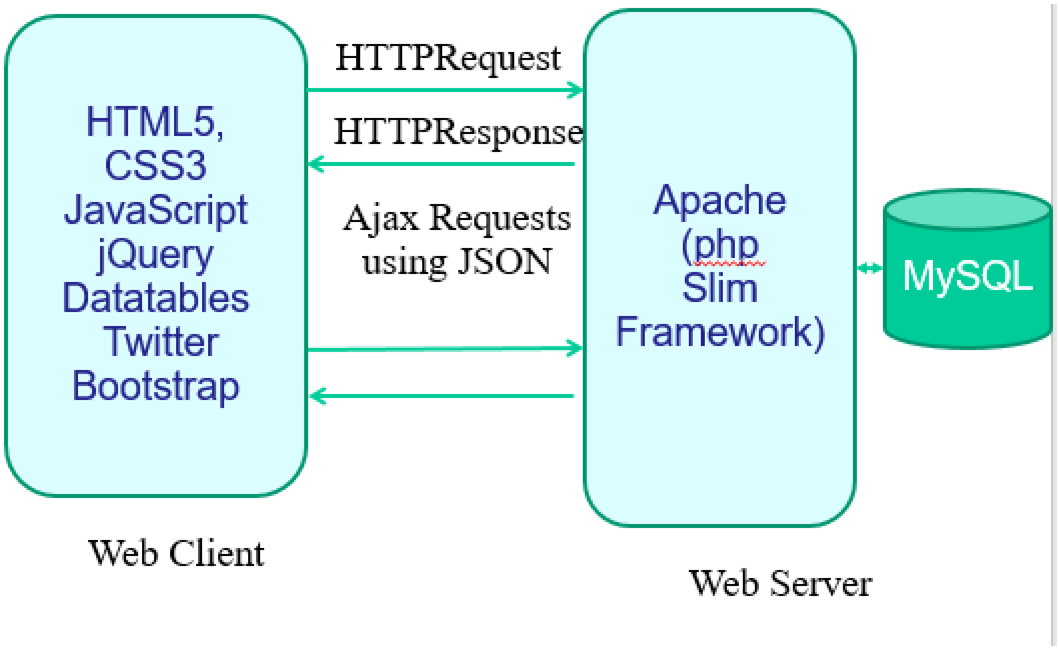


Figure 5:Graphs

3. Project Architecture



4. Deadlines

The project will be completed in 2 parts (sprints), with 2 screencasts explaining your work.

Note: If you miss any part, then you forfeit the marks associated with that part.

Sprint	Date	Marks	Content
#1	**27th October 2023**	**40%**	**User Stories for Sprint#1.** **REST-API demonstrated with Postman**
#2	**24th Nov 2023 (Provisional)**	**60%**	**User Stories for Sprint#2.** **UI tables, cards and nav bar and be responsive** **Include CRUD actions and login/logout using modal**

5. Sprint #1 Server-Side Functionality

On the server side using the given architecture (Section 3), implement a RESTful API that allows for;

Users and for other Entity (e.g. book, movie etc.)

  - Getting details on all
  - Get details for one based on its id
  - Updating details
  - Delete
  - Add

- For the User include attributes name, username, password, image
- All data should be returned in JSON format.
- Use POSTMAN REST client to test the API for the demo.

	What to submit Sprint#1
Code	Zip file containing all your code.
Report	List of User Stories completed
	Test Cases and Test Results for each User Story– Show the output of your automated tests if you have automated them and also the results from POSTMAN.
	Bug List - List any known bugs.
Demo	Demo of working software – Be very clear on the user stories that have been completed. Use POSTMAN to demo. This will be a screencast of you explaining your code and demonstrating its functionality. For the demo have your write up with the test cases to hand. Be prepared to demo all test cases but we may ask for specific ones.
Note	Do not just copy my code and reports… use your own ideas!!!

6. Sprint #2 UI with datatable, navbar, cards, CRUD and Modals.

Details to be specified later

Sample Documentation

Student Number : _________________________

Web Development 3 – Sample Write Up for Sprint#1

SAMPLE SPRINT #1 STORIES

1. As an API Consumer, I want to retrieve data in JSON format for all products, so that I can use the data in my client application.

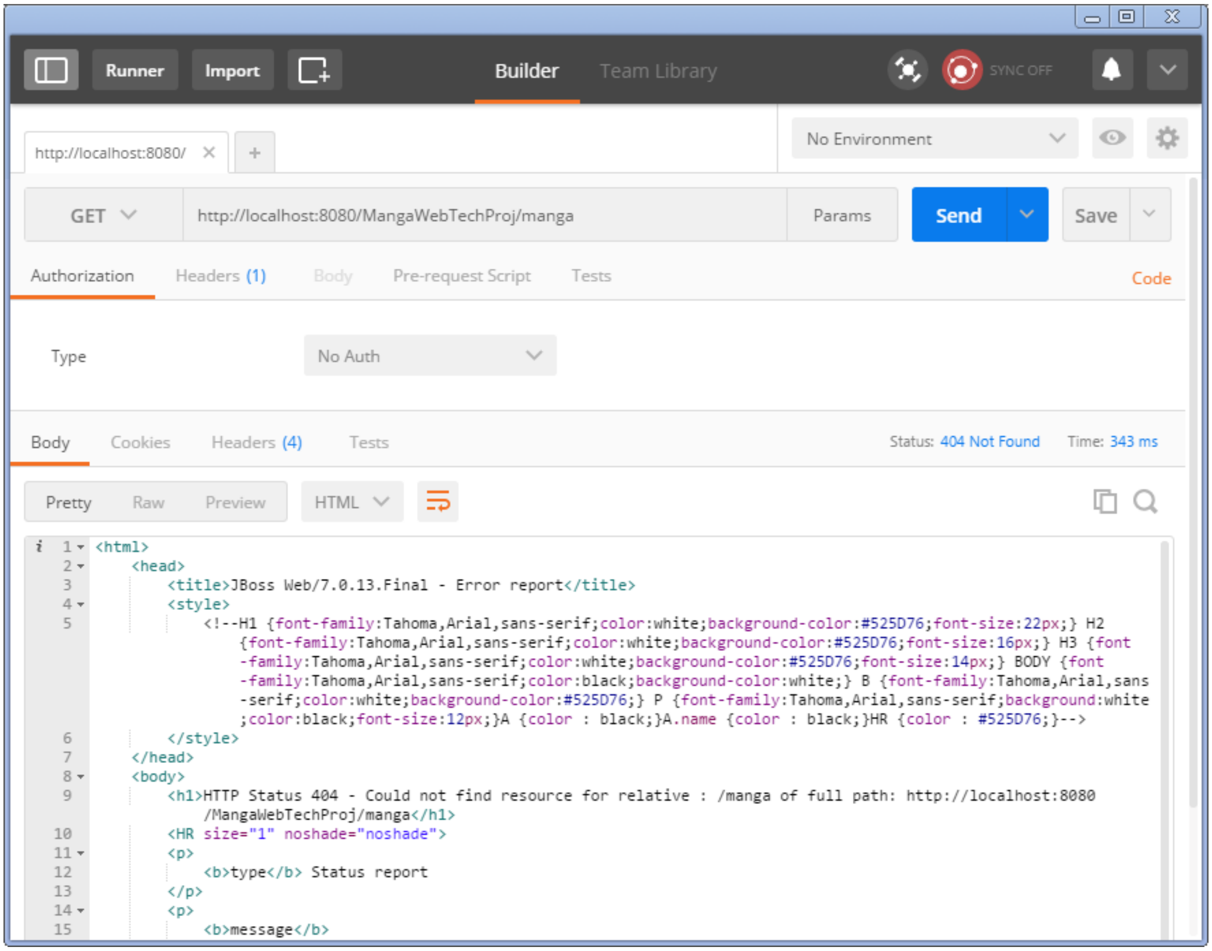
2. As an API Consumer, I want to retrieve data in JSON format for one product based on its ID, so that I can use the data in my client application.

3. …

<Complete your user Story list in this format for all your User Stories>

Manual Testing

User Story	General – Applies to All
Test Case ID	1
Description	Using an invalid URL
Given	1. The Application Server and Database Server is running 2. Postman REST Client Opened
When	1. Set GET request 2. Enter URL http://localhost:8080/MangaWebTechProj/manga/ 3. Press send
Then	1. HTTP Response 404 Not Found
Test Status	Passed



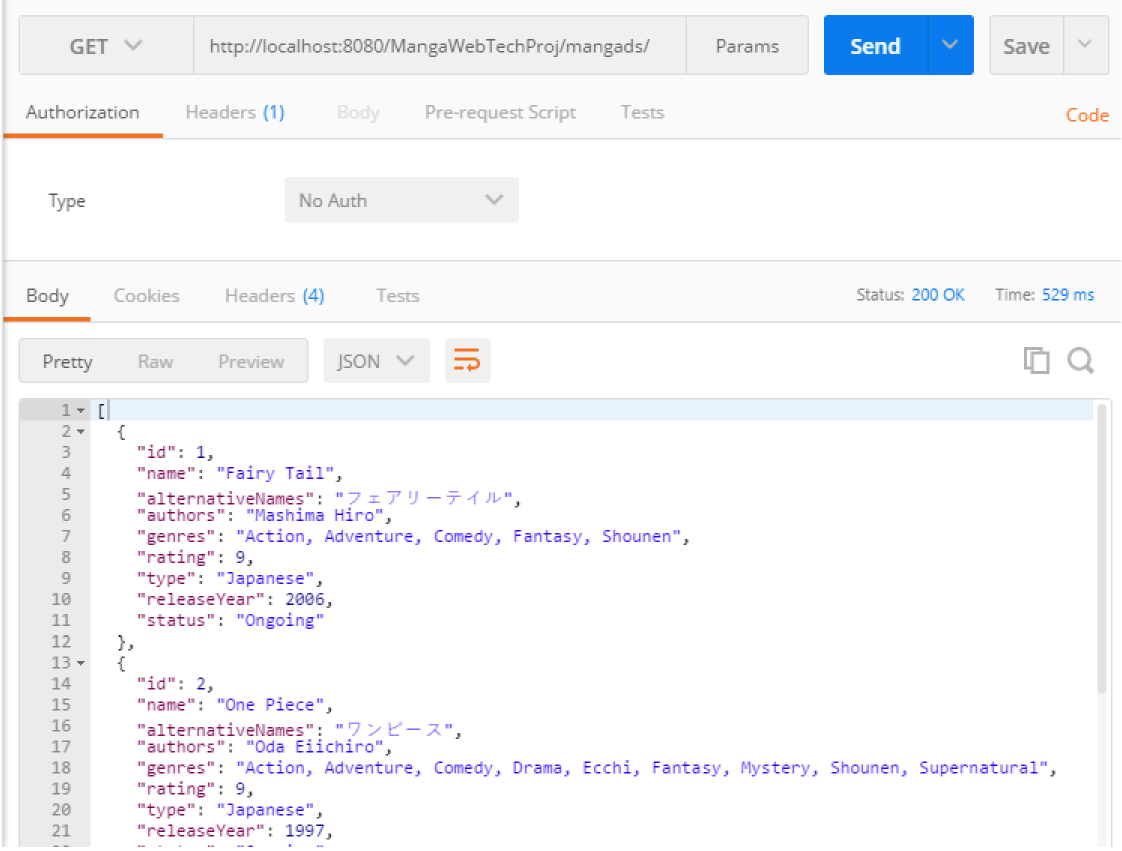
GET http://localhost:8080/MangaWebTechProj/manga

Type: No Auth

Status: 404 Not Found Time: 343 ms

```
<html>
    <head>
        <title>JBoss Web/7.0.13.Final - Error report</title>
        <style>
            <!--H1 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:22px;} H2
                {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:16px;} H3 {font
                -family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:14px;} BODY {font
                -family:Tahoma,Arial,sans-serif;color:black;background-color:white;} B {font-family:Tahoma,Arial,sans
                -serif;color:white;background-color:#525D76;} P {font-family:Tahoma,Arial,sans-serif;background:white
                ;color:black;font-size:12px;}A {color : black;}A.name {color : black;}HR {color : #525D76;}-->
        </style>
    </head>
    <body>
        <h1>HTTP Status 404 - Could not find resource for relative : /manga of full path: http://localhost:8080
            /MangaWebTechProj/manga</h1>
        <HR size="1" noshade="noshade">
        <p>
            <b>type</b> Status report
        </p>
        <p>
            <b>message</b>
```

User Story 1	As an API Consumer, I want to retrieve data in JSON format for all products, so that I can use the data in my client application.
Test Case ID	3
Description	Retrieve data for all manga – Success
Given	1. The Application Server and Database Server is running 2. Postman REST Client Opened
When	1. Set GET request 2. Enter URL http://localhost:8080/MangaWebTechProj/mangads/ 3. Press send
Then	1. HTTP Response 200 OK 2. All database entries are displayed in JSON format
Test Status	Passed



http://localhost:8080/MangaWebTechProj/mangads/

Type: No Auth

Status: 200 OK   Time: 529 ms

```
[
  {
    "id": 1,
    "name": "Fairy Tail",
    "alternativeNames": "フェアリーテイル",
    "authors": "Mashima Hiro",
    "genres": "Action, Adventure, Comedy, Fantasy, Shounen",
    "rating": 9,
    "type": "Japanese",
    "releaseYear": 2006,
    "status": "Ongoing"
  },
  {
    "id": 2,
    "name": "One Piece",
    "alternativeNames": "ワンピース",
    "authors": "Oda Eiichiro",
    "genres": "Action, Adventure, Comedy, Drama, Ecchi, Fantasy, Mystery, Shounen, Supernatural",
    "rating": 9,
    "type": "Japanese",
    "releaseYear": 1997,
```

Sample Test Report Summary

Test Type	Number Passed	Number Failed
Manual	30	0
Automated	10	0

Sample Bug Report

There are currently no known existing bugs in this project.