

Project final report

SMART QURAN



CCSW 315 SOFTWARE PROCESS MODELS

Course instructor: Jamila Alamri

SECTION: SB7

PROJECT MEMBERS:

Raneem Alowide, 2110171 & Manar Mughrbal, 2111298

Dana Alasmari, 2116655 & Yara Bashmail, 2110215

Raghad Askool, 2110645

TABLE OF CONTENT:

- 1- Project initiation:
 - 1.1- Project description
 - 1.2- Problem definition
 - 1.3- Proposed solution
 - 1.4- Scope
 - 1.5- Users
 - 1.6- Methodology and tools
 - 1.7- Prioritized functional and non-functional requirements.
 - 1.6.1- Functional requirements
 - 1.6.2- non-Functional requirements
 - 1.8- Story backlog
 - 1.9- Prioritize stories and defined sprints.
 - 1.10- Use case and class diagrams.
- 2- Sprint 1:
 - 2.1- test cases
 - 2.1.1- USER LOGIN AND AUTHENTICATION
 - 2.1.2- Registration
 - 2.1.3- accessibility
 - 2.2- sprint backlog
 - 2.3- user story
 - 2.4- Trello
 - 2.5- Task allocation
 - 2.6- sprint meeting
 - 2.7- sequence diagram:
 - 2.7.1- Accessibility diagram.
 - 2.7.2- Login diagram.
 - 2.7.3- Registration diagram.
- 3- Sprint 2:
 - 2.1- test cases
 - 2.1.1- Teacher
 - 2.2- sprint backlog
 - 2.3- user story
 - 2.4- Trello
 - 2.5- Task allocation
 - 2.6- sprint meeting
 - 2.7- sequence diagram:
 - 2.7.1- Tutor Session Booking diagram.
- 4- Project Implementation Screenshots.

PROJECT INITIATION

PROJECT DESCRIPTION:

The aim of our project is to develop a software program that uses artificial intelligence techniques to facilitate learning the Holy Quran. The program will include many features such as:

- Translation and Tafsir: The program can provide translations and tafsir of Quran (ayah) in different languages.
- Progress Tracking: The program can track the user's progress and provide feedback and suggestions to improve their learning experience.
- Personalized Learning: The program can use machine learning algorithms to analyze the user's performance and adjust the learning content accordingly. This can help the user to learn at their own pace and improve their weak areas.
- Voice Recognition: The program uses advanced voice recognition technology to help users learn Quran pronunciation and improve their recitation skills. Users can interact with the software using their voice, and the program provides instant feedback and correction.
- bot settings: a bot feature can provide a more interactive experience for users, which can be helpful to users any time and support them based on their individual needs and learning style.
- Teacher: The program will have a feature where learners can have access to a live teacher who can provide Quran classes (Dhikr Circle) answer any questions they may have. This will provide users with the opportunity to receive personalized guidance and feedback on their progress, and to have a more interactive learning experience.

All these features will make the process of learning the Quran smoother and more effective. The program has been designed to meet the needs of different users, including children, beginners, and professionals.

PROBLEM DEFINITION:

Many people struggle to learn the Quran specially the people that don't speak Arabic due to the complexity of the Arabic language, the vast number of (surah) and (ayah). Additionally, finding the time and motivation to attend regular classes can be a challenge for individuals with busy schedules.

PROPOSED SOLUTION:

Developing a software program that uses artificial intelligence techniques to facilitate learning the Holy Quran. The program will include features such as translation and tafsir of Quran in different languages, progress tracking, personalized learning, and voice recognition.

These features are designed to make the learning process smoother and more effective, allowing users to learn at their own pace and improve their understanding and recitation skills. By providing a flexible and interactive learning environment, the program aims to address the challenges of learning the Quran, including the complexity of the Arabic language and the vast number of (surah) and (ayah).

SCOPE:

The scope of the project is to develop a software program that facilitates learning the Quran using artificial intelligence techniques. The program will include various features such as translation and tafsir of Quran in different languages, progress tracking, personalized learning, and voice recognition. The program will be designed to cater to different users, including children, beginners, and professionals, with the goal of improving their understanding and recitation skills of the Quran. The project's scope also includes testing and evaluating the software to ensure its functionality, usability, and effectiveness in addressing the challenges of Quran education. The project does not include the creation of new translations or tafsir, but it will incorporate existing ones.

USERS:

The users of the software program will be anyone interested in learning and understanding the Holy Quran. This includes children, beginners, and professionals from different backgrounds and cultures.

METHODOLOGY AND TOOLS:

- The methodology that will be used is (scrum)
- the project management tool is (Trello)
- the implementation language/tool is (flutter)
- the diagram tool is (lucid chart)

PRIORITIZED FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS:

FUNCTIONAL REQUIREMENTS:

| Requirement Definition | ID |
|------------------------------------------------------------------------------------------------|-------|
| Create an account | FR1 |
| The system shall enable the user and teacher to create an account. | FR1.1 |
| Login | FR2 |
| The system allows all the customers to enter the system with their supposed email and password | FR2.1 |
| View user profile | FR3 |
| The system shall view user profile for the user | FR3.1 |
| Read the Quran | FR4 |
| The system shall listen to the user's reading and corrects his mistakes | FR4.1 |
| Contact teachers | FR5 |
| The system shall allow communication with teachers | FR5.1 |
| Creation of Quranic courses | FR6 |
| The system shall allow create qur'anic courses between the users and teachers | FR6.1 |
| Listening to readers | FR7 |
| The system shall allow the user to listen to different readers | FR7.1 |
| Repeat listening and reading | FR8 |

| | |
|------------------------------------------------------------------------------------------------|--------|
| The system shall read the verses and then listen to the user's reading | FR8.1 |
| View translation | FR9 |
| The system shall show the translation of the verses to the user | FR9.1 |
| Ask bot | FR10 |
| The system shall allow the user to ask a question to the bot and it will automatically respond | FR10.1 |
| Tafsir | FR11 |
| The system shall interpret the verses | FR11.1 |

Interface Requirement:

- The user interface displays the services provided by the application, such as translation, tafsir, and others

Business Requirement:

- The application should be available to most people.

Regulatory/Compliance Requirement:

- The database of the users and their appointment information a functional audit trial.

Security Requirement:

- The program must provide integrity and saves the customers information.
- Program data must be protected from unauthorized.

NON-FUNCTIONAL REQUIREMENTS:

User Interface:

- The program must provide an easy and simple interface that is useable to use for all users.
- The program provides a GUI that suit for the target users of the application.
- The program must provide two interfaces one for the users and other for the teachers.

Hardware Interface:

- The hardware interface must be available on the entity's computers for the developer and on any users' devices.

Software Interface:

- The program must communicate with the database to extract needed information like the username and password and the other information of the users.
- The program provides fast navigation between user interfaces.

Security Requirements:

- The system asks for the password and the code sent to their phones after every 30 days.
- The system must allow only authorized users to improve the system.
- The system shall allow the customers to access only the services which they are authorize to parties.

STORY BACKLOG

Component Name: Registration

Story Name: Registration

Story Sequence No: 01

Story Short Description: Users can create an account to access the smart learning application.

Story Long Description: Users shall be able to create an account by providing their personal information, such as name, email, phone number, and address. The program shall validate the information and check for any duplicate accounts before creating a new user account. The program shall send a verification email to the user's email address to confirm the account creation.

Component Name: User Login and Authentication

Story Name: Login and Authentication

Story Sequence No: 02

Story Short Description: Registered users shall be able to log in to the system with their email and password.

Story Long Description: Users shall be able to access their accounts by entering their email and password. The system shall verify the user's credentials and allow them access to their account if they are valid. If the user enters invalid credentials, the system shall display an appropriate error message.

Component Name: User Progress Tracking

Story Name: Progress Tracking

Story Sequence No: 03

Story Short Description: Users can track their progress and set goals for Quran learning process.

Story Long Description: The program shall track the user's progress in various Quran learning activities, including vocabulary building, and more. The program shall display the user's progress in a dashboard that shows completion rates, scores, and other relevant metrics. Users shall be able to set goals and receive feedback and suggestions on how to improve their performance.

Component Name: Translation and Tafsir

Story Name: Translation and Tafsir

Story Sequence No: 04

Story Short Description: Users can view translations and tafsir of Quran in different languages.

Story Long Description: The program shall display translations and tafsir of Quran in different languages, such as English, Spanish, French, and Urdu. Users shall be able to switch between languages easily and view the translation and tafsir alongside the Arabic text. The program shall use reliable and accurate sources for translations and tafsir.

Component Name: Personalize Learning

Story Name: Personalized Learning

Story Sequence No: 05

Story Short Description: The learner shall be able to adjust the learning content according to their performance and needs.

Story Long Description: The program shall use machine learning algorithms to analyze the user's performance and adjust the learning content accordingly. For example, if the user is struggling with a particular (surah) or (ayah), the program may provide additional resources to help them improve. The program shall also adapt to the user's learning style, such as visual or hearing learners.

Component Name: Voice Recognition

Story Name: Voice Recognition

Story Sequence No: 06

Story Short Description: The program uses voice recognition technology to help users learn Quran pronunciation.

Story Long Description: The program shall use advanced voice recognition technology to recognize the user's voice and provide feedback on their recitation and pronunciation. The program shall analyze the user's pitch, rhythm, and intonation and provide suggestions for improvement. The program shall also allow users to interact with the software using their voice, such as asking questions about specific (surah) or (ayah).

Component Name: Accessibility

Story Name: Accessibility

Story Sequence No: 07

Story Short Description: The program is accessible to users with disabilities.

Story Long Description: The program shall be designed to be accessible to users with disabilities, such as users who are visually impaired or have hearing impairments. The program shall include features such as screen readers, captions, and sign language videos.

Component Name: Teacher

Story Name: Tutor Session Booking

Story Sequence No: 08

Story Short Description: Students can book tutoring sessions with available Teacher.

Story Long Description: Students shall be able to search for Teacher based on their availability. Once they find a Teacher, they can book a session with the Teacher. The system shall handle and store the session booking information in the database and notify the tutor and student about the session details.

Component Name: Contact Bot

Story Name: Ask bot

Story Sequence No: 09

Story Short Description: Users can contact the support team through the Contact Bot.

Story Long Description: Users shall be able to access the Contact Bot and enter their query or feedback. The Contact Bot shall use natural language processing to understand the user's input and provide relevant responses or escalate the query to a human support agent if needed.

PRIORITIZE STORIES AND DEFINE SPRINTS:

Sprint #1:

Story Name: Registration

Story Sequence No: 01

Story Name: Login and Authentication

Story Sequence No: 02

Story Name: Accessibility

Story Sequence No: 07

Sprint #2:

Story Name: Tutor Session Booking

Story Sequence No: 08

Story Name: Voice Recognition

Story Sequence No: 06

Sprint #3:

Story Name: Translation and Tafsir

Story Sequence No: 04

Story Name: Personalized Learning

Story Sequence No: 05

Sprint #4:

Story Name: Progress Tracking

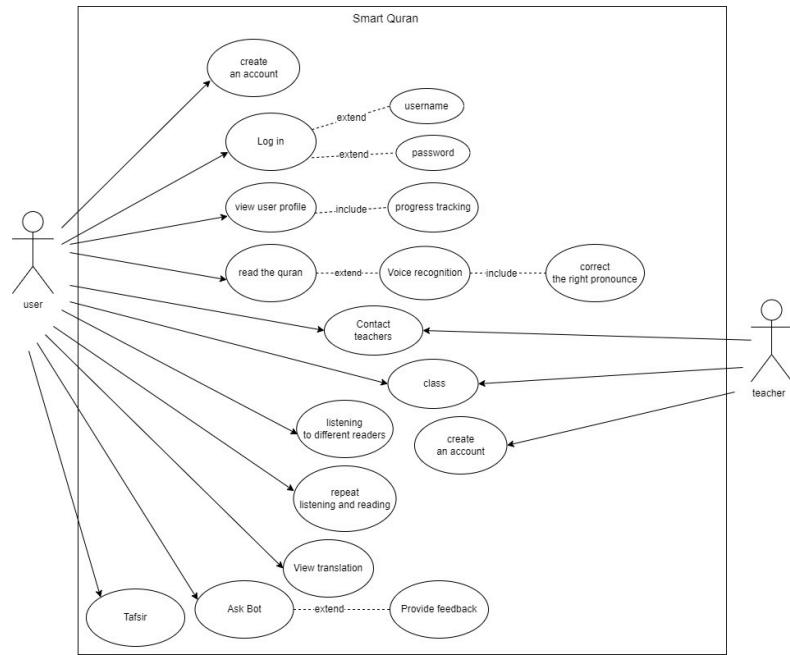
Story Sequence No: 03

Story Name: Ask bot.

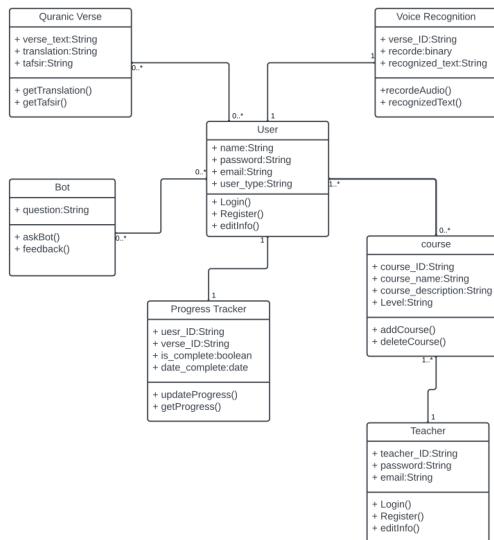
Story Sequence No: 09

USE CASE AND CLASS DIAGRAM:

USE CASE DIAGRAM:



CLASS DIAGRAM:



SPRINT 1

TEST CASE:

TEST CASE NAME: USER LOGIN AND AUTHENTICATION

| Sr. No. | Functional Test Cases | Expected results |
|---------|----------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| 1 | Test if the user enter invalid username/password, and verify that the user is not able to login | The user should not be able to login and should receive an error message indicating that the login information is incorrect |
| 2 | Test if the user login with valid information, and verify that the account is successfully created | The user should be able to successfully login and should be taken to the app's main screen |
| 3 | Test if the user login with empty fields, and verify that an error message is displayed | The user should not be able to login and should receive an error message indicating that the login fields are required |

TEST CASE NAME: REGISTRATION

| Sr. No. | Functional Test Cases | Expected results |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Test if the user register with a username or email that already exists in the apps, and verify that an error message is displayed | The user should not be able to register with a username or email that already exists in the app's and should receive an error message indicating that the username or email is already taken |
| 2 | Test password strength, and verify that the password strength requirements are enforced such as requiring a minimum length or including special characters | The password strength requirements should be enforced, and the user should receive an error message if the password does not meet the requirements |

| | | |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| 3 | Test the expected input fields that required for the registration process. These may include fields such as name, email, password, and date of birth | The user should be able to move on to the next step of the registration process if all required input are filled out correctly |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|

TEST CASE NAME: ACCESSIBILITY

| Sr. No. | Functional Test Cases | Expected results |
|---------|--------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| 1 | Test accessibility features in smart Quran app. These may include features such as screen readers, sign language videos | The accessibility features should be clearly identified and easily accessible to user |
| 2 | Test the accessibility to users with disabilities and gather feedback on the accessibility features | The app should be accessible to users with disabilities and should receive positive feedback |
| 3 | Test the captions feature by turning on captions for videos and verifying that the captions are accurate and synchronised with audio | The captions should accurately reflect the audio content and synchronised with video |

SPRINT BACKLOG:

1. The system shall enable the user and teacher to create an account.
2. The system allows all the customers to enter the system with their supposed.
3. email and password -The user shall be able to choose the accessibility they need.

USER STORY:

Component Name: Registration

Story Name: Registration

Story Sequence No: 01

Story Short Description: Users can create an account to access the smart learning application.

Story Long Description: Users shall be able to create an account by providing their personal information, such as name, email, phone number, and address. The program shall validate the information and check for any duplicate accounts before creating a new user account. The program shall send a verification email to the user's email address to confirm the account creation.

Component Name: User Login and Authentication

Story Name: Login and Authentication

Story Sequence No: 02

Story Short Description: Registered users shall be able to log in to the system with their email and password.

Story Long Description: Users shall be able to access their accounts by entering their email and password. The system shall verify the user's credentials and allow them access to their account if they are valid. If the user enters invalid credentials, the system shall display an appropriate error message.

Component Name: Accessibility

Story Name: Accessibility

Story Sequence No: 07

Story Short Description: The program is accessible to users with disabilities.

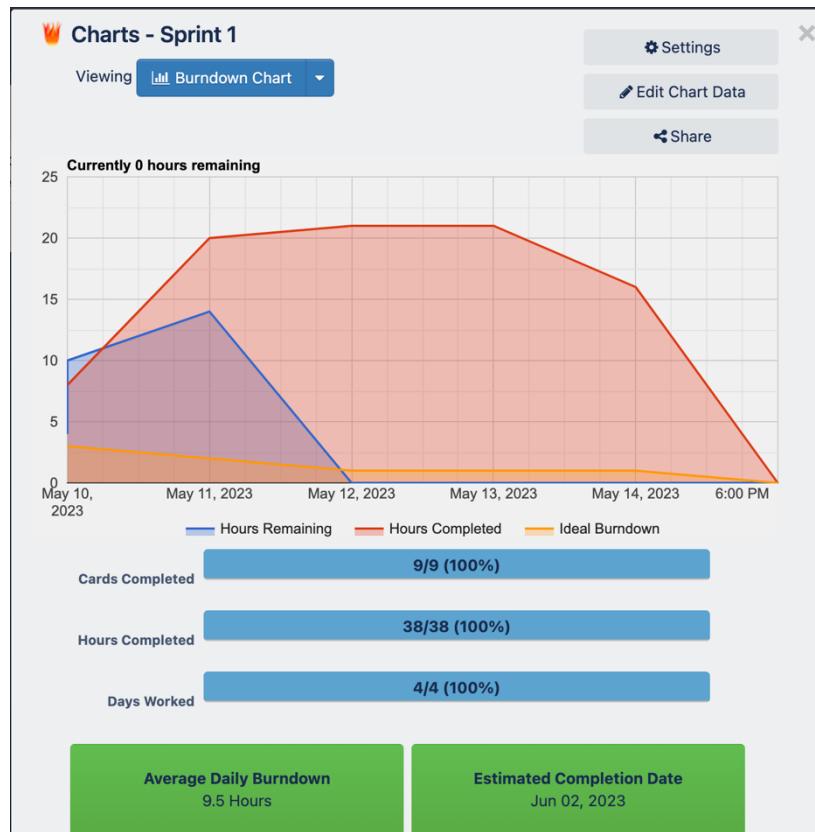
Story Long Description: The program shall be designed to be accessible to users with disabilities, such as users who are visually impaired or have hearing impairments. The program shall include features such as screen readers, captions, and sign language videos.

TRELLO:

The Trello board for Sprint 1 is organized into four main columns:

- Product backlog (Done)**: Contains cards like "as a user, i want to be able to be able to use accessibility when i need it".
- Sprint backlog (Done)**: Contains cards like "The system shall enable the user to create an account.", "The system allows all the users to enter the system with their supposed email and password", and "The user shall be able to choose the accessibility they need".
- Done**: Contains cards for "Design" (May 10 - May 13), "write user story" (May 10 - May 12), "write test cases" (May 10 - May 15), and "Implementation of code" (May 12 - May 14).
- to Do**: A placeholder column with a "+ Add a card" button.

The sidebar shows the workspace "SmartQuran Free" with "Boards", "Members", and "Workspace settings". The "Your boards" section shows "Sprint 1" as the active board. A pink banner at the bottom says "Try Premium free".



TASK ALLOCATION:

- **Raneem Alowide:** Code implementation , user story
- **Manar Mughrbal:** Code implementation, Test cases
- **Dana Alasmari:** Test cases, user story
- **Yara Bashmail:** Sequence diagram, sprint backlog
- **Raghad Askool:** Sequence diagram, product backlog

SPRINT MEETING(S)

Project Name: [Smart Quran]

Project Members: [Raneem Alowide, Manar Mughrbal, Dana Alasmari, Yara Bashmail, Raghad Askool]

Sprint 1 Stand up Meeting

Sprint Duration: one week

Scrum Master: Raneem Alowide

Client: users of the application

Pair Programmers: Raneem Alowide, Manar Mughrbal, Dana Alasmari, Yara Bashmail, Raghad Askool

Stories: [List all stories for that sprint below]

| Component Name | Story Sequence Number | Use Cases (e.g., functionalities) |
|-------------------------------|-----------------------|--------------------------------------------------------------------------------------------|
| Registration | 1 | The system shall enable the user to create an account. |
| User Login and Authentication | 2 | The system allows all the users to enter the system with their supposed email and password |

| | | |
|---------------|----------|--------------------------------------------------------------|
| Accessibility | 7 | The user shall be able to choose the accessibility they need |
|---------------|----------|--------------------------------------------------------------|

Follow-up meeting questions: [SCRUM MASTER NAME or CLIENT NAME should ask the following questions to the pair programmers]:

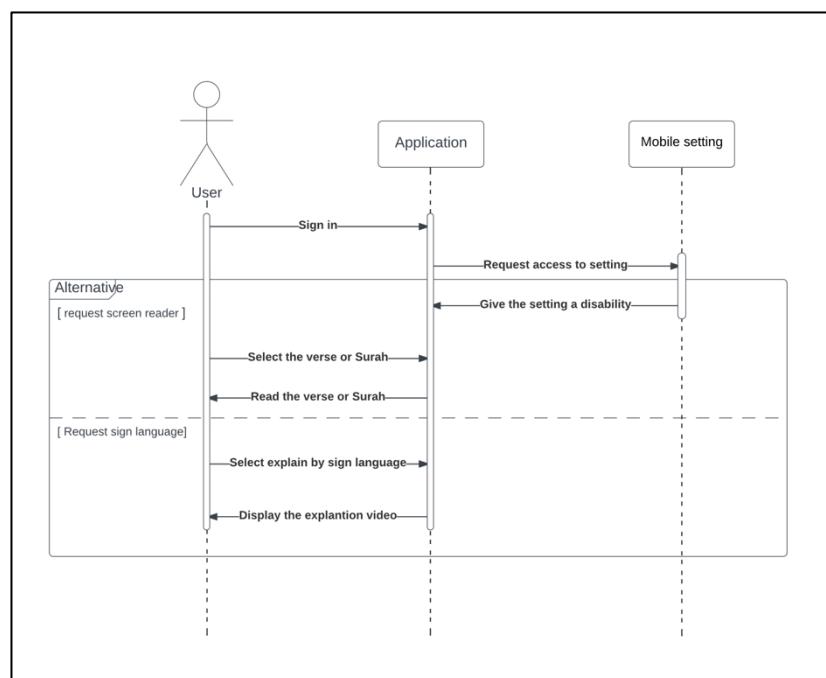
1. What has been completed since the last meeting? New functions have been implemented and completed.
2. What are you going to be working on next? work on developing a new function.
3. Do you have any issues/impediments? Some back-end difficulties, but we were able to resolve them.
4. **Scrum's Master comments based on the above questions:**

Although the teamwork was not so smooth, but we were able to managed it,

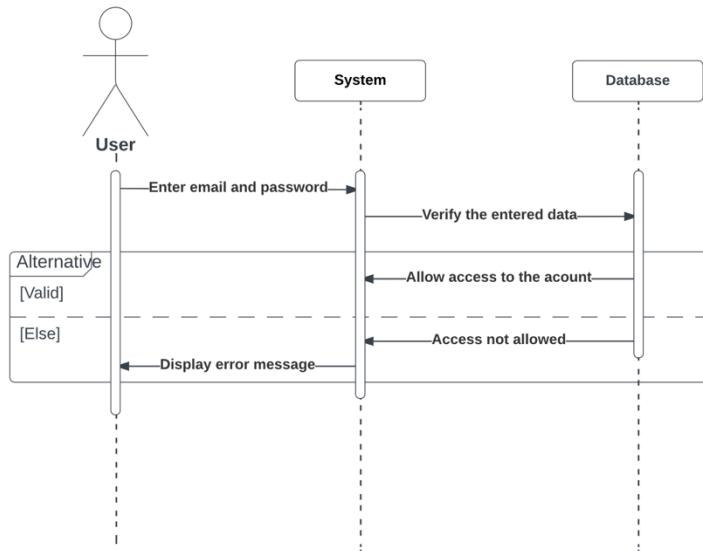
We might concentrate on fewer functions during a sprint.

SEQUENCE DIAGRAM:

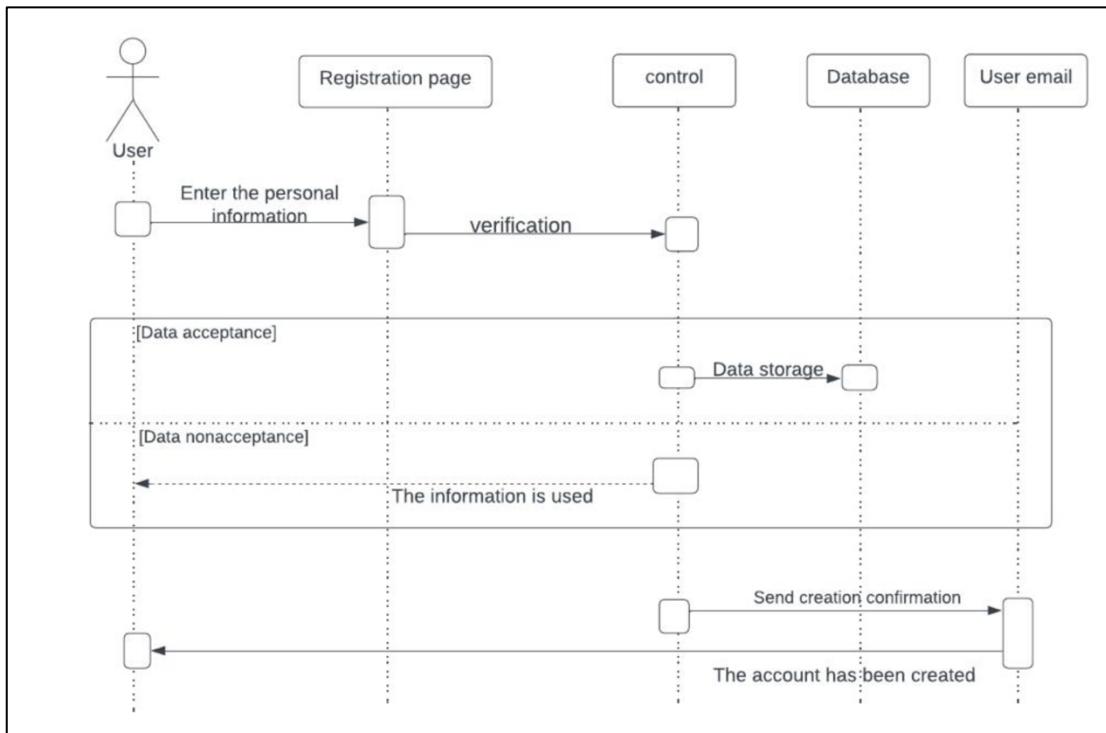
ACCESSIBILITY DIAGRAM:



LOGIN DIAGRAM:



REGISTRATION DIAGRAM:



SPRINT 2

TEST CASE:

TEST CASE NAME: TUTOR SESSION BOOKING

| Sr. No. | Functional Test Cases | Expected results |
|---------|------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| 1 | The if user can view available tutors and their schedules | The page displays a list of available tutors and their schedules |
| 2 | Test if the user can book a session with a tutor | The booking confirmation message is displayed, and the session is added to the users booked session page |
| 3 | Test the user cannot book multiple sessions at the same time with same tutor | The button cannot be clicked more than one time. |
| 4 | Test if the users can see all their booked sessions | The page of the booked sessions is displayed with all booked sessions |

SPRINT BACKLOG:

1. The user should be able to book a session.
2. Show the available sessions with the specified time for the user.
3. Show booked sessions.
4. The user can choose the teacher.

USER STORY:

Component Name: Voice Recognition

Story Name: Voice Recognition

Story Sequence No: 06

Story Short Description: The program uses voice recognition technology to help users learn Quran pronunciation.

Story Long Description: The program shall use advanced voice recognition technology to recognize the user's voice and provide feedback on their recitation and pronunciation. The program shall analyze the user's pitch, rhythm, and intonation and provide suggestions for improvement. The program shall also allow users to interact with the software using their voice, such as asking questions about specific (surah) or (ayah).

Component Name: Teacher

Story Name: Teacher Session Booking

Story Sequence No: 08

Story Short Description: Students can book tutoring sessions with available Teacher.

Story Long Description: Students shall be able to search for Teacher based on their availability. Once they find a teacher, they can book a session with the Teacher. The system shall handle and store the session booking information in the database and notify the tutor and student about the session details.

TRELLO:

Sprint backlog (done)

- the user should be able to book a session (1)
- show the available sessions with the specified time for the user (2)
- show booked sessions (3)
- the user can to choose the teacher (4)

to do

- Design
- write user story
- write test cases
- implementation of code

Done

- write user story (1) May 24 - May 26 (MM)
- write test cases(2) May 24 - May 26 (DA)
- Design May 24 - May 26 (RA YB)
- implementation of code (7) May 24 - May 27 (RA)

You requested to join this Workspace. If a Workspace admin approves your request, other boards in this Workspace will show up here.
<https://trello.com/c/Cd1shgix/1-design>



TASK ALLOCATION:

- **Raneem Alowide:** Code implementation
- **Manar Mughrbal:** user story
- **Dana Alasmari:** Test cases
- **Yara Bashmail:** Sequence diagram, Trello
- **Raghad Askool:** Sequence diagram

SPRINT MEETING(S)

Project Name: [Smart Quran]

Project Members: [Raneem Alowide, Manar Mughrbal, Dana Alasmari, Yara Bashmail, Raghad Askool]

Sprint 2 Stand up Meeting

Sprint Duration: one week

Scrum Master: Raneem Alowide

Client: users of the application

Pair Programmers: Raneem Alowide, Manar Mughrbal, Dana Alasmari, Yara Bashmail, Raghad Askool

Stories: [List all stories for that sprint below]

| Component Name | Story Sequence Number | Use Cases (e.g., functionalities) |
|-----------------------|-----------------------|-------------------------------------------------------------------------------|
| Voice Recognition | 6 | The system shall listen to the user's reading and corrects his mistakes |
| Tutor Session Booking | 8 | The system shall allow create qur'anic courses between the users and teachers |

Follow-up meeting questions: [SCRUM MASTER NAME or CLIENT NAME should ask the following questions to the pair programmers]:

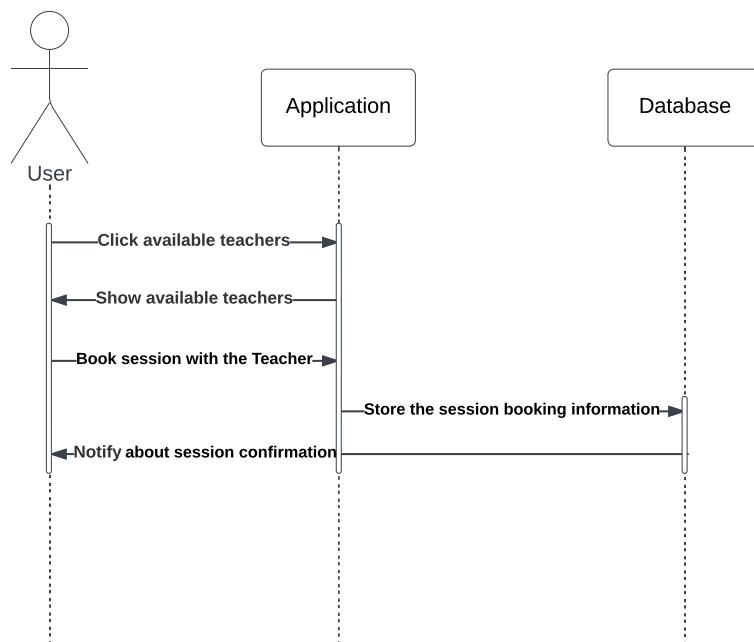
5. What has been completed since the last meeting? New functions have been implemented and completed.
6. What are you going to be working on next? work on developing a new function.
7. Do you have any issues/impediments? Some front-end difficulties, but we were able to resolve them.
8. **Scrum's Master comments based on the above questions:**

Although the teamwork was smooth, the time constraints were quite acute.

We might concentrate on fewer functions during a sprint.

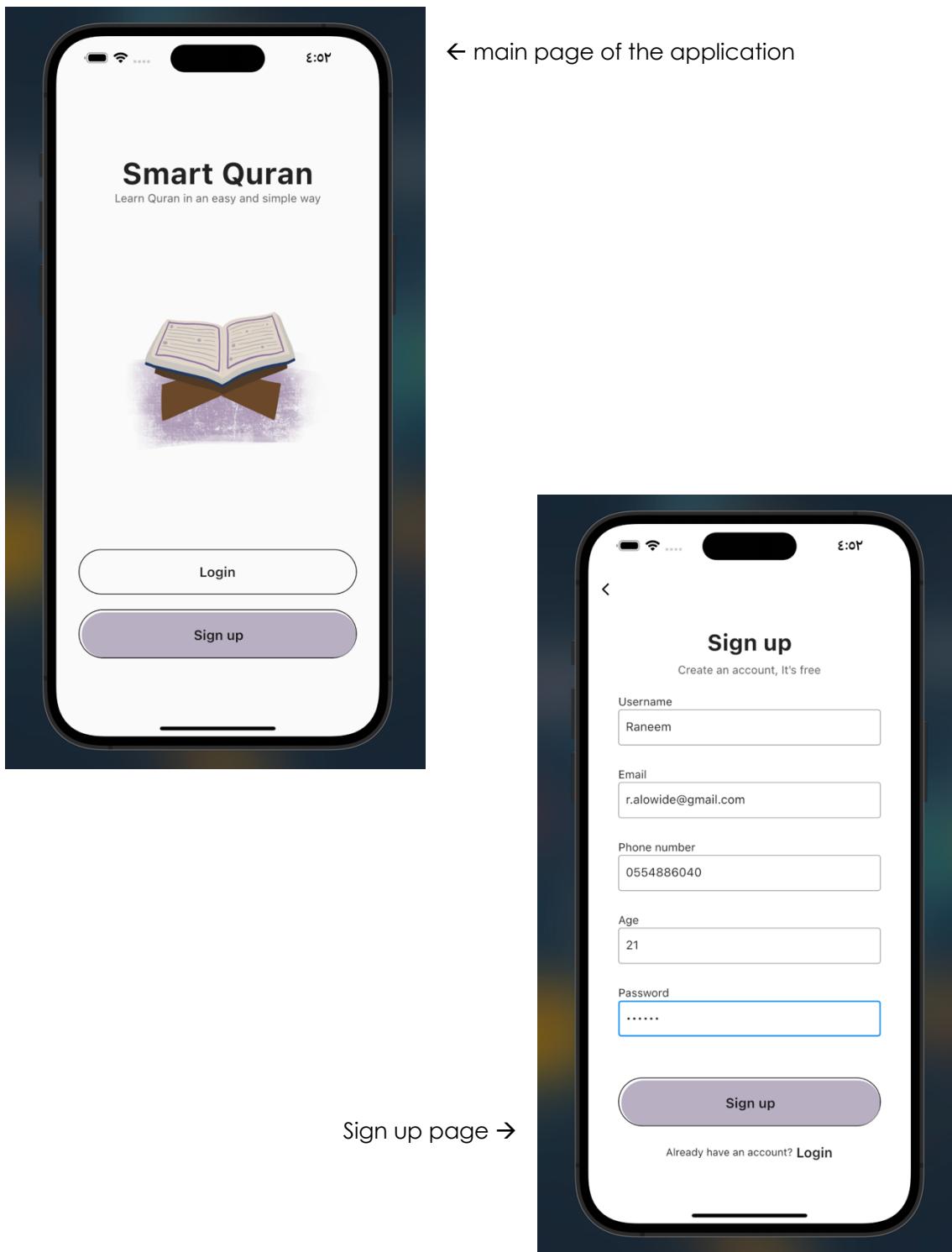
SEQUENCE DIAGRAM:

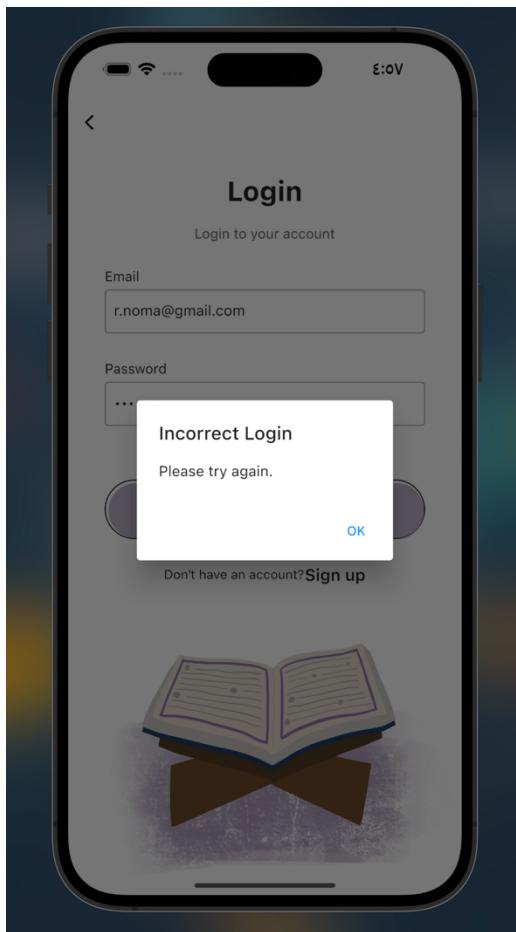
TUTOR SESSION BOOKING DIAGRAM:



IMPLEMENTATION:

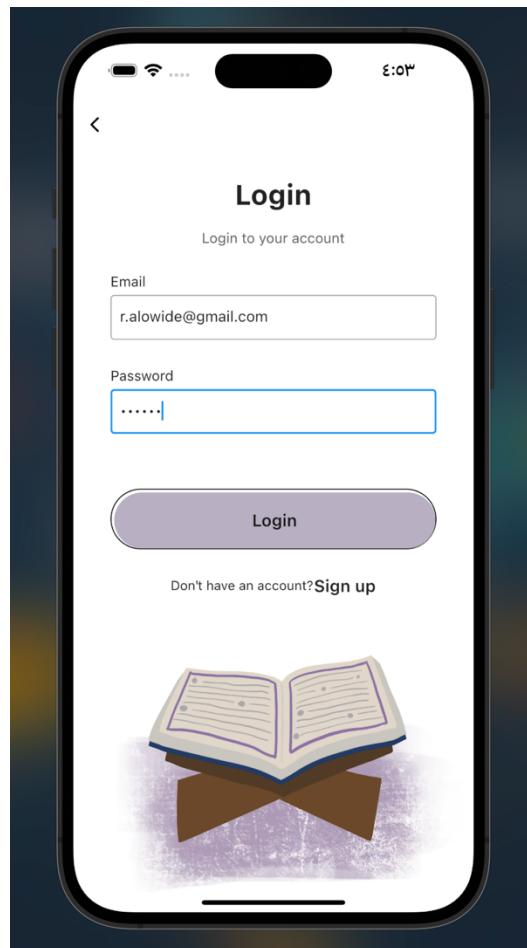
LINK TO THE SOURCE CODE ON GITHUB:
[HTTPS://GITHUB.COM/RNOMA18/MYSMARTQURAN.GIT](https://github.com/rnoma18/MySmartQuran.git)

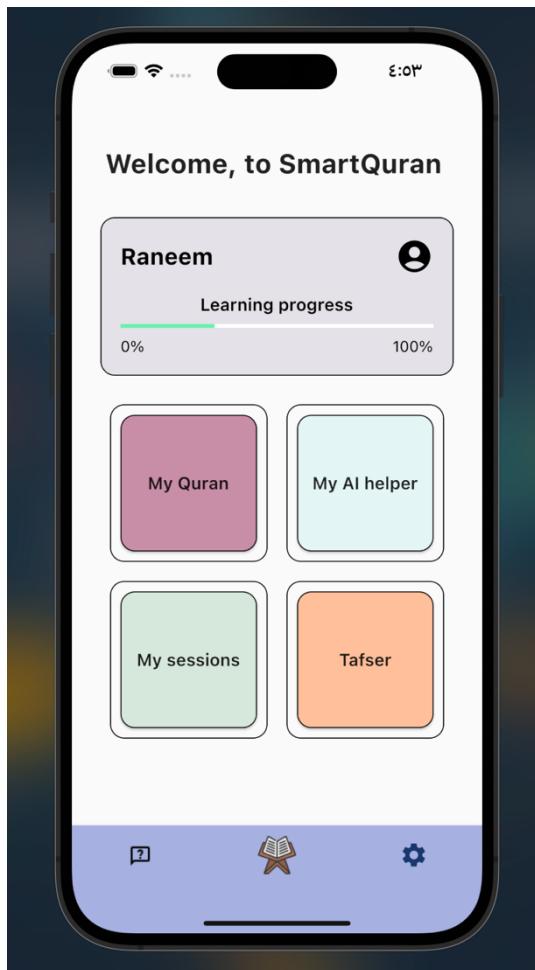




← if user try's to login and doesn't have an account or enters wrong email/password

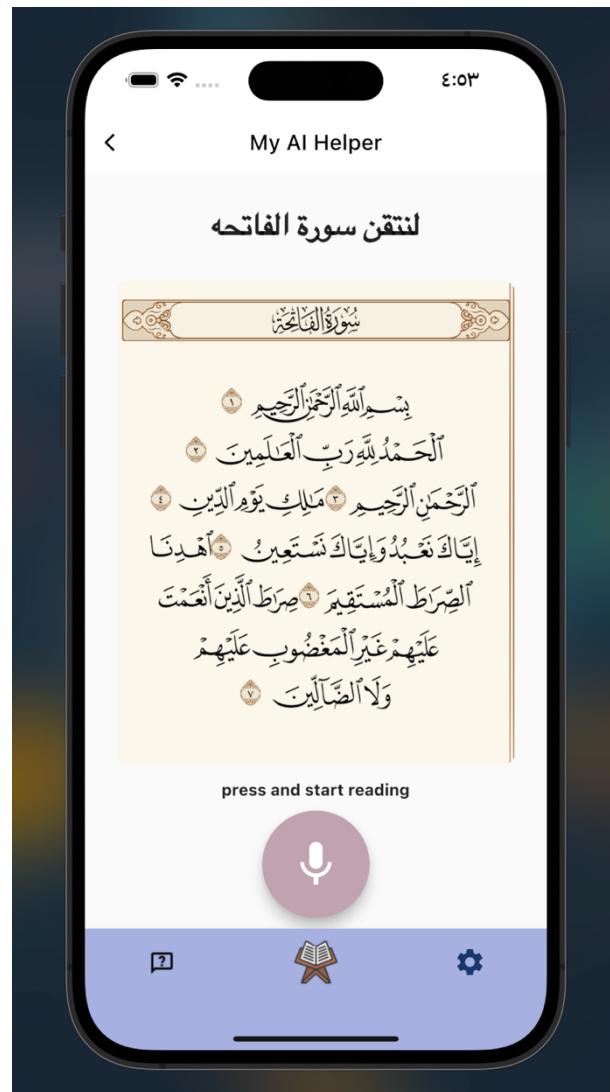
Login page→

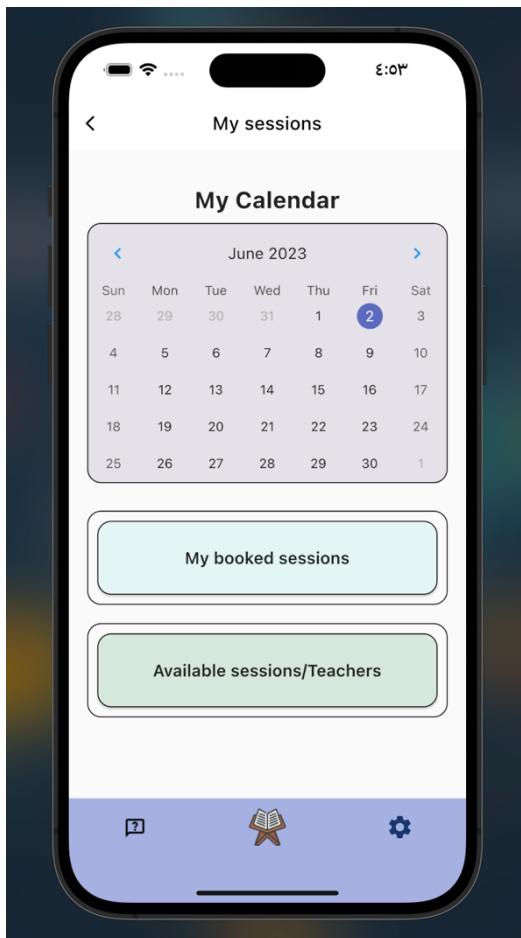




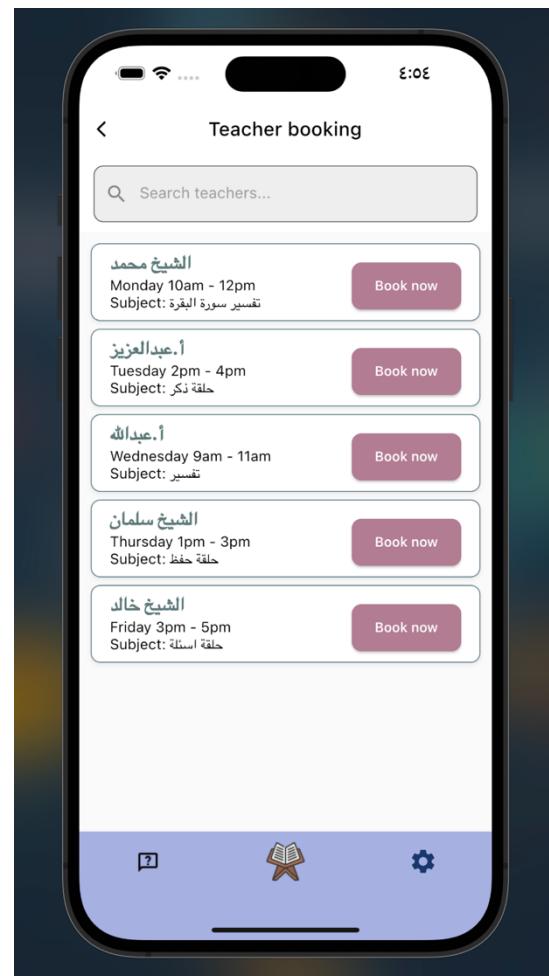
← Home page of application

My AI helper page →

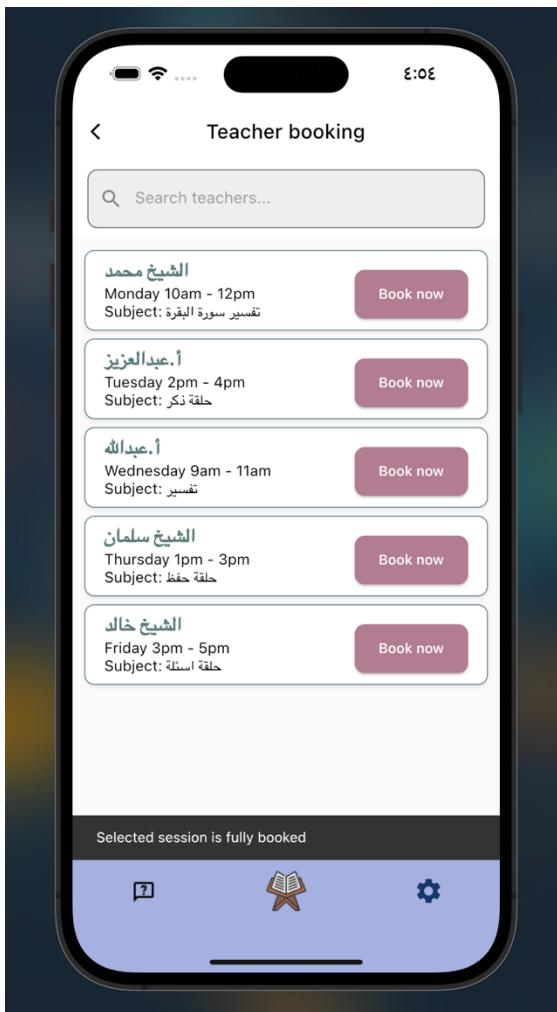




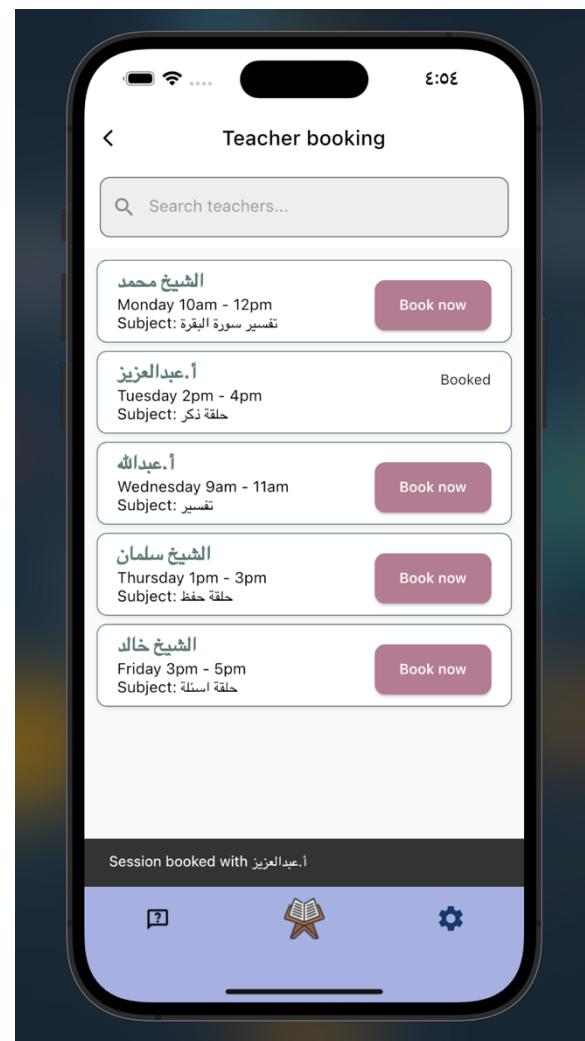
← My sessions main page



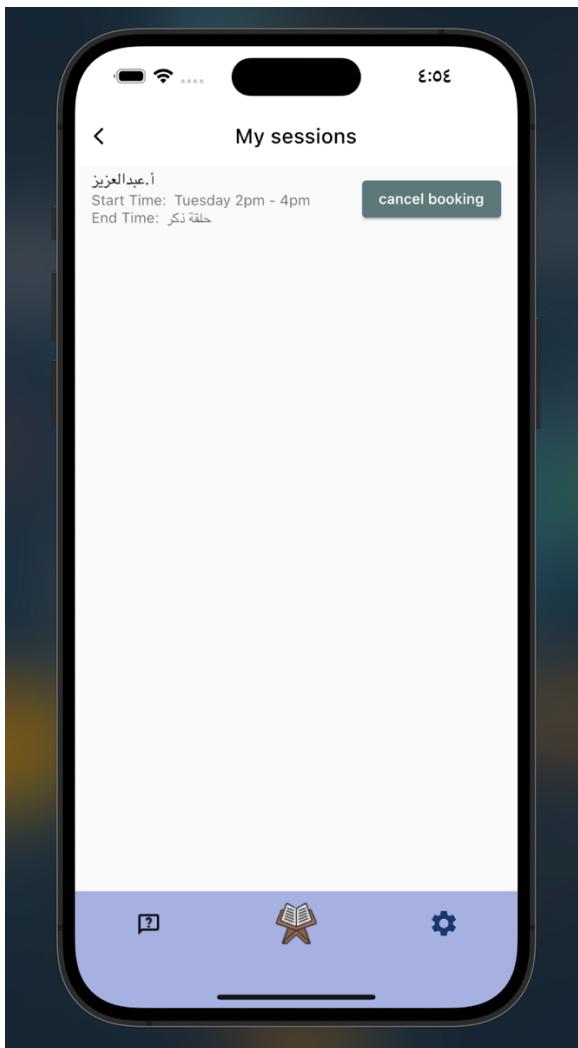
Teacher/Session booking page →



← if user try's to book a fully booked session

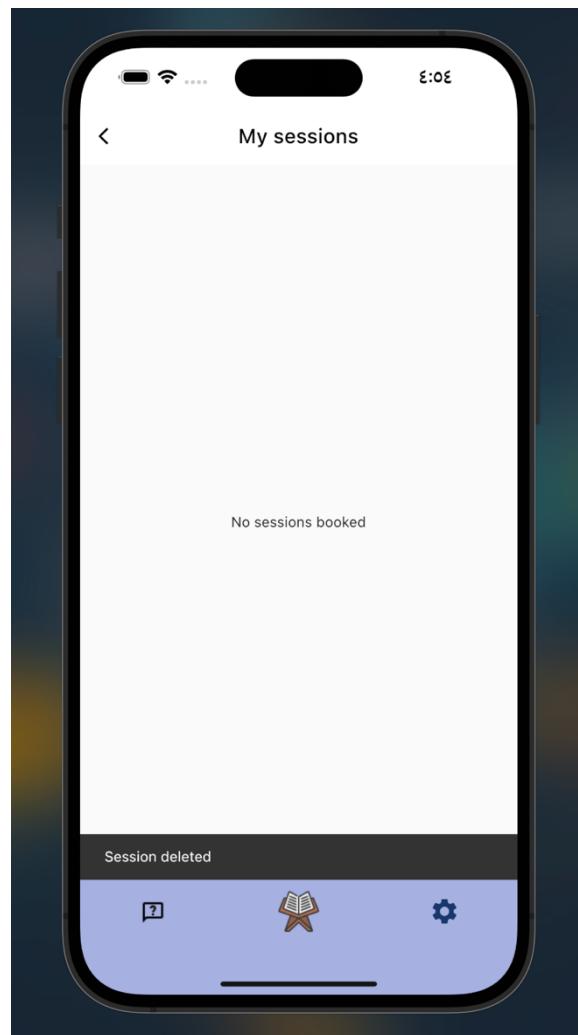


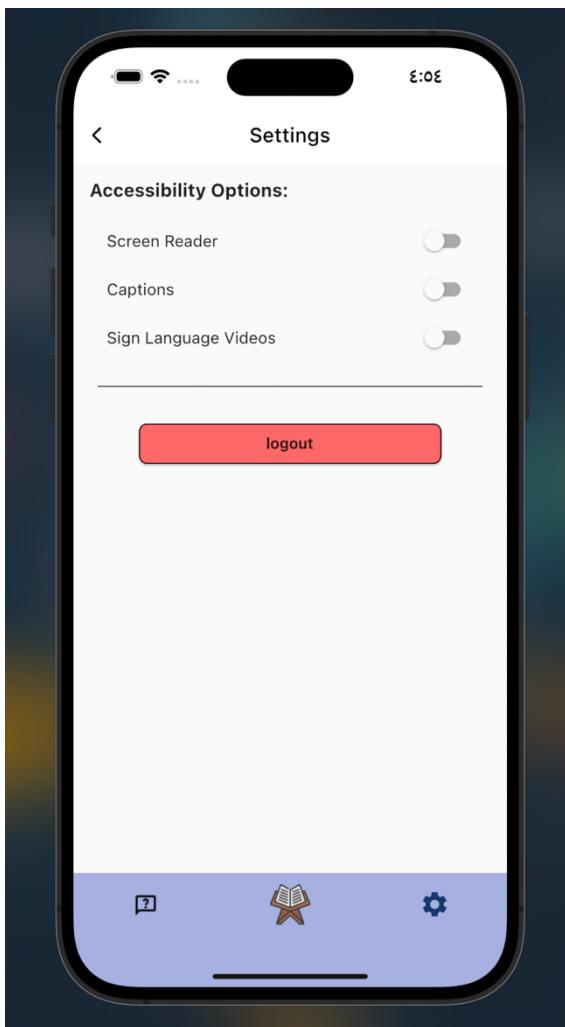
If user books a non-fully booked session →



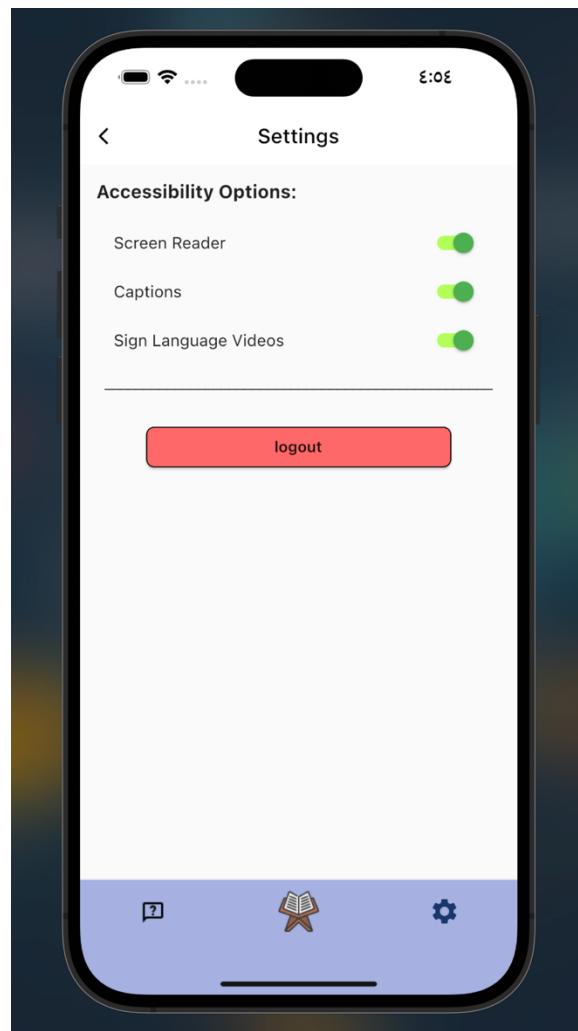
← My booked Sessions page (user he can see all their booked sessions and can cancel if they want)

If user cancels a booking and doesn't have any other bookings →





← settings page (here the user can turn on and off all the accessibilities they need) , user can also logout



If accessibility is turned on →