|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | **Web Application & Service Development with Spring Framework v5** **- 2018** | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | SMT | | face-to-face |

|  |  |
| --- | --- |
| Project name | “CAA Drone Manager” |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| As drones get more and more popular the airspace gets more and more cluttered. By law in Bulgaria to fly your drone you have to make a request and get permission from the GD CAA (General Directorate Civil Aviation Administration) which is done by e-mail or fax and has to be done at least 8 days before flying. The “CAA Drone Manager” will make the process faster. With it the whole process can be done in less than 5 minutes. The “CAA Drone Manager” allows citizens to register and request flying permission.  The system will be developed using ***Spring 5 Application Development Framework***.It will implement web-based front-end client using ***Thymeleaf*** templates and ***jQuery*** or ***Angular*** for asynchronous page data updates. Each page will have a distinct URL, and the routing between pages will be done server side using ***SpringMVC/ WebFlux***. The backend will be implemented as a ***REST API*** using JSON data serialization. There will be also a real-time event streaming from the server to the web client using ***WebFlux*** in order to allow the Authorized users to view on real time the load of air and approve/disapprove requests. The main user roles are:  • *Anonymous User* – can only view the information pages.  • *Citizen* (extends *Registered User*) – can request flying permission.  • *Authorized* (extends *Registered User*) – can block parts of air spaces and approve/disapprove requests.  • *Administrator* (extends *Registered User*) – can manage (create, edit user data and delete) all *Registered Users*, *Requests* and *Airspace maps*. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Browse information** | All visitors can view information on the site like laws, requirements etc. | All users |
| * 1. **Register** | *Anonymous User* can register in the system by providing a valid e-mail address, first and last name and valid EGN. By default, all new registered users have *Student* role. The student can provide a *Token Key* in order to identify the *Student Group* to be enrolled to.  *Administrator* can register new by entering *User Data* and choosing a Role (*Student, Instructor*, or *Administrator*). | *Anonymous User, Administrator* |
| * 1. **Change User Data** | *Registered User* can view and edit his personal *User Data*.  *Administrator* can view and edit *User Data* of all *Users* and assign them more/less rights. | *Registered User, Administrator* |
| * 1. **Manage Users** | *Administrator* can browse and filter users based on different criteria: first and last name, email, Role, EGN.  *Administrator* can choose a *User* to manage, and can manage the chosen User - edit (using Change User Data UC) or delete.  *Administrator* can create a new user using *Register UC*. | *Administrator* |
| * 1. **Manage Requests** | *Citizen can create, edit and remove his own requests.*  *Authorized* can approve/disapprove requests made by *Citizen.*  *Administrator* can do all the things *Authorized* does plus remove request without giving permission. | *Citizen, Authorized, Administrator* |
| * 1. **Monitor Air Space Load** | *Instructor/Administrator* can view how many approved requests are in an area. | *Instructor, Administrator* |
| * 1. **Manage Permitted Air Space** | *Administrator* can set and remove no-fly zones. | *Administrator* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 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. | / |
| * 1. **Requests** | Presents a form to create a new request. *Citizes* ‘s requests. | */requests* |
| * 1. **Air space load** | Presents a view of the currently accepted requests and area. | */load* |
| * 1. **User Registration** | Presents a view allowing the Anonymous Users to register. | */register* |
| * 1. **Login** | Presents a form to login. | */login* |
| * 1. **Citizen Profile** | Presents ability to view and edit personal User Data. | */profile* |
| * 1. **No-Fly zones** | Presents a view with the no-fly zones. | */zones* |
| * 1. **Dashboard** | For *Authorized and above* presents a view of the not processed requests. | */dashboard* |
| * 1. **Citizens** | For *Administrator presents all registered citizens.* | */citizens* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (TODO) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Citizens** | GET *User Data* for all citizens, and POST new *User Profile* (Id is auto-filled and modified entity is returned as result from POST request). | */api/citizens* |
| * 1. **Citizen** | GET, PUT, DELETE *User Data* for *User* with specified *userId*. | */api/citizens/ {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 and invalidating the issued *Security Token*. | */api/logout* |
| * 1. **No-Fly Zones** | GET No-fly zones, and POST new zones (For *Administrator*). | */api/zones* |
| * 1. **Air space load** | GET the approved requests. | */api/load* |
| * 1. **Requests** | GET all requests. Property for approved/declined/not processed | */api/requests* |
| * 1. **Request** | GET, PUT, DELETE *Requests* | */api/request/ {requestId}* |
| * 1. **Request Results** | GET *Request* Results for a specific requestId | */api/request/ {requestId} /results* |