|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | Fullstack Application Development with Node.js + Express.js + React.js - 2019 | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | Martin Yotov 61958 | | face-to-face |

|  |  |
| --- | --- |
| Project name | e-Gift Cards Marketplace |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| The e-Gift Cards Marketplace is an online platform where people can buy gift cards for various online market stores with nice discount or exchange their unneeded gift cards for cash. Users can search for e-vouchers and browse them by store, add them to watchlist or contact sellers.  The system will be developed as a *Single Page Application (SPA)* using ***React.js*** as front-end, and ***Node.js + Express server*** 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. There will be also a real-time online chat between users using ***Socket.IO.***  The main user roles (actors in UML) are:  • *Guest User* – can only view the gift cards page.  • *Registered User* – can contact sellers, add gift cards to watchlist and upload ones for selling  • *Administrator* – can remove gift card offers, create new stores. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| **2.1 Register** | *Guest User* can register in the system by providing a valid e-mail address, username, and choosing password. | *Guest User* |
| **2.2. Login** | *Guest User* can login with credentials. Customers can also login using third party authentication, i.e Facebook login or Google login | *Guest User* |
| **2.3. Change User Data** | *Registered User* can view and edit their personal *User Data, manage their watchlist.*  *Administrator* can view and edit *User Data* of all *Users*. | *Registered User, Administrator* |
| **2.4 Manage Users** | *Administrator* can browse and filter users based on different criteria: first and last name, email, offers | *Administrator* |
| **2.5. Manage Platform Content** | *Administrator* can add new stores.  *Administrator* remove existing gift card offers. | *Administrator* |
| **2.6 Browse gift cards** | The *User* can browse all gift cards. | *All Users* |
| **2.7 Browse Top Deals** | The *User* can browse gift cards with biggest discount. | *All Users* |
| **2.8 Browse stores** | The *User* can browse different gift cards by store. | *All Users* |
| **2.9 Search gift cards** | The *User* search for a gift cards using keywords. | *All Users* |
| **2.10 See gift card details** | The *User* can see gift card details.  *Registered User* add gift cards to watchlist. | *All Users, Registered User* |
| **2.11. Contact Seller** | *Registered User* can contact gift card sellers | *Registered User* |
| **2.12. Create gift card offer** | *Registered User* can add new gift cards for selling with certain discount. | *Registered User* |
| **2.13 Add/Remove gift cards to watchlist** | *Registered User* can add and remove gift cards to his watch list. | *Registered User* |
| **2.14 Verify a deal** | *Registered User* can pay for a gift card and the transaction will be verified. | *Registered User* |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (SPA Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| **3.1. Home** | Presents the introductory information for the purpose of the system as well as detailed instructions how to start using it. Prominently offers ability to register. | / |
| **3.2. Register** | Register form page | /register |
| **3.3. Login** | Login form page | /login |
| **3.4. Gift cards Catalog** | Presents available gift cards. Offers abilities to browse, search and filter. | */gift-cards* |
| **3.5 Gift card details** | Presents the gift card offer details. | */gift-cards/{id}* |
| **3.6 Stores** | Presents the stores for which there are offers about. | */stores* |
| **3.7 Store cards** | Presents a store with the gift card offers for it. | */stores/{id}* |
| **3.8 Users** | A list of all users (can filter sellers) (A*dministrators* only) | */users* |
| **3.9 User Profile** | Presents User’s profile, data, watchlist. | */users/{id}* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Node.js Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| **4.1 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* |
| **4.2 User** | GET, PUT, DELETE *User Data* for *User* with specified *userId*, according to restrictions described in UCs. | */api/users/{userId}* |
| **4.3 Login** | POST *User Credentials* (e-mail address and password) and receive a valid *Security(JWT) Token* to use in subsequent API requests. | */api/login* |
| **4.4 Logout** | POST a logout request for ending the active session*,* and invalidating the issued *Security Token*. | */api/logout* |
| **4.5 Gift Cards** | GET *Gift Cards*, and POST new *Gift Cards* (Id is auto-filled and modified entity is returned as result from POST request) | */api/gift-cards* |
| **4.6 Stores** | GET *Stores*, and POST new *Stores* (Id is auto-filled and modified entity is returned as result from POST request) | */api/stores* |
| **4.7 Best Deals** | GET offers with biggest discount for the buyer. | */api/best-deals/* |
| **4.8 WatchList** | GET, PUT, POST, DELETE offers from user’s watchlist. | */api/users/{id}/watchlist* |