|  |  |  |
| --- | --- | --- |
| **Assignment 1** | Course project | |
| **Course** | Full-stack Application Development with Node.js + Express.js + React.js - 2025 | |
|  | | |
| **Project author** | | |
| Name | | FN |
| Hristiana Karaivanova | | 3MI0700178 |

|  |  |
| --- | --- |
| **Project name** | **Online shop for handmade gifts and clothing** |
| GitHub Repository | https://github.com/hrisa006/RakodelkoSYS |

|  |
| --- |
| 1. **Short project description (business needs and system features)** |
| RakodelkoSYS („*Ръкоделко*“) is a Bulgarian [e-commerce](https://en.wikipedia.org/wiki/E-commerce) company that follows in the tradition of open craft fairs. It gives sellers (craftsmen) an opportunity to have a personal storefront where they can list their goods for a fee of BGN 0.50 per item. The focus is on handmade or vintage items and craft supplies. These items fall under a wide range of categories - jewelry, bags, clothing, home décor and furniture, toys, art, craft supplies and tools. Thanks to this huge variety, the store can reach the interest of all types of buyers.  The store will be developed as a *Single-Page Application* (SPA) with a multi-view structure using *React.js* with TypeScript for the front-end and *Node.js + Express.js* as back-end technologies. The front-end communicates with the back-end via *REST API*, while *PostgreSQL* serves as the primary database for structured data storage. The main user roles (actors in UML) are:   * *Anonymous User* – can view the main pages of the site (*Home*, *New In*, *Sale*, *Login*, *About*); * *Registered User* – can create and publish a listing to the marketplace, as well as delete a listing, can also purchase an item from the marketplace; * *Administrator* – can manage (create, edit and delete) all *Registered Users*, *Items* and *Purchases*. |

|  |  |  |
| --- | --- | --- |
| 1. **Main Use Cases / Scenarios** | | |
| Use case name | Brief Descriptions | Actors Involved |

|  |  |  |
| --- | --- | --- |
| * 1. **Browse through main pages** | *User* can browse through the main pages (*Home*, *New In*, *Sale*, *Login*, *About*) in RakodelkoSYS. | All users |
| * 1. **Register** | *Anonymous User* can register in the system of Rakodelko by providing a valid e-mail address, name and password, which fulfils the requirements. By default, all *Registered Users* are supplied with a seller’s  account but the decision whether to upload or not listings is up to them.  *Administrator* can register new users by entering *User Data*. | *Anonymous User*, *Administrator* |
| * 1. **Login** | *Registered User* can log in to the system of Rakodelko by entering a valid registered e-mail address and a valid password, created by him/her. | *Registered User* |
| * 1. **Manage Users** | *Administrator* can browse and filter users based on different criteria: first name, email.  *Administrator* can choose a *User* and manage him/her - edit (using Change User Data UC) or delete.  *Administrator* can create a new user using Register UC. | *Administrator* |
| * 1. **Manage Items** | *Registered User* can browse his/her *Items*, add new *Item* using Add/Edit Item UC and delete an *Item*.  *Administrator* can browse *Items* of all *Registered* *Users*, edit and delete them. | *Registered User*, *Administrator* |
| * 1. **Manage Purchases** | *Registered User* can browse his/her *Purchased Items*, add new *Item* using Add/Edit Purchased Items UC and cancel (delete) a *Purchase*.  *Administrator* can browse *Purchased Items* of all *Registered Users*, edit and delete them, and cancel (delete) a *Purchase*. | *Registered User*, *Administrator* |
| * 1. **Add/Edit Item** | *Registered User* can upload an *Item* (listing).  *Administrator* can add/edit an *Item*. | *Registered User*, *Administrator* |
| * 1. **Add/Edit Purchased Items** | *Registered User* can add an *Item* (listing) to a *Purchase* or edit an already added one before the purchase request is submitted – increase the amount of it or remove it.  *Administrator* can add/edit *Item* in a *Purchase*. | *Registered User*, *Administrator* |

|  |  |  |
| --- | --- | --- |
| 1. **Main Views (SPA Front-end)** | | |
| View name | Brief Descriptions | URI |
| * 1. **Home** | Introduction page to the RakodelkoSYS website and some information about the company and its purpose. There is a given opportunity for registration of new users or logging in to the site if an account has been created already. | */* |
| * 1. **New In** | A selection of the latest items, which were uploaded by users to the website, is displayed there. | */new* |
| * 1. **User Registration** | Presents a view allowing the *Anonymous Users* to register in the Rakodelko system. | */register* |
| * 1. **Login** | Presents a view allowing the users to login. | */login* |
| * 1. **User Data** | There, the *Registered Users* are allowed to view and edit their personal *User Data*. They can also make a request for deleting their account. | */profile* |
| * 1. **About** | Presents information about the Rakodelko company and its owners, as well as contact information. | */about* |
| * 1. **Cart** | Shows *Purchased Items* that *Registered User* added to the cart. | */cart* |

|  |  |  |
| --- | --- | --- |
| 1. **API Resources (Node.js Back-end)** | | |
| View name | Brief Descriptions | URI |

|  |  |  |
| --- | --- | --- |
| * 1. **Users** | GET *User Data* for all *Users* and POST new *User Data* (ID is auto-filled by RakodelkoSYS 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** | POST *User Credentials* (e-mail address and password) and receive a valid *Security Token* to use in subsequent API requests. | */api/auth/login* |
| * 1. **Logout** | POST *User Credentials* (e-mail address and password) and receive a valid *Security Token* to use in subsequent API requests. | */api/auth/logout* |
| * 1. **Orders** | GET *Order Data* for *User* with specified *orderID* and POST new *Order Data*, according to restrictions described in UCs. | */api/orders* |
| * 1. **Items** | GET *Items* and POST new *Item* (ID is auto-filled by RakodelkoSYS and modified entity is returned as result from POST request). | */api/items* |
| * 1. **Item** | GET, PUT, DELETE *Item Data* for *Item* with specified *itemID*, according to restrictions described in UCs. | */api/items/{itemID}* |
| * 1. **Comments And Ratings** | GET, PUT, DELETE *Comments And Ratings* for *Item* with specified *itemID*. | */api/items/{itemID}/reviews* |
| * 1. **Invoice** | GET *Invoice* for *Item* if *User* request it. | */api/orders /{orderID}/invoice* |
| * 1. **Multimedia** | GET, PUT, DELETE *Pictures*/*Videos* related to *Items*. | */api/items/{itemID}/media* |
| * 1. **Shipping** | GET, PUT, DELETE *Address Data* of *Purchased Item*. | */api/orders /{orderID}/shipping* |