Assignment 1	Project Summary
Course	**Web Application & Service Development with Spring Framework v5 - 2020**

Project author	
№ Pseudonym	Face-to-face/ online
1 Linna	online

Project name	Website for anecdotes

1. Short project description (Business needs and system features)

An anecdote is a short story about a real person or event, usually serving to make the listeners laugh or ponder over a topic. Generally, the anecdote will relate to the subject matter that the group of people is discussing. Sometimes telling a story just makes people laugh or brightens the mood. So it's good to have a web site where we can read jokes and have fun.

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**.

The main user roles (actors in UML) are:

•   *Anonymous User* – can only view the information pages.

•   *Regular User*(extends *Registered User*) – can create, edit and delete their own anecdotes, comment other anecdotes and view information pages.

•   *Administrator* (extends *Registered User*) – can manage (edit user data and delete) all *Registered Users, create, edit and delete anecdote`s categories* and manage (edit and delete) all anecdotes.

2. Main Use Cases / Scenarios		
Use case name	**Brief Descriptions**	**Actors Involved**
2.1. Browse information	The *User* can browse the information views (Home, Anecdote`s Categories, Anecdotes View, About).	All users
2.2. Register	*Anonymous User* can register in the system by providing a valid e-mail address, first and last name, and choosing password.	*Anonymous User*
2.3. Change User Data	*Registered User* can view and edit her personal *User Data*. *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. *Administrator* can choose a *User* to manage, and can manage the chosen User - edit or delete.	*Administrator*
2.5. Manage Anecdote`s Categories	*Administrator* can create, edit and delete anecdote`s categories.	*Administrator*

2.6. Manage Anecdotes	*Regular User* can create, edit and delete their own anecdotes. *Administrator* can edit and delete all anecdotes.(Reasons: repetitive, cynical, vulgar)	*Regular User, Administrator*
2.7. Add comments	*Registered User* can add comments to anecdotes.	*Registered User, Administrator*

3. Main Views (Frontend)		
View name	**Brief Descriptions**	**URI**
3.1. Home	Presents the introductory information for the purpose of the system as well as detailed instructions on how to start using it. Prominently offers ability to register.	/home
3.2. User Registration	Presents a view allowing the *Anonymous Users* to register.	/register
3.3. Login	Presents a view allowing the users to login.	/login
3.4. User Data	Provides ability to view and edit personal *User Data*.	/personal
3.5. Users	Presents ability to manage (CRUD) Users and their User Data (available for *Administrators* only, as described in UCs).	/users
3.6. Anecdote`s Categories	Provides ability to view all anecdote`s categories	/anecdotes-categories
3.7. Categories	Presents ability to manage (CRUD) Anecdote`s Categories (available for *Administrators* only, as described in UCs).	/categories
3.8. Anecdotes View	Provides ability to view all anecdotes by category.	/anecdotes-view

3.9. Anecdotes	Presents ability to manage (CRUD) Anecdotes by category (available for *Administrators* only, as described in UCs).	*/anecdotes*
3.10. About	Presents information about the project and his owner.	*/about*

4. API Resources (Backend)		
View name	**Brief Descriptions**	**URI**
4.1. Users	GET *User Data* for all users, and POST new *User Data* (Id is auto-filled by *OKTS* 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 Token* to use in subsequent API requests.	*/api/login*
4.4. Logout	POST a logout request for ending the active session with *OKTS,* and invalidating the issued *Security Token*.	*/api/logout*
4.5. Anecdote`s Categories	GET *Anecdote`s Category*, and POST new *Anecdote`s Category* (Id is auto-filled by *OKTS* and modified entity is returned as result from POST request). Available only for *Administrators*.	*/api/anecdotes-categories*
4.6. Anecdote`s Category	GET, PUT, DELETE *Anecdote`s Category* with specified *groupId*.	*/api/anecdotes-categories/{categoryId}*
4.7. Anecdotes	GET Anecdote, and POST new Anecdote (Id is auto-filled by *OKTS* and modified entity is returned as result from POST request).	*/api/anecdotes*
4.8. Anecdote	GET, PUT, DELETE Anecdote with specified *testId*.	*/api/anecdotes/{anecdoteId}*