Note Web App

1. Describe high level design Show the main note app components and the logical interactions that will fulfill the requirements.

We will start with an architecture based on the client side.
We will produce 2 pages, being the Logging page and the initial page that list all notes in a display grid two by two.
We will produce 1 modal to work with save and edit notes.
We will need an authentication JWT token and a username transiting the api.
Then we will create a backside with rest methods, using MVC architecture to expose our web service to our client side to consume.

2. Web App UI Provide a wireframe design of the note web app that will fulfill the requirements. Consider what UI components are required and how these interact with the other components. What (if any) validation is required?

Login Frame - username, password and an API logo. Authentication will be retrieved from a webservice and maintained in a local store inside the browser.
Initial Frame - a list with all the notes divided in 2 boxes, each containing 2 icons that deletes or edit the note.
Save or Edit frame - a modal that contains all the data to create or edit a note.

3. Data Model Describe how a note will be modeled consider the required properties

The data model for the note app will have 2 tables, starting with note table with the following fields:
integer id,
varchar note text,
date last updated,
integer login_id(foreign key),
and login table with the following fields:
integer id,
varchar username,
varchar password

4. Restful API Describe the Restful API required to fulfill the note app. how would the web app get the user's notes? how would the web app save a user note? what are the URL for the note resource(s)? and verbs to expose the actions?

We will serve all the endpoints to retrieve/save or edit data so that we can use restful services to perform *GET*, *POST*, *PUT* and *DELETE* functions.

If the user selects the delete icon, the api will call a function that will *DELETE* the note from our database
The URL should be something like: api/note/{id} passing the authorization token in the headers

If the user selects the edit icon, a modal frame will be opened to perform *PUT* action.
The URL should be something like: api/note/{id} passing the authorization token in the headers

If the user selects the new button, a modal frame will be opened to perform *POST* action.
The URL should be something like: api/note/ passing the authorization token in the headers

For *GET* purpose we are going to retrieve all the notes from a specific login id and the URL will be: api/notes/{id} passing the authorization token in the headers.
For *GET* purpose we are going to retrieve a specific note to serve the modal page using the URL: api/note/{id}

5. Web Server Describe how the webserver implements that Restful API: consider how each action will be implemented what (if any) business logic is required? how are the notes saved?

We are going to implement a spring boot rest method to serve all the REST components to the client side.

in PUT action we are going to find the note in the database, then save the new edited note if the id from the login matches.
in POST action we are going to save in the database a new note.
in DELETE action we are going to find the note in the database and delete it if the id from the login matches.
in GET action we are going to get all the notes from the specific login id

in GET action we are going to get a specific note from specific login id