

| | |
|--------------|---|
| Assignment 1 | Project Summary |
| Course | Fullstack Application Development with Node.js + Express.js + React.js - 2019 |

| | | |
|----------------|-----------|----------------------|
| Project author | | |
| № | Pseudonym | Face-to-face/ online |
| 1 | yo-yo | face-to-face |

| | |
|--------------|-----------|
| Project name | BookWorld |
|--------------|-----------|

1. Short project description (Business needs and system features)

With so many books to read it is hard to choose your next one and even harder to know if whatever you choose will be good to read. BookWorld offers you to compose your reading list, see others' reviews and express your opinion on books you have already read. The system will be developed as a *Single Page Application (SPA)* using **React.js** as front-end, and **Node.js + express** as backend technologies. Each view will have a distinct URL, and the routing between pages will be done client side using **React Router**. 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 registration and login page
- *User* (extends *Registered User*)– can see the catalog of books and choose to add them to “Favourites”, “Want to read” and “Read”, can leave a comment and a rating of every book.
- *Administrator* (extends *Registered User*) – can add new books to the catalog and can manage the comment sections and remove comment if needed

2. Main Use Cases / Scenarios

| Use case name | Brief Descriptions | Actors Involved |
|------------------------------------|---|--|
| 2.1. Register | <i>Anonymous User</i> can register in the system by providing a valid e-mail address, first and last name, and choosing password. | <i>Anonymous User</i> |
| 2.2. Browse the catalogs | <i>Registered User</i> can view the catalog <i>Administrator</i> can view and edit the catalog | <i>Registered User</i> , <i>Administrator</i> |
| 2.3. Add books to personal library | <i>Users</i> can choose to add books to their “Favourites”, “Want to read” or “Read” libraries | <i>Registered User</i> |
| 2.4. Add a comment to a book | <i>Users</i> can share their likes and dislikes about a book in the comment section | <i>Registered User</i> |

| | | |
|--|--|------------------------|
| 2.5. Search for books | <i>Users can use a search bar to look for a book by using the whole name or part of the name of a book</i> | <i>Registered User</i> |
| 2.6. Filter the catalog by category | <i>Users can choose to see books by category</i> | <i>Registered User</i> |
| 2.7. Manage catalog | <i>Administrators can add and edit books.</i> | <i>Administrator</i> |
| 2.8. Manage comments | <i>Administrators can choose to delete a comment if it breaks the rules of the site.</i> | <i>Administrator</i> |
| 2.9 Rate book | <i>Users can rate books from 1 to 5 stars</i> | <i>Registered User</i> |
| 2.10 Recommend Book | <i>Users can recommend books to other users</i> | <i>Registered User</i> |

| 3. Main Views (SPA Frontend) | | |
|------------------------------|--|---------------|
| View name | Brief Descriptions | URI |
| 3.1. Home | Presents a small selection of books | / |
| 3.2. Catalog | Presents all books and allows users to add a book to their library without changing the page and to select a category of books to be shown | /catalog |
| 3.3. Book | Presents a book with all available details and all the comments | /book/{id} |
| 3.4. Favourites | Presents all books that the current user have added to their “Favourites” library | /favourites |
| 3.5. Wish list | Presents all books that the current user have added to their “Want to read” library | /wishlist |
| 3.6. Read | Presents all books that the current user have added to their “Read” library | /read |
| 3.7. Registration | Presents a view allowing the <i>Anonymous Users</i> to register in the system | /registration |
| 3.8. Login | Presents a view allowing the users to login. | /login |

| 4. API Resources (Node.js Backend) | | |
|------------------------------------|--|---------------------|
| View name | Brief Descriptions | URI |
| 4.1. Users | GET <i>User Data</i> for all users, and POST new <i>User Data</i> . Available only for <i>Administrators</i> . | /api/users/ |
| 4.2. User | GET, PUT, DELETE <i>User Data</i> for <i>User</i> with specified <i>userId</i> | /api/users/{userId} |

| | | |
|----------------------------|--|----------------------------|
| 4.3. Login | POST <i>User Credentials</i> (e-mail address and password) | <i>/api/login</i> |
| 4.4. Logout | POST a logout request for ending the active session | <i>/api/signOut</i> |
| 4.5. Books | GET, POST <i>Books</i> | <i>/api/books/</i> |
| 4.6. Book | GET, PUT <i>Book</i> with specified bookId | <i>/api/books/{bookId}</i> |
| 4.7. Comment | GET, PUT, POST, DELETE <i>Comment</i> for a specific book | <i>/api/comment</i> |
| 4.8. Favourites | GET, POST, DELETE books from favourites library | <i>/api/favourites</i> |
| 4.9. Want to read | GET, POST, DELETE books from “Wish list” library | <i>/api/wishList</i> |
| 4.10. Read | GET, POST, DELETE books from “Read” library | <i>/api/read</i> |
| 4.11. Notifications | POST,GET,PUT notifications | <i>/api/notifications</i> |