|  |  |  |  |
| --- | --- | --- | --- |
| **Assignment 1** | | Project Summary | |
| **Course** | | Web Application & Service Development with Spring Framework v5 - 2019 | |
|  | | | |
| **Project author** | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1  2  3 | Owl  Суфле  Pik1916 | |  |

|  |  |
| --- | --- |
| **Project name** | Medical examination management system (MEMS) |

|  |
| --- |
| 1. **Short project description (Business needs and system features)** |
| The Medical examination management system (MEMS) provides ability for real time tracking of examination result status, view patient clinical history and easy way to book examination appointments. The system helps doctors to update and view all his/her patient examinations and if needed change visiting hours. In addition to that it allows users to register, and administrators to manage them.  The system will be developed using Spring 5 Application Development Framework. It will implement web-based front-end client using **Thymeleaf/JSP** templates and **jQuery** or **React.js/Vue.js/Angular JavaScript/TypeScript** 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/ WebFlux**. The backend will be implemented as a **REST/JSON API** using JSON data serialization. There will be also a real-time event streaming from the server to the web client using **WebFlux** and **Server Sent Events (SSE)/WebSocket** in order to allow the Patients and Doctors to monitor status of examinations results in real time.  The main user roles (actors in UML) are:   * ***Anonymous User*** - can only view the information pages, check status and result of examination by providing key, without booking appointment. * ***Patient*** (extends Registered User) - can make/cancel/edit appointment for examination, see his/hers medical record for all previous done examinations and monitor the completion status of the examination result in real time. * ***Doctor*** (extends Registered User) - write down the information of patient examination, can make/cancel/edit appointment for examination and see the schedule of the room he/she is working. * ***Administrator*** (extends Registered User) - can manage all Registered Users, by creating, deleting and editing user data and clinical wards, as well as manage appointments by editing them. |

|  |  |  |
| --- | --- | --- |
| 1. **Main Use Cases / Scenarios** | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| **2.1.**  **Browse information** | All Users can browse information in the home and about view of the Medical examination management system. | All users |
| **2.2.**  **Check result from examination** | Anonymous User can check last done examination status and result by providing a key.  Registered Users can choose which concrete examination to see. | All users |
| **2.3.**  **Register** | Anonymous User can register in the system by providing a valid e-mail address, first and last name, and choosing password. By default, all new registered users have Patient role.  Administrator can register new user by entering User Data and choosing a role (Patient, Doctor or Administrator). In case Administrator has chosen Doctor role, he/she needs to assign Clinical ward for the concrete Doctor. | Anonymous User, Administrator |
| **2.4.**  **Change personal data** | Patient, Doctor and Administrator can view and edit her/his personal User Data. | Patient, Doctor, Administrator |
| **2.5.**  **Manage Users** | Administrator can browse and filter users based on different criteria: first and last name, email, role and clinical ward. He/She can also edit and delete all personal User Data, as well as create new user using Register use case. | Administrator |
| **2.6.**  **Manage clinical wards** | Administrator can create a new Clinical ward, as well as browse and filter Clinical ward based on different criteria. Also Administrator can choose Clinical ward to manage, and can add or remove Doctors to it, or delete it if necessary. | Administrator |
| **2.7.**  **View medical record** | Patient can browse and filter his/her own previous examination results based on different criteria: result token, type of examination, date, etc.  Doctor can browse and filter previous examination results on concrete patient from his/her own Clinical ward, based on different criteria: result token, type of examination, date, etc.  Administrator can browse and filter all previous examination results. | Patient, Doctor, Administrator |
| **2.8.**  **Add/Edit examination appointment** | Patient can browse types of medical examination, choose one he/she needs, view related laboratory room time schedule and reserve/cancel hour for medical examination only for himself/herself.  Doctor can view his/her Clinical ward time schedule and reserve/cancel hours for medical examinations for his/her patients.  Administrator can browse types of medical examination, choose one he/she desire, view related laboratory room time schedule and reserve/cancel hours for medical examinations for all patients. | Patient, Doctor, Administrator |
| **2.9.**  **Change examination status** | Doctor and Administrator can change the status of all examination. They choose the exact one which is gonna be altered. The system presents the current timeline of the chosen examination. The changes are saved in real time on the server. On examination completion the result is saved by the system. | Doctor, Administrator |
| **2.10.**  **Add/Edit medical examination result** | Doctor can add/edit medical examination result for patient in his/her Clinical ward.  Administrator can add/edit medical examination result for all patient. | Doctor, Administrator |
| **2.11.**  **Monitor examination status** | Patient monitors in real time the status of its examination.  Doctor monitors in real time the status of examinations done in his/her Clinical ward.  Administrator monitors in real time the status of examinations done in all rooms. | Patient, Doctor, Administrator |
| **2.12.**  **View statistic** | All users can view statistic information provided from the Medical examination management system. | All users |
| **2.13.**  **Notification** | Send notification to exact patient about completed examination result. | Patients |

|  |  |  |
| --- | --- | --- |
| 1. **Main Views (Frontend)** | | |
| **View name** | **Brief Descriptions** | **URI** |
| **3.1.**  **Home** | Presents the introductory information about offered medical examinations in the system. Prominently offers ability to register or check most recent examination status and result. | / |
| **3.2.**  **Medical record** | Presents available medical examination results according to User's Role and identity.  Patient can browse and only read his/her all previous examination results.  Doctors have abilities to browse, choose, filter and create all previous medical examination results only for his/her patients according to exact Clinical ward. | /medical-record |
| **3.3.**  **Examination results** | Present and filter completed examination results according user roles. | /results |
| **3.4.**  **Examination result** | Present detailed information for concrete completed examination result. | /result |
| **3.5.**  **User registration** | Presents a view allowing the Anonymous Users to register in Medical examination management system. | /register |
| **3.6.**  **Login** | Presents a view allowing the users to login. | /login |
| **3.7.**  **User Data** | Provides ability to view and edit personal User Data. | /personal |
| **3.8.**  **Clinical wards** | Presents ability for Administrator to browse, filter and manage Clinical wards and assign Doctors to them. | /wards |
| **3.9.**  **Schedule** | Choose room to view schedule and book an appointment. | /schedule |
| **3.10.**  **Upcoming appointments** | Present information about personal upcoming appointments. | /upcoming-appointments |
| **3.11.**  **Dashboard** | Presents in real time the status of examinations and allows Doctor and Administrator to change it, as well as add result when examination is completed. | /dashboard |
| **3.12.**  **Submit result** | Provides form where the Doctors can write down and upload the result for exact examination and submit that all information to the system.  When this step is ready the system notifies the patient whom result belong to. | /submit-result |
| **3.13.**  **Users** | Presents ability to manage Users and their User Data (available for Administrators as described in use case). | /users |
| **3.14.**  **About** | Presents information about the Мedical examination management system, how to use it and contacts of the laboratory. | /about |
| **3.15.**  **Statistic** | Presents information about different statistics provided from Medical examination management system. | /statistic |

|  |  |  |
| --- | --- | --- |
| 1. **API Resources (Node.js Backend)** | | |
| **View name** | **Brief Descriptions** | **URI** |
| **4.1.**  **Users** | GET User Data for all users.  POST new User Data (Id is auto-filled by the system and modified entity is returned as result from POST request) | /api/users |
| **4.2.**  **User** | GET, POST, DELETE User Data for User with specified userId, according to restrictions described in use cases | /api/users/{userId} |
| **4.3.**  **Login** | POST User credentials (e-mail address and password) and receive a valid Security Token to use in subsequent API requests for initializing new session | /api/login |
| **4.4.**  **Logout** | POST a logout request for ending the active session with the system and invalidating the issued Security Token for ending the session | /api/logout |
| **4.5.**  **Clinical wards** | GET Clinical wards and POST new Clinical ward (Id is auto-filled by MEMS and modified entity is returned as result from POST request) | /api/wards |
| **4.6.**  **Clinical ward** | GET, PUT, DELETE Clinical ward (including assigned doctors) for Clinical ward with specified wardId. | /api/wards/{wardId} |
| **4.7.**  **Room schedule** | GET all booked appointments for exact room | /api/rooms/{roomsId}/schedule |
| **4.8. Appointment** | GET, POST, PUT, DELETE concrete appointment to chosen room | /api/rooms/{roomId}/schedule/{appointmentId} |
| **4.9.**  **Upcoming appointments** | GET all upcoming appointments for exact patient | /api/rooms/{userId} |
| **4.10.**  **All results** | GET all results of all patients for Administrator | /api/results |
| **4.11.**  **Results by types** | GET all patients results for concrete examination | /api/room/{roomId}/results |
| **4.12.**  **Results for patient** | GET all examination results of concrete Patient | /api/results/{userId} |
| **4.13.**  **Result** | GET, POST, DELETE result data with specified Id | /api/results/{resultId} |
| **4.14.**  **Rooms** | GET examination rooms and POST new examination room (Id is auto-filled by MEMS and modified entity is returned as result from POST request), one examination room corresponds to exactly one kind of medical examination, performed there | /api/rooms |
| **4.15.**  **Room** | GET, PUT, DELETE information for particular examination room (type of medical examination) by Id. | /api/rooms/{roomId} |
| **4.16.**  **Progresses** | SSE event streaming of Results’ progress on not completed yet Medical examination results according to User's Role and identity security restrictions | /api/progresses |
| **4.17.**  **Progress** | GET, POST current status of the examination | /api/progresses/{progressId} |
| **4.18.**  **Statistic** | GET statistic measurements | /api/statistic |