Project Title:

Quran Explorer

Submit to:

Mam Anza

Submit by:

M. Uzair Shahbaz

College Roll #:

21302

University Roll #:

411702

Course Title:

Introduction to Software Engineering

Course Code:

SWE-401

Department:

BS Computer Science (4th Semester) "Morning"

College:

Government Graduate College, Samanabad, Faisalabad.

Topic:

“Quran Explorer”

Introduction:

In the Quran Explorer app, you will have access to a vast collection of Quranic verses, recitations, and translations. The app allows you to explore the divine verses of the Quran in multiple languages, enabling a deeper understanding of its profound teachings.

Verse Exploration:

With Quran Explorer, you can easily navigate through the chapters (Surahs) and verses (Ayahs) of the Quran. You can browse through the entire Quran or select specific chapters to study. Each verse is organized properly, providing you with the ability to explore the text in a structured manner.

Recitation:

The app offers an extensive library of recitations by renowned reciters from around the world. You can listen to the beautiful and melodious recitations. The recitations are available in various recitation styles (Qira'at) to meet to different preferences.

Translation:

Quran Explorer provides translations of the Quran in multiple languages, allowing you to comprehend the meanings of the verses. These translations help bridge the language barrier, making the profound message of the Quran accessible to individuals from diverse linguistic backgrounds.

With Quran Explorer, you can immerse yourself in the beauty, guidance, and wisdom of the Quran, unlocking its transformative power and enriching your spiritual journey.

1) Functional Requirements

Description:

The user who wants to search recite or translate any chapter, its some part or even single verse will register first and easily find them. This feature will help him to use the app conveniently.

REQ-01: User can browse the desired verse.

REQ-02: User can search the desired verse.

REQ-03: User can create account.

REQ-04: User can login or resister himself.

REQ-05: User can select verse.

REQ-06: User can hear recitation of verse.

REQ-07: User can translate verse.

REQ-08: User can add verses to favorites.

REQ-09: User can also add to watch later.

REQ-10: User can see time of salaah.

REQ-11: User can recite daily routine prayers.

REQ-12: User can use tasbeeh counter.

REQ-13: User can see Islamic date, month and year.

REQ-14: User can find explanations and interpretations of the Quranic verses from renowned scholars or Tafsir books to help users gain a deeper understanding of the text.

REQ-15: User can share specific verses or passages through social media, messaging apps, or email.

REQ-16: User can set app font size, theme, reciter preference, and translation options.

REQ-17: Users could have the option to set reminders for daily Quranic readings,

REQ-18: User can memorise Quranic verses, using memorization tracker, recitation repetition,etc.

REQ-19: User can use concordance feature that allows users to search for specific words or phrases across the entire Quran.

REQ-20: User can customize the recitation experience, such as adjusting the reciter's speed, voice, or audio playback settings.

REQ-21: User can access detailed information about each surah, including its name, revelation order, number of verses, themes, and background context.

REQ-22: Administration stores the search history of User.

REQ-23: Administration is able to manage app database.

REQ-24: Administration will be able to add new features by collaborates with the development team

REQ-25: Administration will be able to provide friendly enviornment.

REQ-26: Administration will handle complains/feedbacks given by the User.

REQ-27: The administrator may be responsible for promoting the app and increasing its user base.

REQ-28: The administrator oversees user accounts and user-related activities within the app.

2) Non-Functional Requirements

- **Performance:**

  The app should be responsive and provide quick search results, smooth scrolling, and fast loading times, ensuring a seamless user experience even on slower devices or network connections.

- **Usability:**

  The app should have a user-friendly interface with intuitive navigation, clear labels, and easily understandable features. It should be accessible to users of varying technological proficiency.

- **Compatibility:**

  The app should be compatible with multiple platforms, such as iOS, Android, and web browsers, ensuring that users can access it on their preferred devices.

- **Security:**

  The app should prioritize the security and privacy of user data. It should employ appropriate security measures to protect user information and ensure secure transactions, if applicable.

- **Reliability:**

  The app should be stable and reliable, minimizing crashes, errors, or unexpected behavior. It should undergo rigorous testing to ensure its functionality under various conditions.

- **Scalability:**

  The app should be designed to handle a potentially large user base and be capable of scaling its resources to accommodate increased traffic or usage without significant performance degradation.

- **Accessibility:**

  The app should adhere to accessibility standards, ensuring that individuals with disabilities can use it effectively. It should support features such as screen readers, text resizing, and color contrast options.

- **Offline Functionality:**

  The app could provide offline access to previously accessed verses, translations, or study materials, allowing users to continue their exploration of the Quran even without an internet connection.

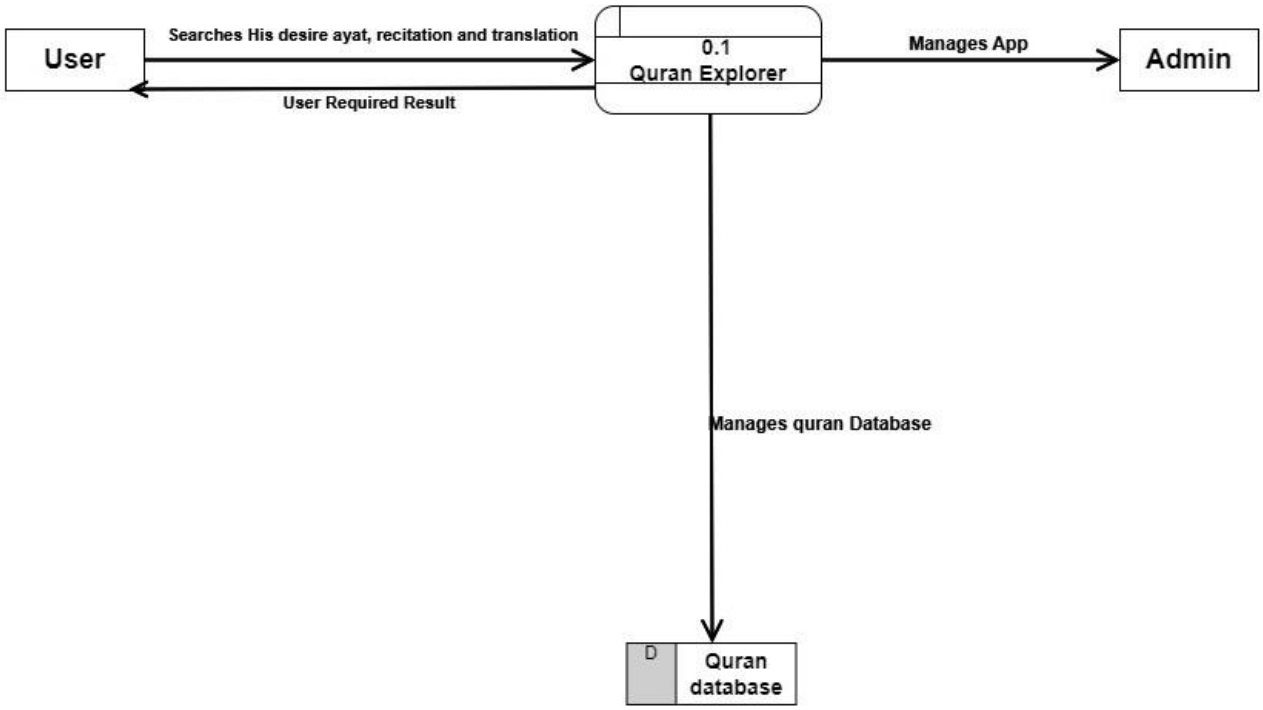
- **Updates and Maintenance:**

  The app should have a well-defined plan for updates, bug fixes, and ongoing maintenance to ensure that it remains up-to-date, secure, and compatible with evolving technologies.

Data Flow Diagrams:

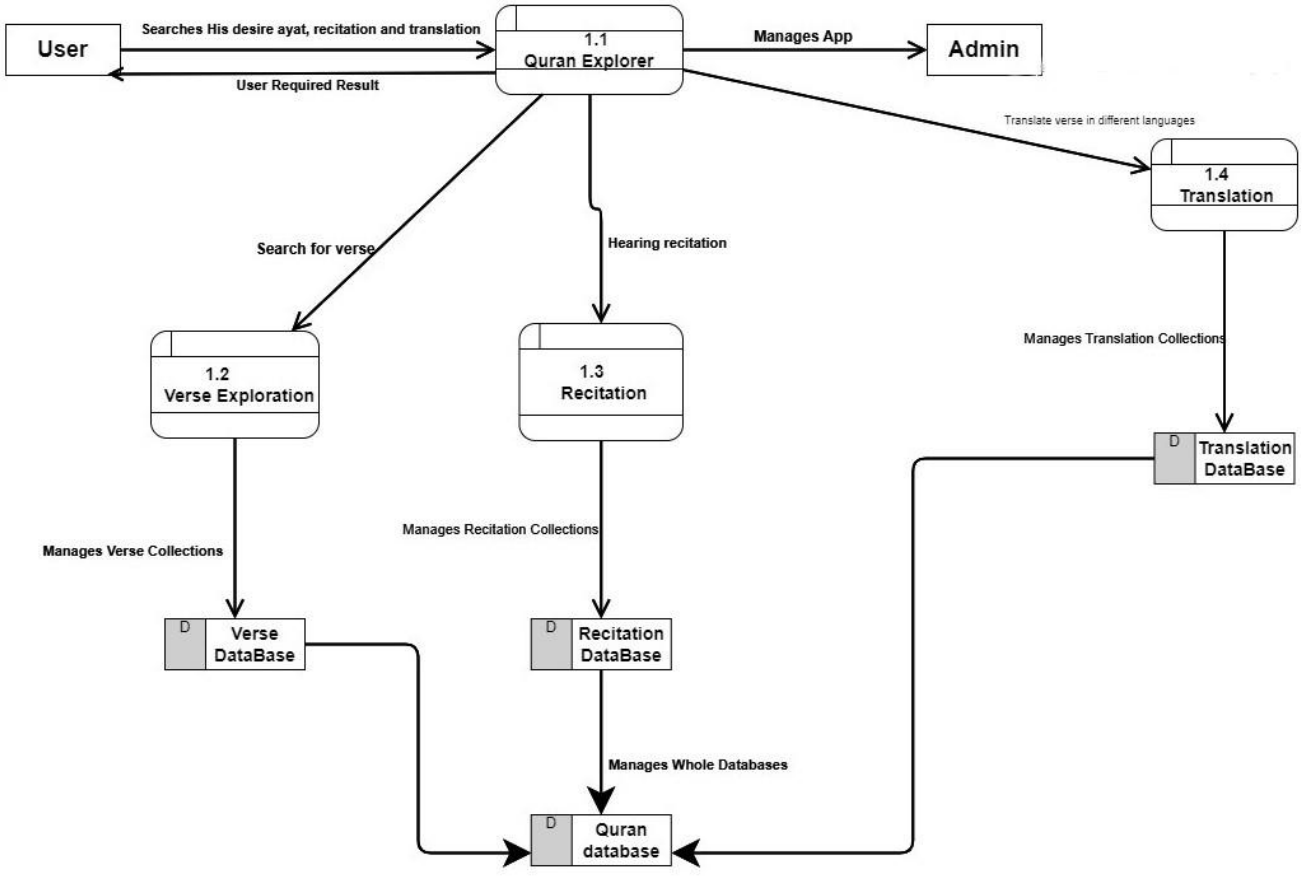
0 level DFD:

In the zero level data flow diagram, User (entity) request to the Quran Explorer (process) to search his desired ayat and then it delivered the required result to the user. Similarly, Admin (entity) manages and maintain the whole app. Quran database (warehouse) manages the database of the app.



1 level DFD:

In the one level data flow diagram, User (entity) request to the Quran Explorer (process) to search his desired verse and then it delivered the required result to the user. Similarly, Admin (entity) manages and maintain the whole app. But in this section, Quran Explorer is further divided into three processes i.e, Verse Exploration, Audio Recitation and Translation. All these processes have their own databases which manages them. At the end, these databases are maintained by the Quran database (warehouse)



Entity Relationship Model:

The ER model for the Quran Explorer app outlines the essential entities and their relationships. It includes the 'User' entity for user profiles, 'QuranicText' for Quranic content, 'Translation' for translations, 'Recitation' for audio recitations, and 'Bookmark' for user-generated bookmarks. Users can create bookmarks, access translations, and listen to recitations, while QuranicText connects to translations and recitations. This model forms the data structure that powers the app's features, facilitating Quranic exploration and study for users.

Entity & Attributes:

1. USER: (UserID , Username, Password, Email, First_Name, Last_Name, Profile_Picture)

2. QURANIC_TEXT: (TextID ,Surah, Ayat, Arabic_Text, Translation_Text, Tafsir_Text)

3. TRANSLATION: (TranslationID, Name, Language, Translator)

4. RECITATION: (RecitationID ,Name, Reciter, Audio URL)

5. BOOKMARK: (BookmarkID , Date)

Relationships:

1. USER – BOOKMARK (One-to-Many):

- Each user can have multiple bookmarks.
- Each bookmark is associated with one user.

2. USER - RECITATION (Many-to-Many):

- Each user can have multiple recitations they've listened to.
- Each recitation can be associated with multiple users who have listened to it.

3. USER - TRANSLATION (Many-to-Many):

- Each user can have multiple translations they've used.
- Each translation can be associated with multiple users who have used it.

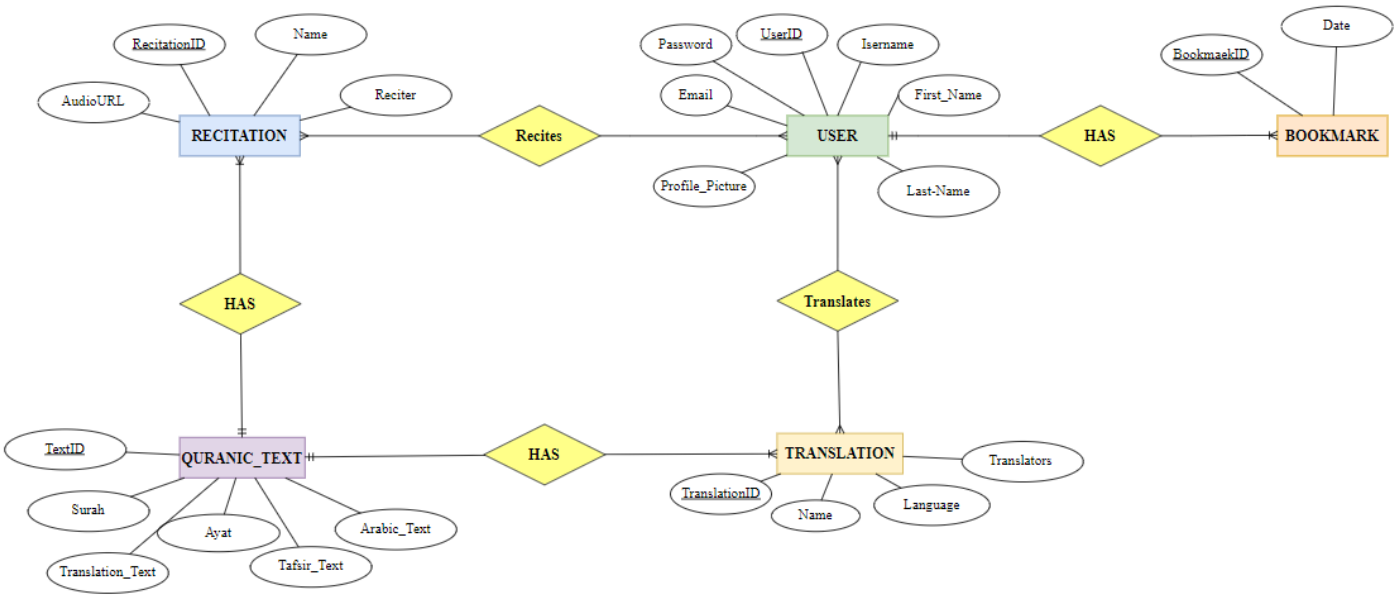
4. QURANIC_TEXT - TRANSLATION (One-to-Many):

- Each QuranicText can have multiple translations associated with it.
- Each translation is associated with one QuranicText.

5. QURANIC_TEXT - RECITATION (One-to-Many):

- Each QuranicText can have multiple recitations associated with it.
- Each recitation is associated with one QuranicText.

ER Diagram:



Use Case Model

A Use Case Model describes the proposed functionality of new system. A Use Case represents a discrete unit of interaction between a user (human or machine) and the system.

List of Actors:

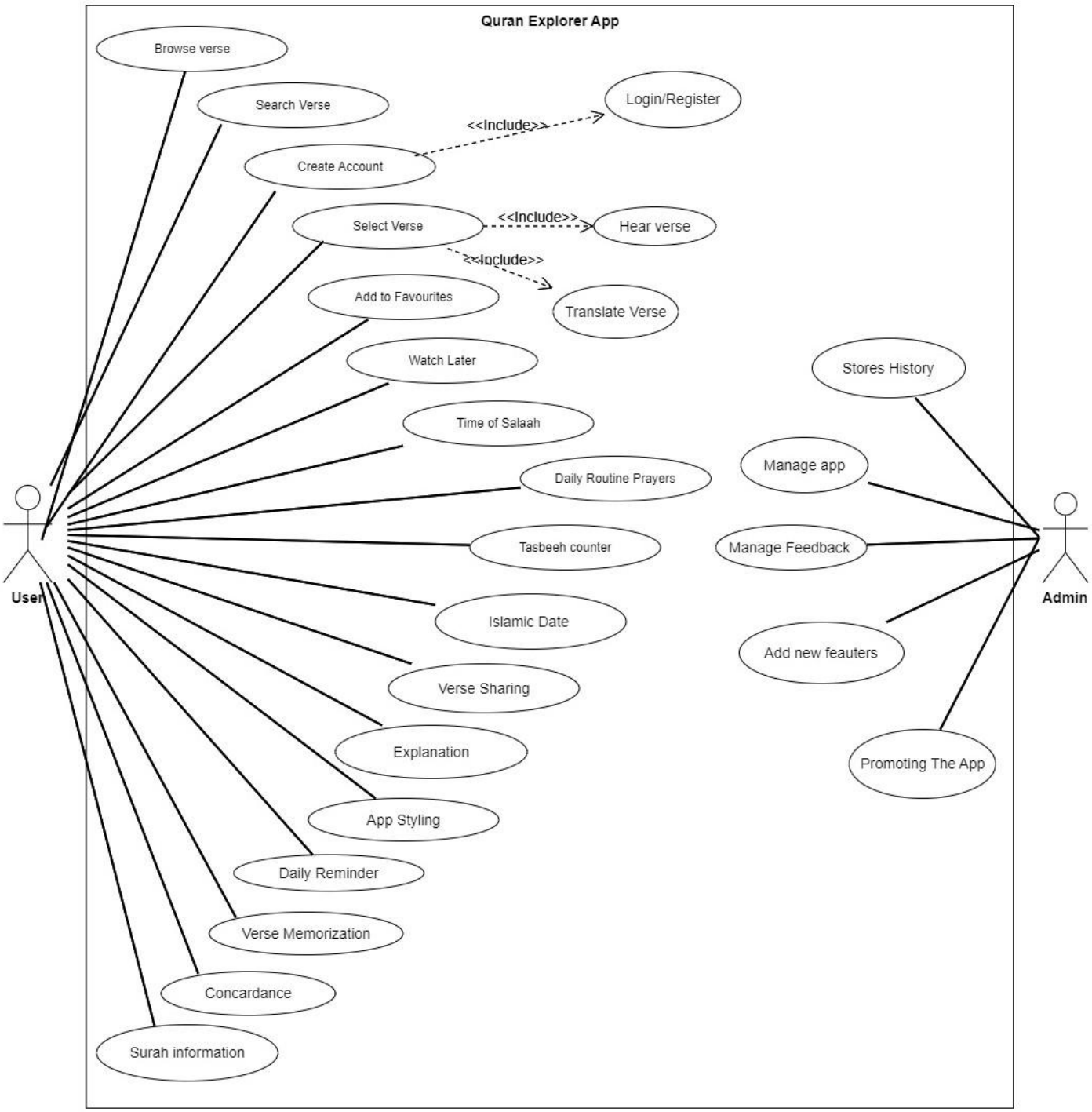
- **User:** this person recites, hears and translates the verse.
- **Administration:** this person manages the whole system.

List of Use Cases:

- **Browse Verse:** User browses the desired verse.
- **Search Verse:** User searches the desired verse..
- **Create Account:** User creates account.
- **Login/Register:** User logins or resisters himself.
- **Select Verse:** User selects verse.
- **Hear Verse:** User hears recitation of verse
- **Translate Verse:** User translates verse.
- **Add to Favourite:** User adds verses to favorites.
- **Add to watch later:** User also adds to watch later.
- **Time of Salaah:** User see time of salaah.
- **Daily Routine Prayers:** User recites daily routine prayers.
- **Tasbeeh Counter:** User uses tasbeeh counter.
- **Islamic Date:** User sees Islamic date, month and year.
- **Sharing Verse:** User shares verse on social media.
- **Explanation:** User finds explanations and interpretations of the Quranic verses.
- **App Styling:** User sets app font size, theme, reciter preference, and translation options.
- **Daily Reminder:** User uses option to set reminders for daily Quranic readings,
- **Verse Memorization:** User memorises Quranic verses, using memorization tracker, recitation repetition,etc.

- **Concordance:** User uses concordance feature that allows users to search for specific words or phrases across the entire Quran.
- **Surah Information:** User accesses detailed information about each surah, including its name, revelation order, number of verses.
- **Stores History:** Administration stores the search history of User.
- **Manage App:** Administration manages app database..
- **Manage Complain/Feedbacks:** Administration handles complains/feedbacks given by the User.
- **Add new Features :** Administration adds new features by collaborates with the development team.
- **Promoting the App :** The administrator promotes the app and increasing its user base.

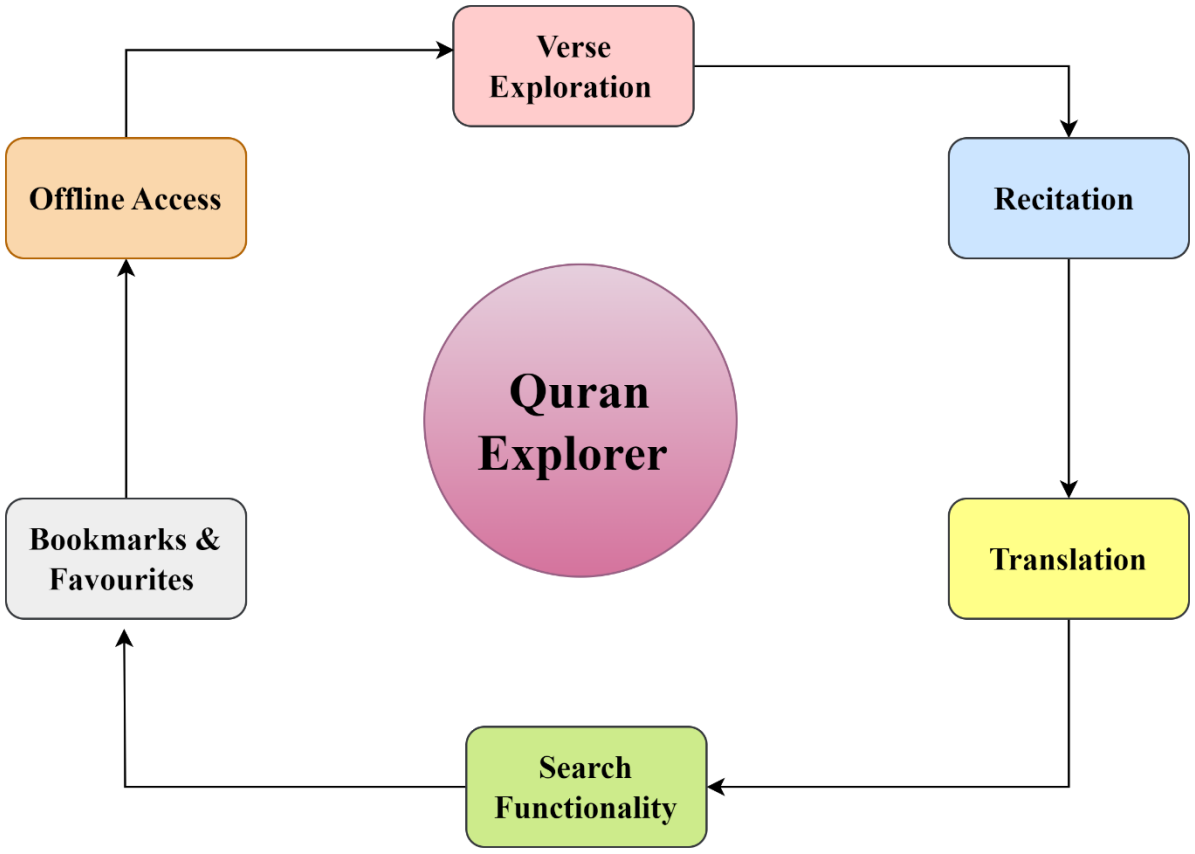
System Level Use Case Diagram:



Software Requirement Specifications:

Software Requirements Specification is a comprehensive and formal document that outlines the detailed requirements and specifications for a software project. The primary purpose of an SRS is to serve as a blueprint or a reference document for all persons involved in the development process, including software developers, designers, project managers, and clients.

SRS of Quran Explorer app is as follows:



A short description of entities involved in SRS of Quran explorer is as follows:

1) **Verse Exploration:**

With Quran Explorer, you can easily navigate through the chapters (Surahs) and verses (Ayahs) of the Quran. You can browse through the entire Quran or select specific chapters to study.

2) **Recitation:**

The app offers an extensive library of recitations by renowned reciters from around the world. You can listen to the beautiful and melodious recitations. The recitations are available in various recitation styles (Qira'at) .

3) **Translation:**

Quran Explorer provides translations of the Quran in multiple languages, allowing you to comprehend the meanings of the verses.

4) **Search Functionality:**

It implement a fast search feature that allows users to easily find specific verses, topics, or keywords within the Quran, making it convenient for them to access relevant content.

5) **Bookmarking and Favorites:**

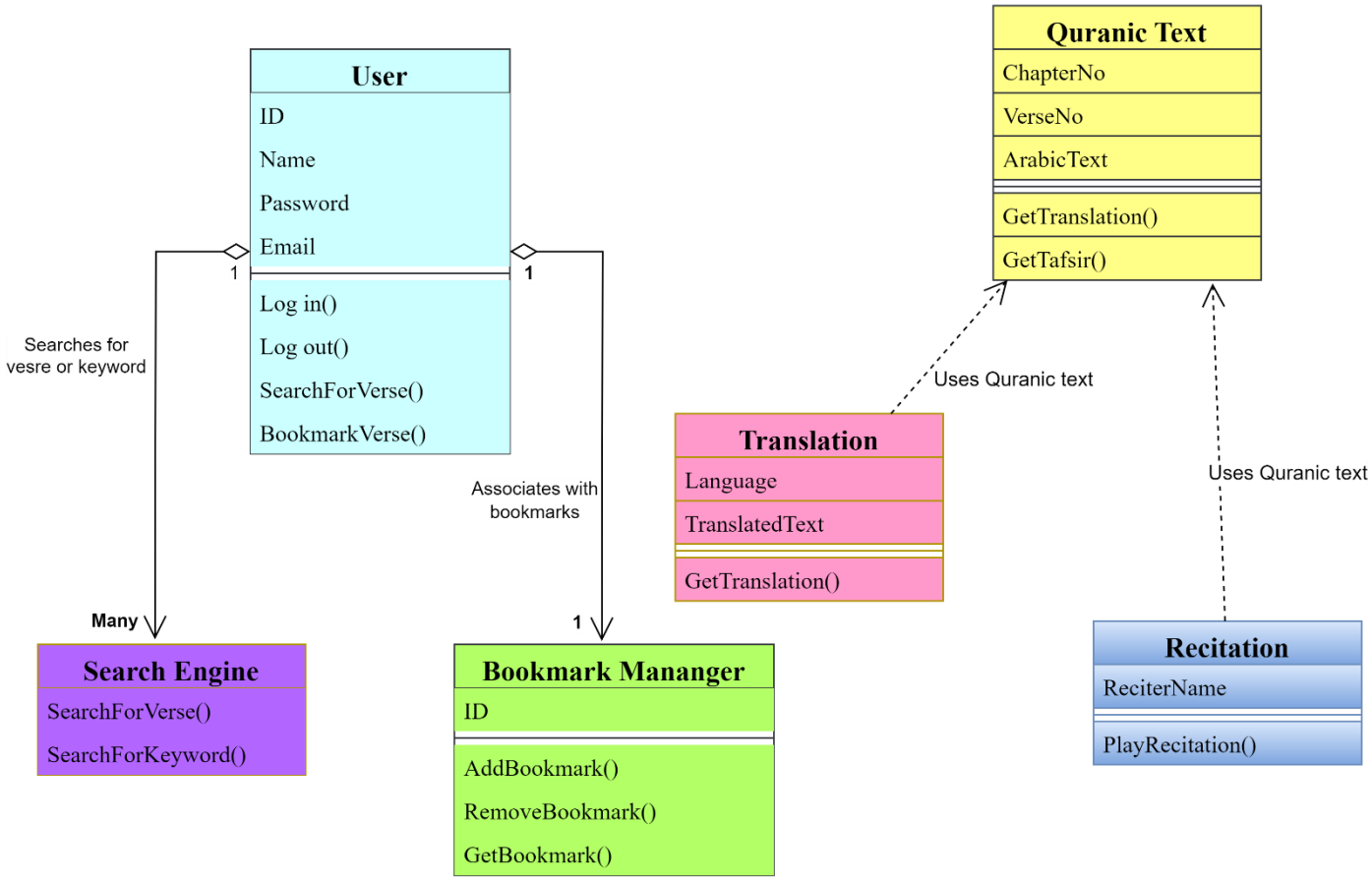
It enable users to bookmark their favorite verses or save them to a favorites list for quick and easy access during their study sessions.

6) **Offline Access:**

It allow users to access the Quranic text, translations, and previously accessed content even without an internet connection, ensuring continuous usability and access in areas with limited connectivity.

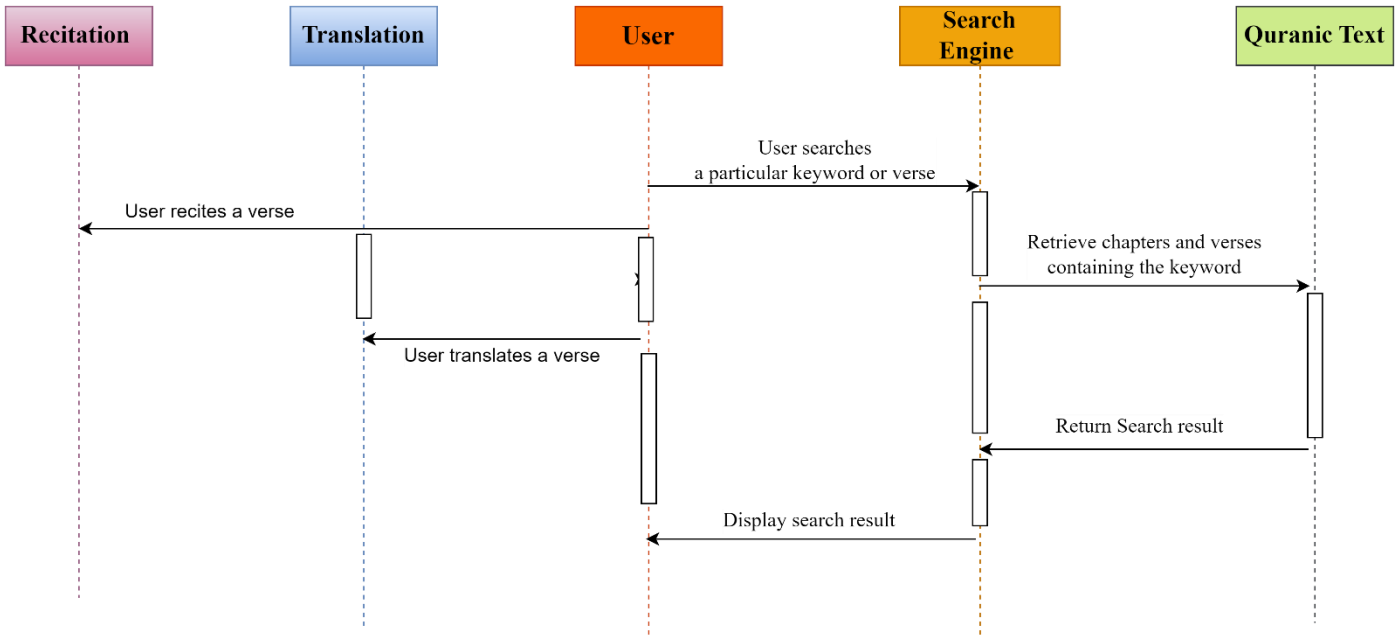
Class Diagram:

In the Quran Explorer app's class diagram, some main classes have been identified to represent the app's core functionalities. These classes include 'User' to manage user profiles and authentication, 'QuranicText' to handle the Quranic verses and related data, 'Translation' for managing translations of the Quran, 'Recitation' to handle audio recitations, 'SearchEngine' to facilitate efficient searches, and 'Bookmark' to manage user-generated bookmarks. These classes work in harmony to create a comprehensive Quranic exploration experience for users. Together, these classes form the backbone of the Quran Explorer app, delivering a rich and user-friendly Quranic exploration platform.



Sequence Diagram:

The sequence diagram for the Quran Explorer app provides a visual representation of how the app processes user requests, showing the interactions between classes and objects in real-time.In the sequence diagram for the Quran Explorer app, we outlined the dynamic interactions between different components during the execution of specific user actions. For instance, when a user initiates a search for a Quranic verse, the diagram illustrates the sequence of events, starting from the ‘User’ interface, which sends a search request to the ‘SearchEngine’ class. The ‘SearchEngine’ then interacts with the ’QuranicText’ class to retrieve the relevant data. Once the data is obtained, it's presented to the user through the user interface. Now it is user’s choice either this ersult is moved to ‘Translation’ class or ‘Recitattion’ class.



Work Breakdown Structure (WBS):

The Work Breakdown Structure (WBS) for the Quran Explorer app offers a hierarchical breakdown of the project's tasks and components. At the top level, it categorizes the project into phases, including Requirement Analysis, Design and Planning, Development, Testing, Deployment, Documentation, Quality Assurance, Risk Management, Project Management, and Documentation and Reporting. Each phase is further divided into specific tasks and sub-tasks. For example, the 'Development' phase encompasses 'Frontend Development,' 'Backend Development,' This structured WBS facilitates project management, resource allocation, and tracking by providing a clear outline of the project's scope and tasks, from high-level objectives down to granular activities.