|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | Fullstack Application Development with Node.js + Express.js + React.js - 2019 | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | Van97 | | face-to-face |

|  |  |
| --- | --- |
| Project name | Online Beauty Store |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| We’ve changed the way we go shopping.. The ***Online Beauty Store (OBS)*** provides the best beauty shopping destination to buy cosmetic products, hair products & skincare online at best prices. In addition to that it allows users to register and rate the products. 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 **mongoDB**. There will be also a PayPal Checkout with **Smart Payment Buttons** integration. The main user roles are:  • *Anonymous User* – can only view products information and add them to the cart.  • *Registered User* – can view products information, rate products, add them to the cart, submit delivery details and checkout through PayPal service.  • *Administrator* (extends *Registered User*) – can manage (create, edit products data and delete) all *Products*, as well as *Registered Users and track orders.* |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Review products information** | The *User* can review the main products information in *OBS*, and can click on a specific product to read description.  *Administrator* can view and edit *products information* of all *products* | All users  *Administrator* |
| * 1. **Add to Cart** | *The User* can choose from the product’s page or from the product’s detail modal to add a specific item to its cart. | All users |
| * 1. **Manage Cart** | *The User* can increase/decrease each product's amount, fully empty the cart and see the tax and discount depending on the changes done. | 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. **Change User Data** | *Registered User can change its account data (*e-mail address, first, last name, password*).*  *Administrator* can view and edit *User Data* of all *Users* | *Registered User,*  *Administrator* |
| * 1. **Rate Products** | *Users can see the rating of all products. Rating is calculated using arithmetic mean of the data of all submits.*  *Registered Users can rate each product (not more than once for a product)*. | All users,  Registered Users |
| * 1. **Enter**   **Delivery Information** | *Registered Users* can add new address and telephone for their order or use the old one. | *Registered Users* |
| * 1. **Paying** | *Registered Users* can choose to pay cash or using the PayPal services. | *Registered Users* |
| * 1. **Tracking** | *Administrators* can track orders in real-time. | *Administrator* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (SPA Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Home** | Presents the products and promotions the store is offering and abilities to choose details, add them to cart, submit rate and allowing Administrators to manage them. | / |
| * 1. **Details** | Presents description of a product and allow users to add it to cart or return back and continue shopping. | */details* |
| * 1. **Cart** | Presents list of all added products and options to manage their amount in parallel with the tax and total price, remove each or empty the cart. Also allows adding delivery information, choose a payment method and pay. | */cart* |
| * 1. **User Registration** | Presents a view allowing the *Anonymous Users* to register in *OBS*. | */register* |
| * 1. **Navbar** | Presents a view allowing the users to go shop for products, check their cart, register, login or logout. |  |
| * 1. **Profile** | Presents ability to view and edit personal *User Data*. | */profile* |
| * 1. **Orders** | Presents in real time the submitted order. | */orders* |
| * 1. **Users** | Presents ability to manage (CRUD) Users and their User Data (available for *Administrators* only, as described in UCs). | */users* |
| * 1. **About** | Presents information about the *OBS* project and his owner. | */about* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Node.js Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Users** | GET *User Data* for all users, and POST new *User Data* (Id is auto-filled by *OBS* 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 decribed 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/login* |
| * 1. **Logout** | POST a logout request for ending the active session with *OBS,* and invalidating the issued *Security Token*. | */api/logout* |
| * 1. **Products** | GET *products* and POST new *products* (Id is auto-filled by *OBS* and modified entity is returned as result from POST request). | */api/products* |
| * 1. **Product** | GET, PUT, DELETE product for *products* with specified *productId*. | */api/products/{productId}* |
| * 1. **Orders** | GET orders and POST new *order* (Id is auto-filled by *OBS* and modified entity is returned as result from POST request). | */api/order* |
| * 1. **Order** | GET, PUT, DELETE *order* for *orders* with specified *orderId*. | */api/orders/{ordersId}* |