|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | **Web Application & Service Development with Spring Framework v5** **- 2020** | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | Ralcho | | online |

|  |  |
| --- | --- |
| Project name | RateIT |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| ***RateIT*** provides the ability for users to search movies and series by category and the ability to rate them. The system will be developed using ***Spring 5 Application Development Framework***.It will implement web-based front-end client using ***React.js/Vue.js/Angular*** libraries for asynchronous page data updates. Each page will have a distinct URL, and the routing between pages will be done server side using ***SpringMVC.*** The backend will be implemented as a ***REST/JSON API*** using JSON data serialization. The main user roles (actors in UML) are:  • *Anonymous User* – can only view the movies/series and their information but cannot rate them  • *Registered User* – can view all movies/series and information about them, as well as rate them  • *Administrator* – can manage (create, edit user data and delete) all *Registered Users, add new movie and series to the database* |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Browse information** | The *User* can browse the information views (Home, Movies, Series) | All users |
| * 1. **Register** | *Anonymous User* can register in the system by providing a valid e-mail address, first and last name, and choosing password. | *Anonymous User* |
| * 1. **Login** | *Anonymous User can login in the system by providing the same e-mail address and password, which was provided during registration.* | *Registered User, Administrator* |
| * 1. **Change User Data** | *Registered Users can view and edit their personal User Data.* | Registered User |
| * 1. **Manage Users** | Administrator can browse and filter users based on different criteria: first and last name, email. | *Administrator* |
| * 1. **Add rating to a movie and series** | Registered Users can rate all movies and series from the scale of one to five | *Registered User* |
| * 1. **Search for a movie/series** | *All users can search for available movies and series by providing their name in the search field* | *All users* |
| * 1. **Search by category** | All users can search for available movies/series on their category | *All users* |
| * 1. **View information of a movie/series** | *All users can view the information of a movie/series* | *All users* |
| * 1. **Add new movies and series** | *Administrator can add new movies and series in the database* | *Administrator* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Home** | Presents header with logo, movies/series dropdown with categories, login/register section, search section, random movie/series picker by category (optional) and footer with helpful links and other information | / |
| * 1. **Register** | Presents a view allowing the Anonymous User to register | */register* |
| * 1. **Login** | Presents a view allowing the Users to login | */login* |
| * 1. **Movies** | Presents the movies that are stored in the database and the ability to add, change, delete movies as defined by the Ucs | */movies* |
| * 1. **Series** | Presents the series that are stored in the database and the ability to add, change, delete series as defined by the Ucs | */series* |
| * 1. **User Data** | Provides ability to view and edit personal *User Data* | */personal* |
| * 1. **About** | Presents information about the website | */about* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Users** | GET *User Data* for all users, and POST new *User Data* (Id is auto-filled and modified entity is returned as result from POST request). Available only for *Administrators*. | */api/users* |
| * 1. **User** | GET, PUT, DELETE *User Data* for *User* with specified *userId*, according to restrictions described in UCs. | */api/users/{userId}* |
| * 1. **Login/Register** | POST *User Credentials* (e-mail address and password) and receive a valid *Security Token* to use in subsequent API requests. | */api/auth/login*  */api/auth/register* |
| * 1. **Logout** | POST a logout request for ending the active session and invalidating the issued *Security Token*. | */api/logout* |
| * 1. **Movies** | GET movies and POST new movies (id is auto-filled and modified entity is returned as a result from POST request. | */api/movies* |
| * 1. **Movie** | GET, PUT, DELETE Product Data for movies with specified productid | */api/movies/{movietId}* |
| * 1. **Series** | GET comment for a series with specified serialId and POST new comments (id is auto-filled and modified entity is returned as a result from POST request | */api/series* |
| * 1. **Serial** | GET, PUT, DELETE Comment Data for serial with specified serialID | */api/series/{serialId}* |