1.4 Problem Statement

We are developing a mobile application designed to facilitate the sharing and exchange of classroom notes among students. The app will enable users to upload, categorize, search, filter, and access notes from various courses and institutions, promoting collaborative learning and academic support. Targeting university and high school students, the app will help users acquire notes they missed or need for classes they're not currently enrolled in. This will be especially beneficial for revising prior information. Unlike existing solutions such as group chats, Quizlet, iCollege/Canvas, and Google Drive, our app aims to provide a streamlined, centralized platform specific to this local university or school, enhancing accessibility and organization.

Key Enhancements:

o Search and Filtering Functionality: Users can efficiently find notes using keyword searches, subject filters, institution filters, and rating-based sorting.

o Offline Access to Notes: Users can download notes for offline viewing, ensuring accessibility without an internet connection.

o Rating and Peer Review System: Users can rate and review submitted notes to highlight high-quality contributions and improve the overall learning experience.

o Private and Request-Based Note Sharing: Notes can be set as private, requiring users to send a request to access restricted materials, adding a layer of control and exclusivity for note owners.

Target Customers:

The primary users of the app will be students, professors, and teaching assistants. Students will be able to submit notes from different universities, rate user-submitted notes, and sort notes by rating, upload date, or subject. Professors and teaching assistants can also contribute verified academic materials. Users will have access to publicly available notes, while private notes will require a request-based access system.

Top-Level Objectives:

o Enable users to upload, browse, search, and filter notes by subject, teacher, and rating.

o Include Student login and Professor login pages for role-based access.

o Integrate a rating and peer review system to improve note quality.

o Allow users to download notes for offline access.

o Implement private note-sharing with a request-based access mechanism.

o Structure the app to fit institutional needs and optimize collaboration among students.

Technical Details:

The app will be developed using platforms like Android Studio for app design and structure, and SQL or Firestore for data storage and access. The technical challenges presented by this project will require various techniques to build and deploy the app efficiently, making it an interesting endeavor from a technical perspective. The system will incorporate robust search, filtering, and data management capabilities to provide a seamless user experience.

1.5 System Requirements

1.5.1 Context Diagram

Description:

Student users can upload, download, rate, and request access to notes, enabling collaborative learning. Professors and teaching assistants can review and manage notes, as well as upload their materials for students to access. System administrators oversee user management and content moderation to ensure appropriate use of the platform. The authentication system secures user accounts

and prevents unauthorized access. The database stores notes, metadata, and user profiles, ensuring efficient organization and retrieval. A notification system keeps users informed about important updates, including access requests and new uploads. Lastly, the file storage system allows users to save notes and access them offline.

1.5.2 Activity Diagram

Description:

o In the app, both professors and students will be able to create their accounts.

o Once logged in the student will have access to several functionalities.

o On a student account, students will be able to take or upload notes in the app and they can share notes with their peers. They can also review each other's notes.

o Once logged in the professors will have access to their own set of functionalities.

o On a professor account, they will be able to record or upload notes/recordings in the app and they can share and review student notes .

2.0 REQUIREMENTS

2.1 Use Cases

Use Case # 1:

Use Case Name: Log-in

Actors: [students, professors, and database]

Description:

o The user will be given the login page, as the first step to access our app.

o The user will enter an email and password.

o After the inputs are complete, the user will have to submit the form.

o If the user is registered and typed everything correctly, the user will get to our

home page.

o If unsuccessful, the system will inform the user to try again.

o The database will do the authentication, to validate if the user is registered.

o If the user is not registered, they can register using the link provided on the page.

Exception: if the user tries more than 5 times, the email associated with his account will be locked. the user will have to wait for a certain amount of time to retry again.

Alternate Path: There is only one path in this case. The user will not access the app's features until they successfully log in.

Pre-condition: A valid account must already be available. If not, the user must first register.

Post-condition: Accessing the home page for the NoteX app.

Use Case # 2:

Use Case Name: Share

Actors: [students, professors, and database]

Description:

o The user will have a share icon on the notes page.

o The user can click the share icon to start the sharing process.

o the user will have to select the notes they wish to share.

o After the selection of notes, the user will have the option to select student accounts they wish to share the notes.

o This can be a group sharing where the user selects multiple student accounts.

o the user will then be presented with a share icon which will automatically send the notes to the destination account.

o If unsuccessful, the system will inform the user to try again.

o The database will update the destination accounts by inserting the notes in their accounts and listing who sent the notes.

Exception: The owner of the notes has to permit the usage of his notes if they are

set to private.

Alternate Path: There is only one path in this case.

Pre-condition: Valid accounts for both the sender and the receiver must already be available.

Post-condition: Accessing shared notes.

Use Case # 3:

Use Case Name: Upload

Actors: [students, professors and database]

Description:

o The user will have an upload icon on the homepage.

o The user can click the upload icon to start the uploading process.

o the user will have to select the notes they wish to upload.

o After the selection of notes, the user will have the option to select where they wish to place the notes.

o The user will then be presented with a save button which will automatically save the notes to the destination account.

o If unsuccessful, the system will inform the user to try again.

o The database will update the user account by inserting the notes in their accounts.

Exception: The owner of the notes has to set the notes as public or private.

Alternate Path: There is only one path in this case.

Pre-condition: A valid account of the user must be present.

Post-condition: Accessing the uploaded notes.

Use Case # 4:

Use Case Name: User Account Management

Actors: [System Administrators, Database]

Description:

o The administrator logs in using a valid account.

o The administrator navigates to the User Management section.

o The administrator views a list of users.

o The administrator selects a user account to manage.

o The administrator has options to approve, deactivate, or modify user accounts.

o The system updates the database based on the action performed.

o The administrator receives a confirmation of the action taken.

Exception: If the system encounters an error, it notifies the administrator.

Alternate Path: There is only one path in this case

Pre-condition: A valid administrator account must be present.

Post-condition: The user account details are updated in the database.

Use Case # 5:

Use Case Name: Rating and Reviewing Notes

Actors: [Students, Database]

Description:

o The user logs in to the platform using a valid account.

o The user views a specific set of notes

o The user clicks the Rate and Review option

o The user inputs a rating and writes a review

o The user submits the review

o The system updates the database with the new rating and review

o The system recalculates the average rating for the notes

Exception: If the user submits inappropriate content, the review may be flagged for moderation

Alternate Path: There is only one path in this case

Precondition: A valid user account must be present

Post-condition: The note's average rating and review list are updated.

Use Case # 6:

Use Case Name: Managing Uploaded Content

Actors: [Professors, Teaching Assistants, Database]

Description:

o The professor or teaching assistant logs in using a valid account.

o The user navigates to the Manage Content section

o The user selects specific notes to review

o The user has options to edit, delete, or approve the notes.

o The user confirms the action taken (deletion)

o The system updates the database on the action performed.

Exception: If the system encounters an error during moderation it notifies the user.

Alternate Path: There is only one path in this case.

Pre-condition: The user must have a valid account with administrative privileges.

Post-condition: The changes to the notes are reflected in the database.

Use Case # 7:

Use Case Name: Exam Preparation

Actors: [Student, System]

Description:

o The user can access curated study guides and past exam notes to help with exam preparation.

o The system categorizes study materials by subject and difficulty level.

o Users can filter and search for materials relevant to their exams.

o Downloadable content ensures offline access for better flexibility.

o Users can contribute their study guides for peer collaboration.

Exception: If the database is down or a file is corrupted, the system displays an error message and suggests re-uploading.

Alternate Path: If no relevant study materials are found, the system suggests related content. If study materials are outdated, users can request updated versions from peers.

Pre-Condition: The user is logged in, and the study materials are in the system.

Post-Condition: User successfully accesses relevant study materials.

Use Case # 8:

Use Case Name: Multi-Format System

Actors: [Student, System]

Description:

o Users can upload and access notes in different formats such as PDFs, images, handwritten notes, and audio summaries.

o The system ensures seamless compatibility and provides a preview option.

o Users can convert files between formats (e.g., text-to-audio, image-to-text).

o Uploaded files are securely stored and accessible for future reference.

o The system ensures efficient file retrieval and categorization for easy access.

Exception: If an upload fails due to a server issue or the file is corrupted, the system prompts a retry and suggests re-uploading.

Alternate Path: If the file format is unsupported or too large, the system suggests an alternative format or prompts the user to compress or split the file.

Pre-Condition: The user is logged in, and the uploaded file meets system size and format criteria.

Post-Condition: The note is successfully uploaded, converted if necessary, and accessible in multiple formats.

Use Case # 9:

Use Case Name: Flashcards and Quizzes

Actors: [Student, System]

Description:

o Users can convert their notes into flashcards for active studying.

o The system can auto-generate quizzes based on uploaded notes.

o Users can manually edit or add questions to customize their quizzes.

o A progress tracker helps students monitor their performance over time.

o Users can share flashcards with peers for collaborative learning.

Exception: If the system encounters an error or the user attempts to create flashcards from unsupported content, an alert notifies the user and displays a warning.

Alternate Path: If automatic conversion fails or a flashcard set is incomplete, the system prompts the user to create flashcards manually and suggests additional content.

Pre-Condition: The user has existing notes in the system, and the system supports flashcards and quiz generation.

Post-Condition: Flashcards or quizzes are successfully generated from the notes, available for review, and ready for interactive studying.

Use Case #10:

Use Case Name: Request Notes

Actors: [Students, Professors, System]

Description:

o Users can request specific notes from peers or professors.

o The system allows users to post requests detailing the subject, topic, and any additional requirements.

o Requested users receive a notification and can upload the requested notes.

o The system sends a confirmation message when the requested notes are uploaded.

Exception: If no one responds to the request within a set timeframe, the request expires.

Alternate Path: If a request remains unanswered, the system suggests similar available notes.

Pre-condition: The requester must be logged in and specify clear requirements.

Post-condition: Requested notes are uploaded and accessible.

Use Case #11:

Use Case Name: Collaborative Note Editing

Actors: [Students, Professors, System]

Description:

o Users can invite others to collaborate on notes.

o Real-time editing is enabled, with version history tracking.

o Different permissions can be assigned to collaborators (edit/view).

Exception: If an unauthorized user tries to edit, they receive an access error message.

Alternate Path: If real-time editing is unavailable, the system allows users to edit in offline mode and sync later.

Pre-condition: Users must be logged in and have edit permissions.

Post-condition: Notes are collaboratively edited and updated in real time.

Use Case #12:

Use Case Name: Bookmarking Notes

Actors: [Students, System]

Description:

o Users can bookmark frequently accessed notes.

o Bookmarked notes appear in a dedicated section for quick access.

Exception: If the system experiences a synchronization error, bookmarks may be temporarily unavailable.

Alternate Path: If notes are deleted by the owner, bookmarks are removed automatically.

Pre-condition: Users must be logged in.

Post-condition: Bookmarked notes are saved and easily accessible.

Use Case #13:

Use Case Name: AI-Powered Note Summarization

Actors: [Students, System]

Description:

o Users can generate summaries of long notes.

o The system uses AI to extract key points while preserving essential details, while preserving details.

o Users can edit or refine the generated summary.

Exception: If a note is too short, the system may be unable to generate a summary.

Alternate Path: If AI summarization fails, the system suggests manual summarization.

Pre-condition: Users must be logged in and have access to notes.

Post-condition: Summarized notes are generated and stored for future reference.

Use Case #14:

Use Case Name: Offline Note Access

Actors: [Students, System]

Description:

o Users can download notes for offline viewing.

o The system encrypts downloaded files for security.

Exception: If storage is full, the system notifies the user and prevents additional downloads.

Alternate Path: If a user tries to download private notes without permission, access is denied.

Pre-condition: Users must be logged in and have download permissions.

Post-condition: Notes are saved and accessible offline.

Use Case #15:

Use Case Name: Notifications & Alerts

Actors: [Students, System]

Description:

o Users receive notifications for shared notes, requests, comments, and system updates.

o Notifications appear on the dashboard and via email (if enabled).

Exception: If notifications fail to deliver, users can check manually in their inbox.

Alternate Path: Users can enable or disable notifications in settings.

Pre-condition: The user must be logged in.

Post-condition: Notifications are delivered and stored in the system.

2.2 Requirements

Requirement 1:

Use Case 1:

Introduction: easy look and feel for the 2-input page from the user

Destination: Home page

Rationale: Login page for the user to log in and start uploading, browsing, and sharing notes.

Input: Email and password

Requirement Description: The user should be able to log in and this is possible

only when all requirements are met.

o All Fields must be completed.

o Requirements from the system are met.

o Database authentication must be met.

Output: Routed to our Home Page.

Requirement 2:

Use Case 2:

Introduction: The share icon will be placed on the top right of the screen when notes are selected.

Destination: receiver account

Rationale: The user will have the option to share the notes they own.

Input: Notes, Student Email(s), Permission (full access, keep private)

Requirement Description: The user should be able to share their notes and this is possible only when all of the following requirements are met.

o Both parties must have a valid account.

o The owner must specify a valid student account.

o The owner must specify the permissions on the notes if they are private.

o If the notes are public the students can share only links pointing to the notes.

Output: Dialog box showing if the notes were sent successfully. Routed back to the notes.

Requirement 3:

Use Case 3:

Introduction: The upload icon will be placed on the top right of the screen on the homepage.

Destination: Uploaded notes.

Rationale: The user will have the option to upload their notes to their NoteX account.

Input: Notes.

Requirement Description: The user should be able to upload their notes and this is possible only when all of the following requirements are met.

o The user must have a valid account.

o The user must set the notes as public/private.

o If the notes are public then the student must be informed that their notes will be viewed by everyone on the platform.

Output: Dialog box showing if the notes were uploaded successfully. Routed back to the homepage.

Requirement 4:

Use Case 4:

Introduction: The authentication system will be integrated into the platform to verify and authorize user accounts, enabling secure access and preventing unauthorized users from interacting with the platform.

Destination: User authentication and authorization.

Rationale: The system needs to ensure that only registered users with valid accounts can upload, download, and manage notes. Different user roles (e.g., students, professors, administrators) should have specific permissions based on their roles.

Input: User login credentials (username and password)

Requirement Description: The system must provide secure authentication for all users, supporting role-based access control for:

o Students: Upload, download, rate, and review notes.

o Professors and TAs: Upload, edit, and manage notes.

o Administrators: Oversee user management and content moderation.

Output: A successful login grants access to the platform's features. Unauthorized attempts or failed logins display an error message and prompt the user to try again.

Requirement 5:

Use Case 5:

Introduction: A notification system will be implemented to inform users of important updates including access requests, new uploads, and administrative announcements

Destination: User notifications section.

Rationale: The system needs to keep users informed of actions that require their attention, such as pending access requests, status updates, or new materials uploaded by others.

Input: System-triggered notifications based on user actions.

Requirement Description: The notification system must:

o Notify users when they receive access requests for private notes.

o Notify users when their access requests have been approved or denied.

o Notify users when new materials are uploaded to the platform.

o Display notifications prominently on the homepage or in a dedicated notifications section.

Output: Notification messages are displayed to users, either as pop-up alerts or as messages in the notifications section.

Requirement 6:

Use Case 6:

Introduction: A data backup and recovery system will be implemented to ensure that user-generated content, such as uploaded notes, metadata, and account information, is protected from data loss or corruption.

Destination: Backup storage system and recovery mechanism.

Rationale: The system must ensure that critical user data is securely backed up and can be recovered in the event of a system failure, data corruption, or accidental deletion.

Input: Scheduled or triggered data backup processes.

Requirement Description: The data backup and recovery system must:

o Perform regular automated backups of all user-uploaded content, metadata, and account details.

o Store backup files in a secure and separate location from the primary database.

o Include a recovery mechanism to restore data in case of system failure or accidental deletion.

o Log backup activities and recovery events for audit purposes.

Output: A secure backup of user data that can be used to restore lost or corrupted information, with a confirmation message indicating successful recovery if needed.

Requirement 7:

Use Case 7:

Introduction: The system must provide structured study materials for exam preparation, ensuring efficient organization and accessibility.

Destination: Study Materials Section

Rationale: Users need access to past exam notes and study guides for effective exam preparation.

Input: Selected subject and topic

Requirement Description:

o Users should be able to browse categorized study materials.

o Search filters must allow sorting by subject, topic, and difficulty level.

o Users can download and request specific study materials.

Output: Display of study materials and downloadable files.

Requirement 8:

Use Case 8:

Introduction: The system must support multiple file formats for diverse note taking methods.

Destination: User's Uploaded Files

Rationale: Users should be able to upload and access notes in various formats for flexibility.

Input: File upload (PDF, DOCX, PNG, MP3, etc.)

Requirement Description:

o Users should be able to upload and preview multiple file formats.

o The system must support conversions (e.g., text-to-audio, image-to-text).

o Uploaded files must be securely stored and efficiently retrieved.

Output: Successfully uploaded and accessible notes in multiple formats.

Requirement 9:

Use Case 9:

Introduction: The system should allow students to generate flashcards and quizzes from notes to enhance learning.

Destination: Flashcards and Quiz Section

Rationale: Flashcards and quizzes make revision interactive and efficient.

Input: Selected notes or manually entered flashcard content

Requirement Description:

o Users should be able to auto-generate or manually create flashcards.

o The system should provide quiz-based learning from the user's notes.

o Users can track progress and revisit previous quiz attempts.

Output: Generated flashcards and quizzes available for interactive study.

Requirement 10:

Use Case 10:

Introduction: The system must allow users to request specific notes from peers or professors.

Destination: Requested Notes Section

Rationale: Users may need specific study materials that are not publicly available.

Input: Subject, topic, additional request details

Requirement Description:

o Users should be able to post a request for specific notes.

o The system should notify the requested user(s) about the note request.

o Users should be able to upload requested notes in response.

o A confirmation notification must be sent when notes are uploaded.

Output: Notification sent to the requester and uploaded notes accessible.

Requirement 11:

Use Case 11:

Introduction: Users should be able to collaborate and edit notes in real time.

Destination: Shared Notes Section

Rationale: Real-time collaboration enables multiple users to work on the same notes.

Input: Notes, collaborator emails, permission settings

Requirement Description:

o Users should be able to invite others for collaborative editing.

o A version history should track all changes.

o Real-time synchronization must be supported.

o Users must have the ability to set permissions (view-only, edit, comment)

Output: Notes are collaboratively edited and updated in real time.

Requirement 12:

Use Case 12:

Introduction: Users should have the ability to bookmark frequently accessed notes.

Destination: Bookmarked Notes Section

Rationale: Users need quick access to important study materials.

Input: Selected notes for bookmarking

Requirement Description:

o Users should be able to bookmark notes for future reference.

o The system must display a dedicated section for bookmarked notes.

o Bookmarked notes should be synchronized across devices.

Output: Notes are successfully bookmarked and displayed in the bookmarked section.

Requirement 13:

Use Case 13:

Introduction: The system should generate AI-powered summaries for lengthy notes.

Destination: Summarized Notes Section

Rationale: AI-generated summaries help users quickly understand key points.

Input: Selected notes

Requirement Description:

o Users must be able to generate AI-powered summaries.

o The system should extract essential points while maintaining accuracy.

o Users should be able to refine AI-generated summaries if needed.

Output: A summarized version of the notes is available for review and study.

Requirement 14:

Use Case 14:

Introduction: Users should have the ability to download notes for offline access.

Destination: Local Storage (Offline Access)

Rationale: Offline access ensures students can study without an internet connection.

Input: Selected notes for download

Requirement Description:

o Users must be able to download notes for offline use.

o The system should encrypt downloaded files for security.

o Notes should remain accessible within the app without an internet connection.

Output: Notes are stored locally and accessible offline.

Requirement 15:

Use Case 15:

Introduction: The system should notify users about updates, shared notes, and requests.

Destination: Notifications Section

Rationale: Users need real-time alerts about important actions.

Input: System-generated notifications based on user activity and system updates

Requirement Description:

o Users must receive notifications for shared notes, comments, and updates.

o Notifications should be displayed in the app and optionally via email.

o Users should be able to customize notification preferences.

o The system should provide a centralized notification center for easy management

o Users should be able to mark notifications as read or dismiss them as needed

Output: Notifications are successfully displayed and stored for future reference.