|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | **Web Application & Service Development with Spring Framework v5** **- 2020** | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | undefined | | online |

|  |  |
| --- | --- |
| Project name | Accounting made easy |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| Accounting made easy is an online platform which makes the sharing of documents easy in this hard times. People can send/receive documents thousands of kilometers away. 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 Company/Accountant/Company user to monitor documents in real time. The main user roles (actors in UML) are:  • *Anonymous User* – can only view the information pages/login/register  • Admin - everything  • *Company* – can view/delete/add documents, edit/remove/add company use  rs and accountants  • *Company user* – can view/delete/add only the documents which he uploaded,  • *Accountant*– can manage documents (create, edit user documents and delete) for given company. And can remove companies from its list. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Register** | *2 type registrations:*  *2.1 Company registration*  *Fields: email, name, Bulstat and password*  *2.2 User registration – 2 types:*  *2.2.1 Worker in a company – name, email, company name and password*  *2.2.2 Accountant – name, email and password*  *-счетоводна къща с няколко счет.*  *- recommendation matchmaking* | *Anonymous User* |
| * 1. **Login** | *Same as registration – choose type* | *Anonymous User* |
| * 1. **Change User Data** | *Change personal info (name,email)* | *Company User, Accountant, Company* |
| * 1. **Manage Users** | *Companies* can browse and filter company users based on different criteria: first and last name, email.  Set/Remove accountant | *Company* |
| * 1. **View documents** | *Company user is with restricted permissions* | *Company, Company user, Accountant* |
| * 1. **Add/Delete Documents** | Add/Delete documents – company  Delete documents – accountant  Add/Delete documents – company user. Only the documents which he uploaded | *Company, Company user, Accountant* |
| * 1. **See documents** | *The accountant can see documents for all. Company too. And company user can see only the documents which he uploaded* | *Accountant, Company user, Company* |
| * 1. **Browse Company Documents** | *The accountant can browse company documents if he/she is the company accountant. Same for company user* | *Accountant, Company, Company user* |
| * 1. **Find company** | *Apply for a company* | *Accountant* |
| **2.10 Find accountant** | *Apply for an accountant* | *Company* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Home** | Presents the main page with documents | / |
| * 1. **Profile** | Presents the user/company data. It gives the ability to remove/edit users depending on the type of ROLE. And presents ability to edit personal info | */profile* |
| * 1. **Upload document** | Presents a view to upload the new document | */upload* |
| * 1. **Login** | Presents a view allowing the users/companies to login. | */login* |
| * 1. **User Registration** | Presents a view allowing the *Anonymous Users* to register with different type of roles(Company, Company user, Accountant) | */register* |
| * 1. **Company Data** | Presents ability to browse, filter and manage *Users/Accountants* | */company* |
| * 1. **Dashboard** | Presents the statistics for the company/users | */dashboard* |
| * 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 project and his owner. | */about* |
| **3.10 Accountants** | Presents a view with available accountants | */accountants* |
| **3.11 Companies** | Presents a view with available companies | */companies* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Companies** | GET All Companies Data and all its users/accountants, POST creates new Company | */api/company* |
| * 1. **Apply as an Accountant** | POST apply for a job as an accountant at a company | */api/company/{companyId}/apply* |
| * 1. **Apply for an accountant** | POST request an accountant | */api/accountants/{accountantId}* |
| * 1. **Company** | GET Company Data and all its users/accountants, PUT edits the given company | */api/company/{companyId}* |
| * 1. **Users** | GET *User Data* for all users for given type, and POST new *User Data*. Available only for *Administrators*. | */api/users?type=* |
| * 1. **User** | GET, PUT, DELETE *User Data* for *User* with specified *userId*, according to restrictions described in UCs. Available only for *Administrators* | */api/users/{userId}* |
| * 1. **Login** | POST *User Credentials* (e-mail address,type 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. **Documents** | GET all documents (according to *User's Role* and identity) and POST new *Document*. | */api /documents* |
| * 1. **Document** | GET, PUT, DELETE *File Data*  for *Document* with specified *documentID*. | */api/documents/{documentID}* |