**Notes App**

* **Requirement Analysis:**
* Create note
* Update note
* Delete note
* **Technology used:**
* Frontend-React JS
* Backend –Nodejs
* Database –Mongodb
* **Database:**

**+------------------+ +------------------------------+**

**| notes | | users |**

**+------------------+ +-------------------------------+**

**| note\_id (PK) | user\_id (PK) |**

**| title | | username |**

**| content | | email |**

**| created\_at | | password |**

**| tags | | first\_name |**

**| attachments| | last\_name |**

**+------------------+ | date\_of\_birth |**

**| profile\_image |**

**| created\_at |**

**| last\_login\_at |**

**| is\_active |**

**| roles |**

**+------------------------------+**

* **Outline:**

**Actor:**

User: Interacts with the application to manage notes.

Use Cases:

**Create Note:**

Description: Allows users to create a new note.

Actors: User

Preconditions: User is authenticated and authorized.

Flow of Events:

User selects the option to create a new note.

User fills in the title, content, tags, and attachments for the note.

User submits the note creation form.

System validates the input data.

System saves the new note in the database.

System notifies the user of successful note creation.

**Update Note:**

Description: Enables users to modify an existing note.

Actors: User

Preconditions: User is authenticated and authorized.

Flow of Events:

User selects the note they want to update.

User edits the title, content, tags, or attachments of the note.

User saves the changes.

System validates the updated data.

System updates the note in the database.

System notifies the user of successful note update.

**Delete Note:**

Description: Allows users to remove a note from the system.

Actors: User

Preconditions: User is authenticated and authorized.

Flow of Events:

User selects the note they want to delete.

User confirms the deletion action.

System removes the note from the database.

System notifies the user of successful note deletion.

Extensions:

**Error Handling:**

Handle scenarios such as invalid input, failed database operations, etc.

Authentication and Authorization: Ensure that only authenticated and authorized users can perform note-related actions.

User Profile Management: Additional use cases for managing user profiles, such as updating profile information or uploading a profile image.

* **Class Diagram:**

[**https://drive.google.com/file/d/1cwlyQiMBeDif6W4n8ubDvCrxwe5E8Xqm/view?usp=drive\_link**](https://drive.google.com/file/d/1cwlyQiMBeDif6W4n8ubDvCrxwe5E8Xqm/view?usp=drive_link)

* **Sequence diagram:**

[**https://drive.google.com/file/d/1qcPOYOxuiPtDNeFL2Yo8Ri7YDGINcw9U/view?usp=drive\_link**](https://drive.google.com/file/d/1qcPOYOxuiPtDNeFL2Yo8Ri7YDGINcw9U/view?usp=drive_link)

* **Flow chart:**

[**https://drive.google.com/file/d/1\_gJaMJM\_GH6ztJrdTf9MZMLowUye6Dnp/view?usp=drive\_link**](https://drive.google.com/file/d/1_gJaMJM_GH6ztJrdTf9MZMLowUye6Dnp/view?usp=drive_link)