|  |  |  |  |
| --- | --- | --- | --- |
| 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 | AutoDeli |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| Nowadays every house has one or more cars***. AutoDeli*** is an online magazine for auto parts***.*** It provides the users to search and buy specific auto parts and consumables such as engine oil, windshield wipers and more. 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 view all auto parts and consumables, search by specific car model and make an order  • *Registered User* – can view all auto parts and consumables, search by specific car model, make an order and get offers in provided email address  • *Administrator* – can manage (create, edit user data and delete) all *Registered Users, add new auto part or consumable and send offers to Registered Users* |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Browse information** | The *User* can browse the information views (Home, Auto Parts, Consumables) | 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. **Manage Users** | Administrator can browse and filter users based on different criteria: first and last name, email. | *Administrator* |
| * 1. **Add new auto part or consumable** | Administrator can add new auto part or consumable with all it needed details | *Administrator* |
| * 1. **Search by auto part or consumable** | *All users can search for available auto parts and consumables* | *All users* |
| * 1. **Search by specific car model** | All user can search for auto parts and consumables | *All users* |
| * 1. **View information of a auto part/consumable** | *All users can view the information of an auto part or consumable* | *All users* |
| * 1. **Add auto part or consumable to shopping cart** | *All Users can add new auto part or consumable to shopping cart* | *All Users* |
| * 1. **View or edit shopping cart** | *All Users can view and edit their shopping cart* | *All Users* |
| * 1. **Make an order** | *All Users can make an order by providing phone number, first and last name and email address. If the users is registered the fields are auto filled with his credentials* | *All Users* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Home** | Presents header with logo, login/register section, search box | / |
| * 1. **Register** | Presents a view allowing the Anonymous User to register | */register* |
| * 1. **Login** | Presents a view allowing the Users to login | */login* |
| * 1. **Oils and liquids** | Presents the variety of engine oil, brake fluid | */oil* |
| * 1. **Batteries** | Presents the variety of car batteries | */batteries* |
| * 1. **Supplements** | Present the variety of car supplements | */supplements* |
| * 1. **Orders** | Present information about the ordered items | */orders* |
| * 1. **Cart** | Present information about the items in the shopping cart | */cart* |
| * 1. **Auto Parts** | Present the variety of car auto parts | */autoparts* |

|  |  |  |
| --- | --- | --- |
| 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. **Oils** | GET oils and POST new oil (id is auto-filled and modified entity is returned as a result from POST request) | */api/oils* |
| * 1. **Oil** | GET, PUT, DELETE Product Data for oil with specified productid | */api/oils/{oilId}* |
| * 1. **Batteries** | GET batteries and POST new battery (id is auto-filled and modified entity is returned as a result from POST request) | */api/batteries* |
| * 1. **Battery** | GET, PUT, DELETE Product Data for battery with specified batteryID | */api/batteries/{batteryId}* |
| * 1. **Supplements** | GET supplements and POST new supplement (id is auto-filled and modified entity is returned as a result from POST request) | */api/supplements* |
| * 1. **Supplement** | GET, PUT, DELETE Product Data for supplement with specified supplementID | */api/supplements/{supplementId}* |
| * 1. **Brands** | GET supplements and POST new car brand (id is auto-filled and modified entity is returned as a result from POST request) | */api/brands* |
| * 1. **Brand** | GET, PUT, DELETE Product Data for brand with specified brandID | */api/brands/{brandId}* |
| * 1. **Models** | GET supplements and POST new car brand model (id is auto-filled and modified entity is returned as a result from POST request) | */api/brands/{brandId}/models* |
| * 1. **Model** | GET, PUT, DELETE Product Data for car brand model with specified modelID | */api/brands/{brandId}/models/{modelId}* |
| * 1. **Engines** | GET supplements and POST new car brand model engine (id is auto-filled and modified entity is returned as a result from POST request) | */api/brands/{brandId}/models/{modelId}/engines* |
| * 1. **Engine** | GET, PUT, DELETE Product Data for car brand model engine with specified engineID | */api/brands/{brandId}/models/{modelId}/engines/{engineId}* |
| * 1. **Auto Parts** | GET supplements and POST new auto part (id is auto-filled and modified entity is returned as a result from POST request) | */api/brands/{brandId}/models/{modelId}/engines/{engineId}/autoparts* |
| * 1. **Auto Part** | GET, PUT, DELETE Product Data for auto part with specified autoPartID | */api/brands/{brandId}/models/{modelId}/engines/{engineId}/autoparts/{autopartId}* |