|  |
| --- |
|  |

|  |
| --- |
| MERN Based Website |
| Noor Al Eman |

**Submitted by**

* Zainab Eman 22F-3738
* Imama Kainat 22F-3661
* Noor Fatima 22F-3634

**Submitted to:**

* Dr. Qamar uz Zaman
* Mr. Awais Azam

**BSSE-6A**

**FAST NUCES CFD**

**Table of Contents**

[**Introduction** 1](#_Toc197894810)

[**Scope of the Project** 1](#_Toc197894811)

[**Target Users / Audience** 2](#_Toc197894812)

[Folder Structure 2](#_Toc197894813)

[Project Guide 3](#_Toc197894814)

[Installation Guide 4](#_Toc197894815)

[User Guide 4](#_Toc197894816)

[Features 7](#_Toc197894817)

[**Prayer Time** 7](#_Toc197894818)

[**Quran** 10](#_Toc197894819)

[**Azkar** 14](#_Toc197894820)

[**Tasbih** 17](#_Toc197894821)

[**Community** 20](#_Toc197894822)

[**Calendar** 21](#_Toc197894823)

[**Ramadan** 23](#_Toc197894824)

[Database Schema 25](#_Toc197894825)

[Architecture Diagram 26](#_Toc197894826)

[Advantages 26](#_Toc197894827)

[Testing and Validation 27](#_Toc197894828)

[Deployment (Planned) 27](#_Toc197894829)

**Introduction**

*Noor Ul Eman* is a full-stack Islamic web application developed for the Web Engineering course using the MERN stack (MongoDB, ExpressJS, ReactJS, NodeJS). It offers users a structured platform to access, recite, and reflect on Islamic content through modern web technologies.

**Scope of the Project**

The application integrates user authentication, secure data handling, and dynamic content modules including Qur’an recitation, Azkar, Namaz tracking, Ramazan goals, reflections, and a community forum. Additional features include activity logging, reCAPTCHA protection, and UI testing. It is designed for future extensibility.

**Target Users / Audience**

The platform is intended for Muslims of all age groups seeking a digital companion for daily worship, learning, and spiritual reflection. It is suited for users familiar with web applications who value structure and community interaction in their religious practice.

## Folder Structure

/

├── frontend/ # React Frontend Application

│ ├── public/ # Static public assets (index.html, favicon)

│ ├── src/ # Main source code

│ │ ├── assets/ # Images, icons, and static files

│ │ ├── components/ # Reusable UI components (e.g., Navbar, Footer)

│ │ ├── pages/ # Individual pages (Dashboard, Quran, Azkar)

│ │ ├── services/ # API request handlers (Axios or Fetch wrappers)

│ │ ├── hooks/ # Custom React hooks

│ │ ├── context/ # React Context API (user auth state, etc.)

│ │ ├── utils/ # Utility functions

│ │ ├── App.jsx # Root application component

│ │ ├── main.jsx # Entry point for Vite

│ ├── .env # Frontend environment variables

│ ├── vite.config.js # Vite configuration

│ ├── tailwind.config.js # Tailwind CSS configuration

│ ├── postcss.config.js # PostCSS configuration

│ └── package.json # Project metadata and dependencies

│

├── backend/ # Node.js / Express Backend

│ ├── config/ # Database and environment configuration

│ │ └── db.js # MongoDB connection setup

│ ├── controllers/ # Business logic for each route

│ │ ├── authController.js # Login, signup, logout, reCAPTCHA

│ │ ├── quranController.js # Fetching Qur'an surahs and audio

│ │ ├── azkarController.js # Morning, evening, post-prayer azkar

│ │ ├── prayerController.js # Namaz tracker logic

│ │ ├── goalsController.js # Ramazan goals

│ │ ├── reflectionController.js# User reflections

│ │ └── communityController.js # Forum interaction

│ ├── middleware/ # Authentication and validation middleware

│ │ ├── auth.js # JWT verification

│ │ └── recaptcha.js # Google reCAPTCHA middleware

│ ├── models/ # Mongoose schemas

│ │ ├── User.js # User model

│ │ ├── Azkar.js # Azkar schema

│ │ ├── Reflection.js # Reflections schema

│ │ ├── Prayer.js # Namaz tracker schema

│ │ ├── Goal.js # Ramazan goal schema

│ │ └── Community.js # Posts and replies

│ ├── routes/ # API route definitions

│ │ ├── authRoutes.js

│ │ ├── quranRoutes.js

│ │ ├── azkarRoutes.js

│ │ ├── prayerRoutes.js

│ │ ├── goalsRoutes.js

│ │ ├── reflectionRoutes.js

│ │ └── communityRoutes.js

│ ├── .env # Backend environment variables

│ ├── server.js # Main entry point for Express

│ └── package.json # Project metadata and dependencies

│

├── README.md # Project overview and instructions

└── .gitignore # Files and folders to ignore in version control

## Project Guide

To run the Noor Ul Eman application locally in a development environment, the following system dependencies must be installed and configured:

##### **Hardware Requirements**

* Minimum RAM: 4 GB (8 GB recommended)
* Processor: Dual-core or higher
* Storage: At least 500 MB of free space

##### **Software Requirements**

|  |  |  |
| --- | --- | --- |
| **Component** | **Minimum Version** | **Purpose** |
| Node.js | 18.x | Backend runtime and dependency manager |
| npm | 9.x | Node package manager for dependencies |
| MongoDB | 6.x (or Atlas) | Database server for storing application data |
| Git | Latest | Version control and repository cloning (optional) |
| Modern Web Browser | Latest | To access and test the frontend UI |
| Vite | Integrated | Development server and bundler for React |
| Operating System | Windows 10+, macOS, or any modern Linux distribution |  |

**Note:** Internet access is required to install packages, access MongoDB Atlas (if used), and stream Qur’an audio.

## ****Installation Guide****

**1. Clone the Repository**

Open your terminal and run:

git clone https://github.com/ZainabEman/Noor-Al-Eman.git

cd Noor-Al-Eman

**2. Install Dependencies**

**Backend**

cd backend

npm install

**Frontend**

cd frontend

npm install

**3. Running the Application**

**Backend**

cd backend

npm run dev

**Frontend**

cd frontend

npm run dev

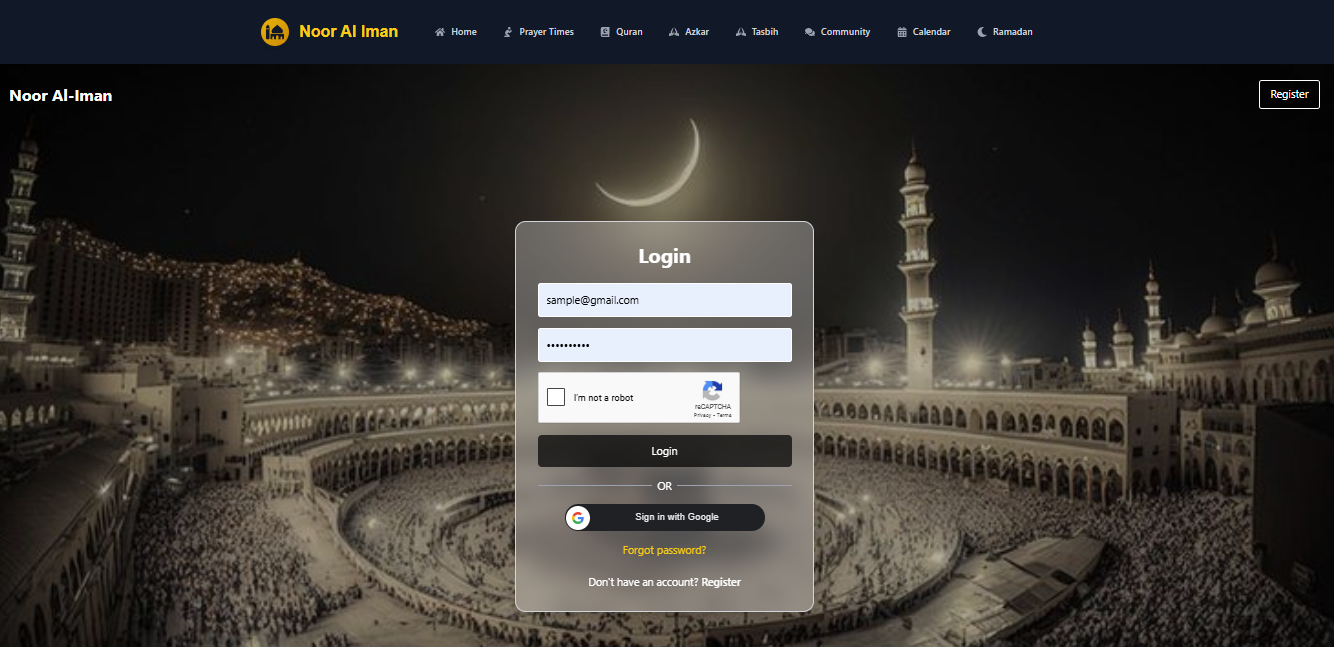
**4. Access the Application**

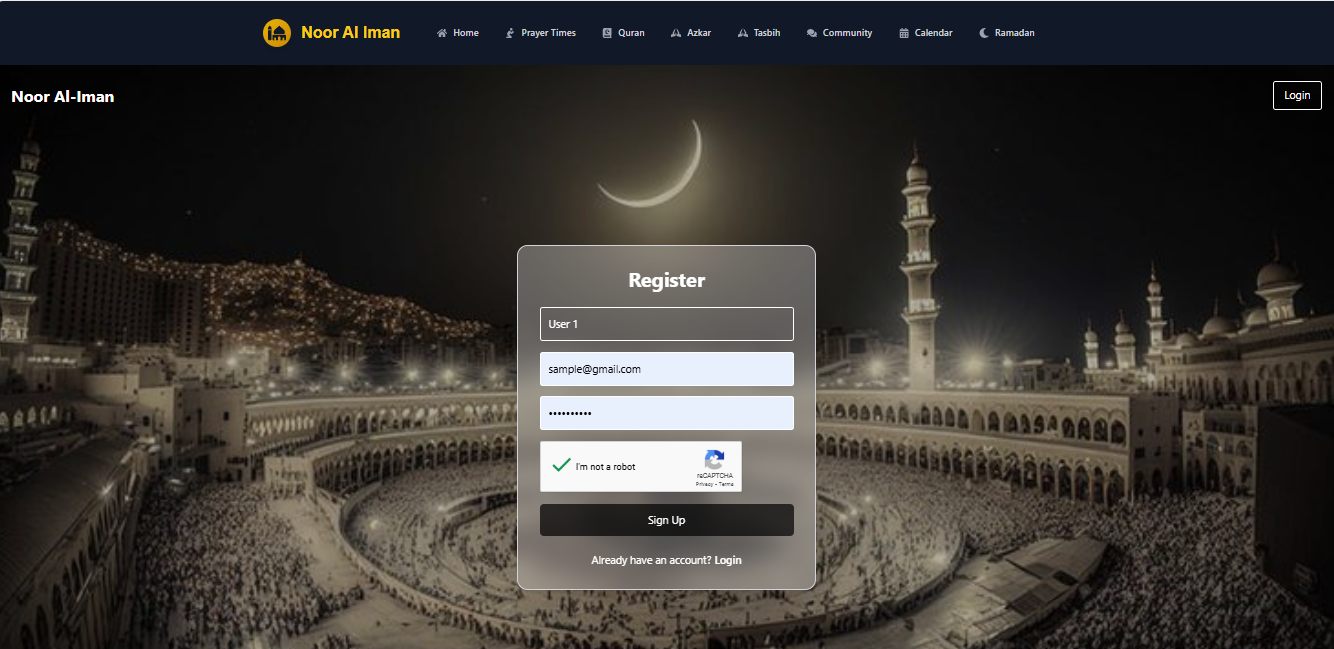
* **Frontend:** <http://localhost:3000>
* **Backend API:** <http://localhost:5000>

Make sure **both servers are running** in separate terminal windows simultaneously.

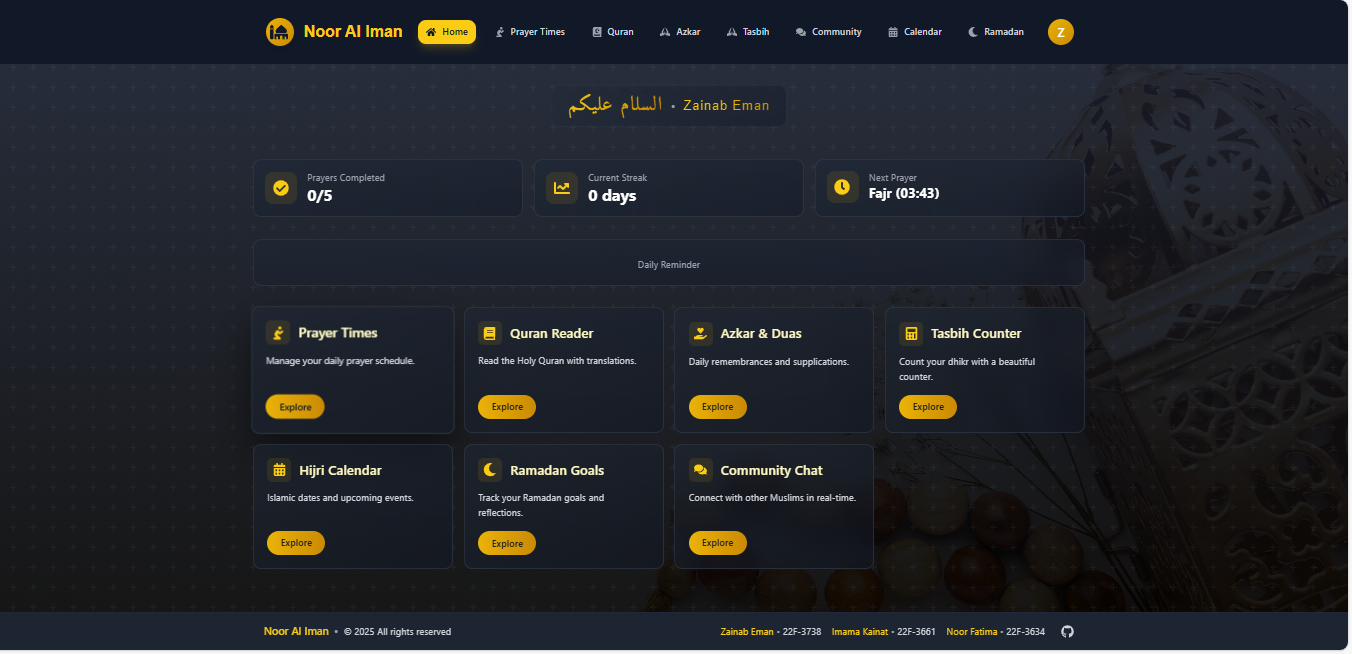
## User Guide

1. **Registering and Logging In**
   * Visit the registration page and enter a valid email and password.
   * Complete the Google reCAPTCHA to confirm human interaction.
   * Upon successful registration, the user receives a confirmation email and is automatically redirected to the dashboard—no need for separate login.
   * Existing users can log in by entering their credentials and completing reCAPTCHA; successful login also redirects to the dashboard.
   * Users may alternatively log in using their Google account through the OAuth sign-in option.
   * If a user forgets their password, they can use the "Forgot Password" link to initiate a reset. A password reset link will be sent via email.
   * Email notifications are triggered on both registration and password reset actions.





1. **Navigating the Dashboard**
   * Upon login or successful registration, the user is taken directly to the main dashboard.
   * The dashboard provides access to all major modules through a clearly structured navigation bar:
     + Qur’an
     + Azkar
     + Namaz Tracker
     + Ramazan Goals
     + Reflections
     + Community Forum

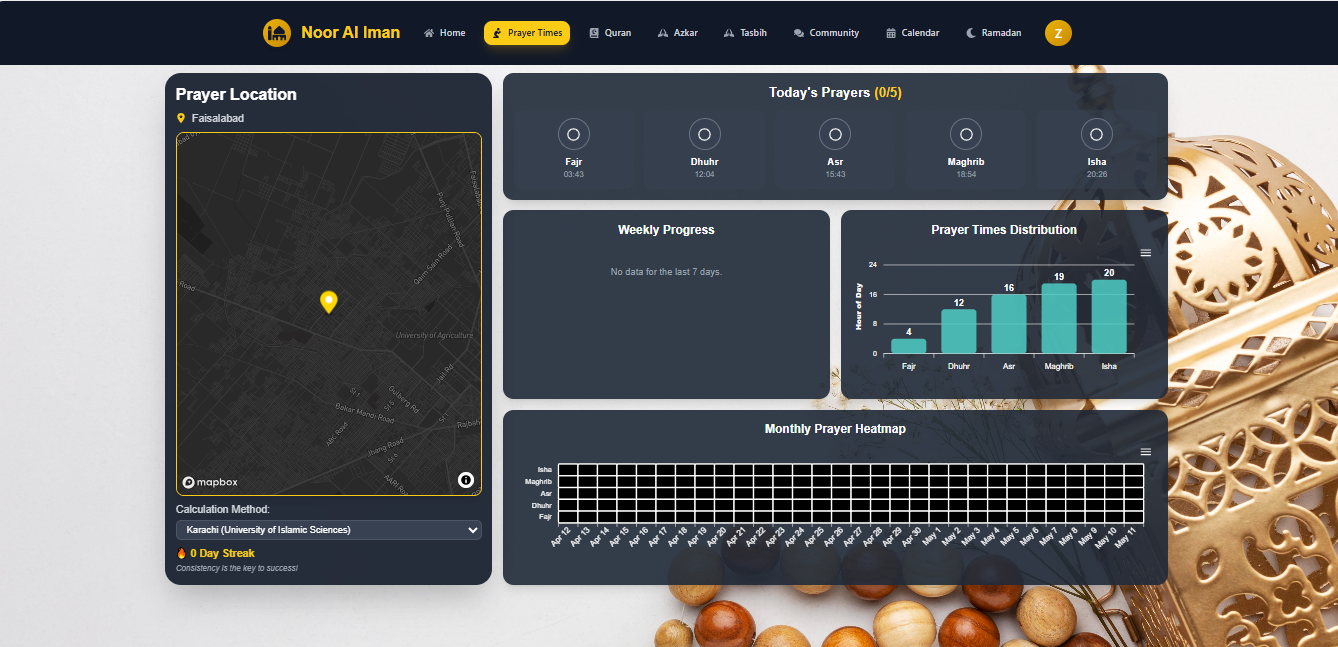
****.

## Features

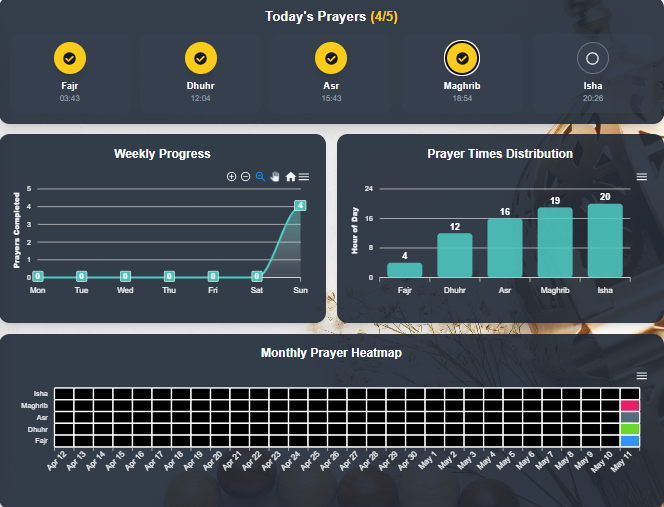
### **Prayer Time**

The Prayer module in Noor Ul Eman offers a complete system for tracking, visualizing, and managing daily Salah activities, personalized to each user's location and usage history.

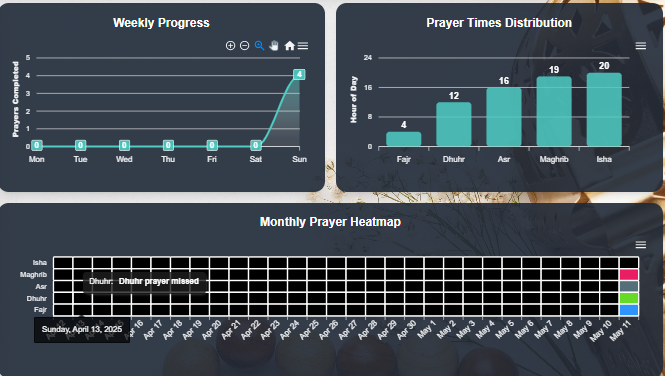
* **Automatic Location Detection**
  + The system detects the user's geographical location using browser-based geolocation services.
  + Once location access is granted, the map is displayed to the user, confirming the detected city or area. User can also manually select the location from the drop down

****

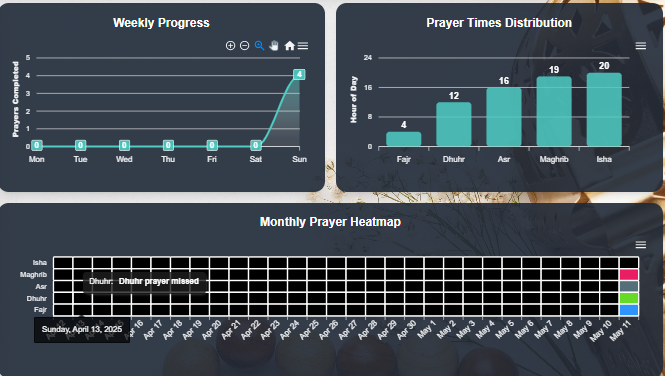
* **Real-Time Prayer Timings**
  + Based on the user's location, daily prayer times (Fajr, Dhuhr, Asr, Maghrib, and Isha) are automatically calculated and displayed.
  + The timings are shown in the main panel for quick reference..
* **Namaz Completion Toggles**
  + At the bottom of the screen, five toggles represent each daily prayer.
  + Users can mark prayers as completed by toggling each corresponding button.
  + This interaction updates the prayer record in real-time using backend API calls.



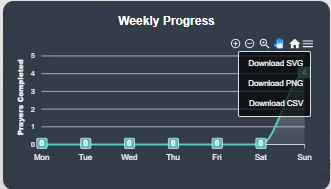
* **Streak and Weekly Progress Visualization**
  + The system maintains a streak count of consecutive days with complete prayers.
  + Weekly progress is visualized using:
    - **Line Charts** – tracking day-wise prayer count.
    - **Bar Charts** – showing comparative volume of completed prayers by type (Fajr to Isha).

****

* **Monthly Heatmap Overview**
  + The heatmap presents a calendar grid where each cell represents one prayer per day.
  + Each of the five daily prayers is shown using a distinct color
  + This layout allows users to quickly assess **which specific prayers** they have consistently offered throughout the month, rather than only the total count.

****

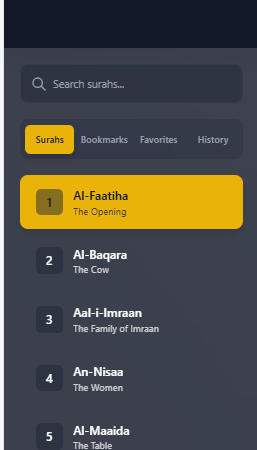
* **Downloadable Reports**
  + Users can export their progress in multiple formats:
    - PNG or SVG for graphical charts.
    - CSV for raw data.
  + This feature supports reflection and reporting for personal accountability.



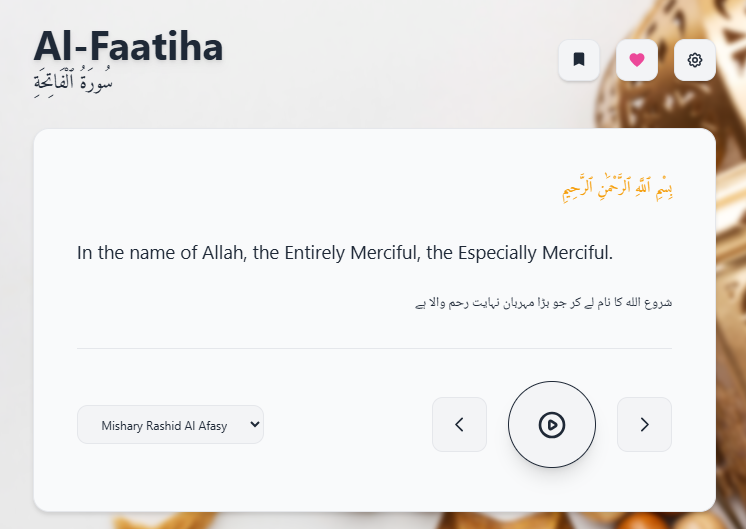
### **Quran**

The Qur’an module in Noor al Eman is designed to offer a highly personalized and accessible recitation and reading experience, complete with navigation, audio playback, customization, and historical tracking features.

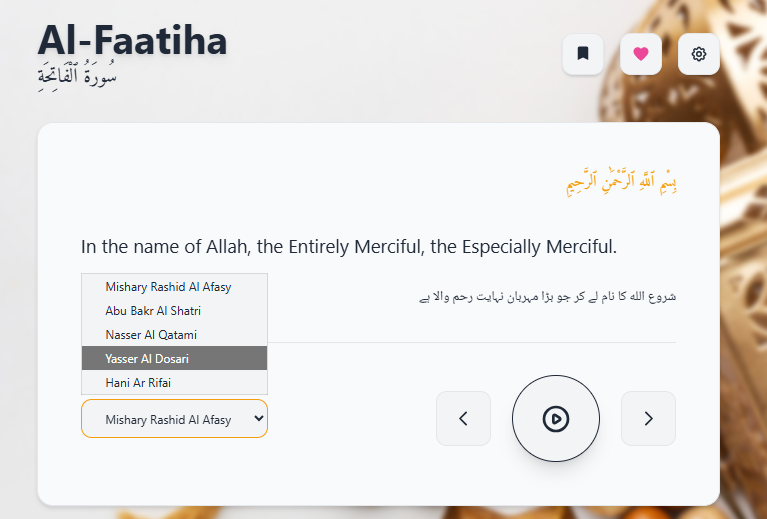
* **Surah Sidebar Navigation**
  + A sidebar lists all **114 Surahs** in both English and Arabic.
  + Users can scroll or use the **search bar** to locate a specific Surah by name.
  + Three additional tabs are provided:
    - **Bookmarks** – displays all ayats bookmarked by the user.
    - **Favourites** – lists user-marked favourite ayats.
    - **History** – tracks the last-read Surah and the last-read Ayat position for easy continuation.

****

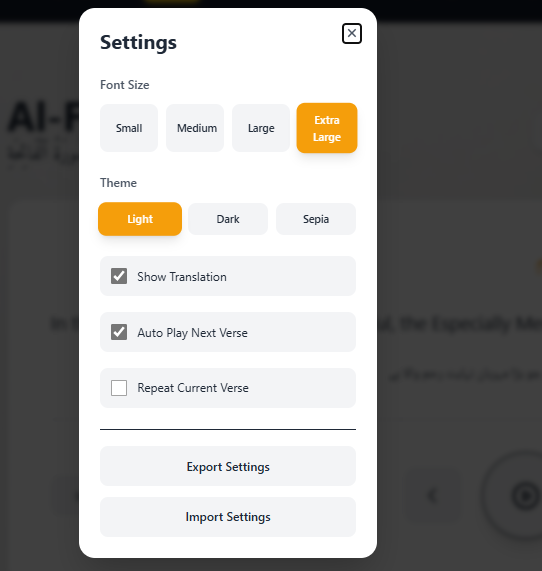
* **Ayah Display Panel**
  + The main screen displays **one Ayat at a time** for focused reading.
  + Each Ayat is presented with:
    - Arabic text
    - English translation
    - Urdu translation
  + **Navigation buttons** allow the user to move forward or backward through Ayats sequentially.
  + The Surah title is displayed at the top in both English and Arabic for contextual clarity.

****

* **Audio Playback Options**
  + A **play button** allows users to listen to the recitation of the currently displayed Ayat.
  + A **Qari selection dropdown** lets users choose from available reciters. Once selected, playback uses the chosen Qari’s voice.

****

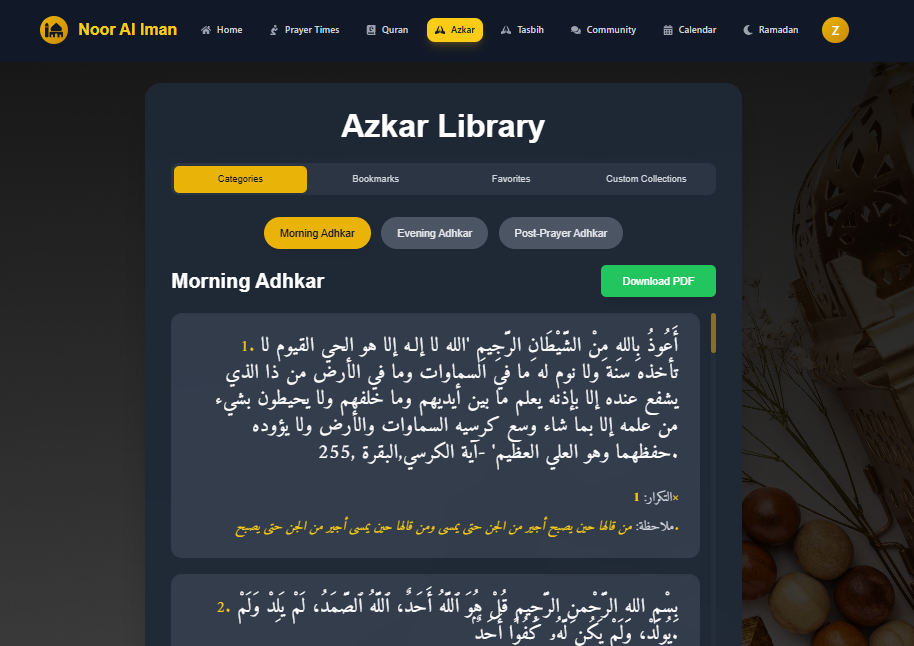
* **Interactive Toolbar (Upper Right)**
  + Located at the top right of the main panel:
    - **Bookmark** – marks the current Ayat for later reference.
    - **Favourite** – saves the Ayat to the user's favourites list.
    - **Settings** – opens the customization panel.
* **Settings Options**
  + **Font Size**
    - Small
    - Medium
    - Large
    - Extra Large
  + **Theme Selection**
    - Light
    - Dark
    - Sepia
  + Toggle to show/hide translations (English and Urdu
  + Auto Play: Automatically move to the next Ayat after recitation
  + Repeat: Continuously loop the current Ayat
  + Export Settings: Download current configuration as a file
  + Import Settings: Upload a configuration file to apply saved preferences



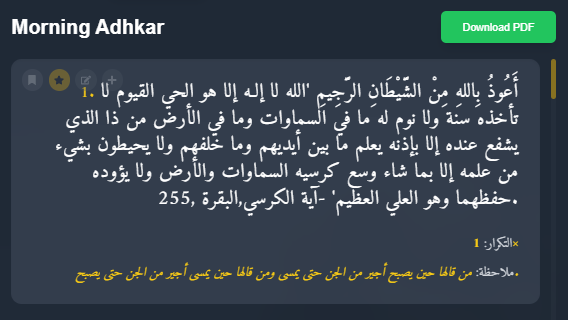
### **Azkar**

The Azkar module in Noor Ul Eman serves as a well-organized digital library of daily supplications, enabling users to read, organize, and personalize their engagement with various types of Azkar. It offers built-in translations, tracking options, and exportable formats for convenience and reflection.

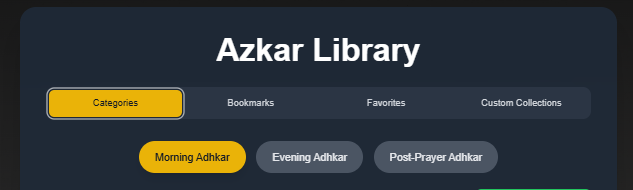
* **Azkar Categories**
  + The library is divided into three primary tabs:
    - **Morning Azkar**
    - **Evening Azkar**
    - **Post-Prayer Azkar**
  + Each tab displays a list of Azkar entries containing:
    - Arabic text of the dhikr
    - Urdu translation
    - The prescribed **count** for each dhikr (number of repetitions recommended)

****

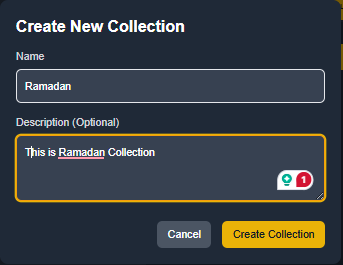
* **Azkar Entry Controls**  
  Each individual dhikr entry provides several interaction options:
  + **Add Note** – Attach a personal note to the dhikr.
  + **Bookmark** – Save for quick access in the bookmarks section.
  + **Favourite** – Mark as a favourite for regular use.
  + **Add to Custom Collection** – Assign to one or more user-defined collections for personalized grouping.

****

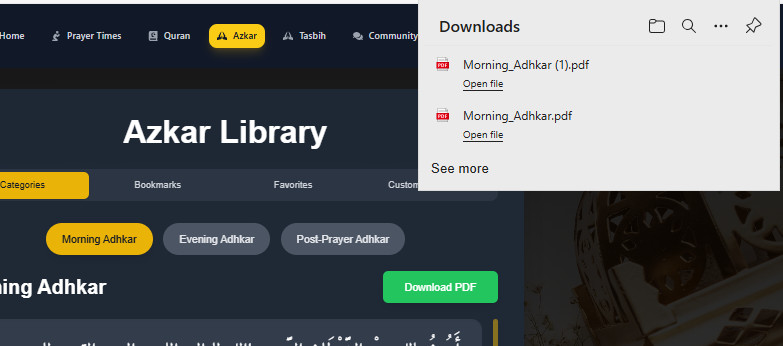
* **Upper Navigation Tabs**  
  The upper section of the Azkar module includes four main navigation tabs:
  + **All** – Displays all Azkar grouped under the three main categories.
  + **Bookmarks** – Shows all Azkar the user has bookmarked.
  + **Favourites** – Lists Azkar marked as favourites.
  + **Custom Collections** – Contains all user-created collections with grouped Azkar.



1. **Custom Collection Management**
   * Users can create collections by specifying:
     + **Name** (e.g., "Before Sleep", "Spiritual Uplift")
     + **Description**
   * While browsing the main Azkar library, any dhikr can be added to one or more collections.



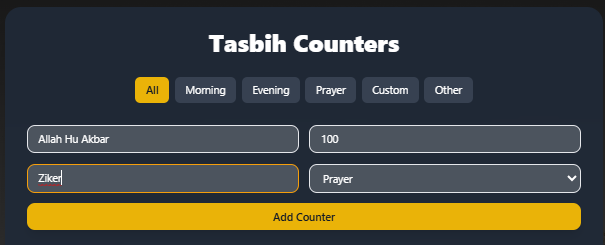
1. **PDF Export Options**
   * Users can export any of the following in **PDF format**:
     + Entire tab (e.g., all Morning Azkar)
     + Bookmark tab
     + Favourite tab
     + Any custom collection
   * This allows offline access and printing for personal or group use

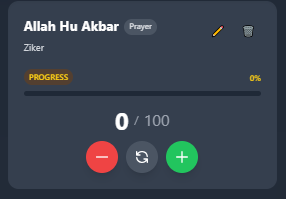


### **Tasbih**

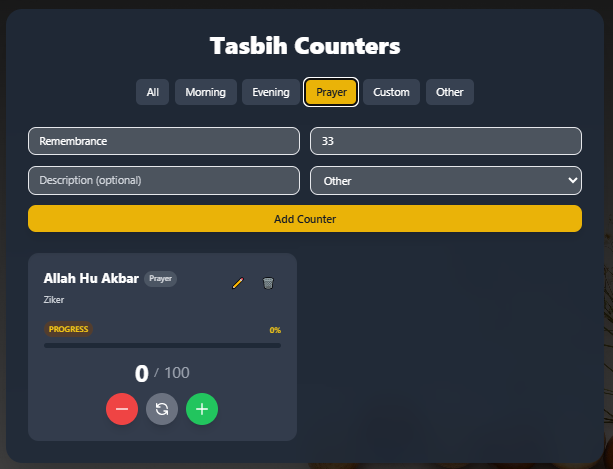
The Tasbih module functions as a digital dhikr counter, allowing users to create, organize, and track their tasbih (remembrance) sessions with ease. It offers both personalization and categorization, supporting a focused and goal-oriented experience for spiritual recitation.

* **Create a New Tasbih**
  + Users can add a tasbih entry by providing:
    - **Name** (e.g., Allahu Akbar, SubhanAllah)
    - **Description** (optional)
    - **Target Count** – the desired number of repetitions
    - **Collection Assignment** – the tasbih can be added to one or more predefined collections for organization
  + Once created, the tasbih will be visible in relevant tabs based on collection type

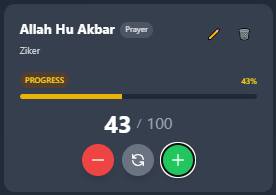
****

****

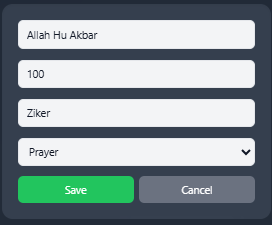
* **Collection-Based Categorization**
  + Tasbih entries can be organized into the following tabs:
    - **All** – displays all tasbih entries created by the user.
    - **Morning**
    - **Evening**
    - **Prayer**
    - **Custom**
    - **Other**
  + Assigning a tasbih to a specific collection automatically classifies it into the corresponding tab.

****

* **Progress Tracking and Interaction**
  + Each tasbih entry includes:
    - A **"+" button** to increment the count.
    - A **"-" button** to decrement the count.
    - A **progress line** that visually tracks the completion percentage.
    - A **percentage label** showing current count vs. target count.
    - A **reset button** to revert the count back to zero.
  + Users can continuously engage with the counter in real-time as they complete dhikr repetitions.

****

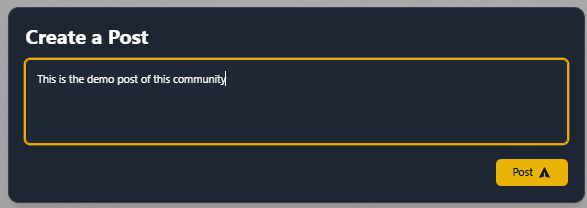
* **Edit and Delete Functionality**
  + Users may modify any existing tasbih entry to:
    - Change the name, target count, or description
    - Move the tasbih to another collection
  + Tasbih entries can also be deleted entirely if no longer needed.

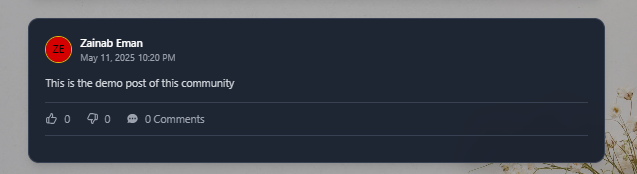


### **Community**

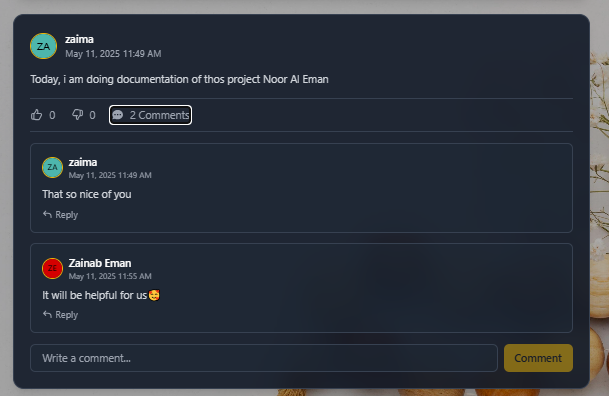
The Community module provides a space for users to share reflections, ask questions, and engage in meaningful discussions with others. It is designed to foster a supportive environment grounded in Islamic values where users can exchange thoughts, seek guidance, or offer advice.

* **Create a Post**
  + Users can write and publish a post directly within the community feed.
  + Posts can include:
    - Reflections or thoughts
    - Questions for the community
    - Any general content suitable for open discussion





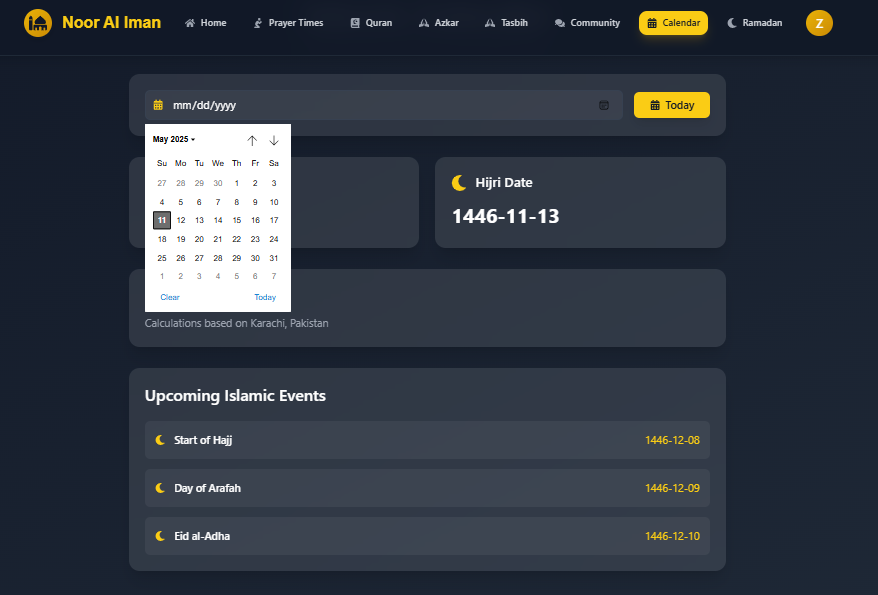
* **Post Visibility and Interaction**
  + Once published, a post is visible to all users in the community.
  + Other users can:
    - **Like** or **Dislike** the post
    - **Comment** on the post to engage in discussion or offer responses
  + Each post displays a comment count, like/dislike count, and timestamp.



### **Calendar**

The Calendar module allows users to view and explore important Islamic dates and events. It integrates both the Gregorian and Hijri calendars and offers a seamless experience for navigating key dates in the Islamic year.

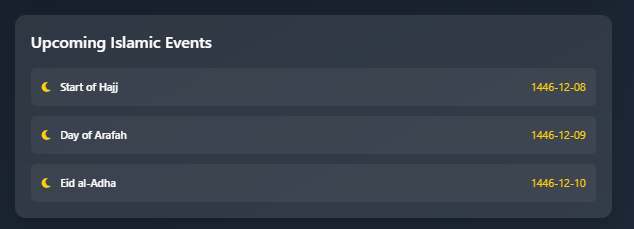
* **Date Selection and Auto-Detection**
  + The module automatically detects and displays the **current date**.
  + Users may also manually **select a different date** using a date picker interface.

****

* **Dual Calendar Display**
  + For the selected date, the system shows:
    - **Gregorian date** (e.g., 11 May 2025)
    - **Hijri date** (e.g., 12 Shawwal 1446 AH)
  + Both formats are displayed prominently at the top of the module.

****

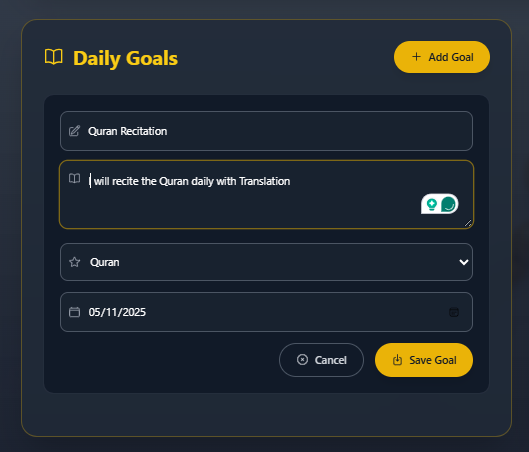
* **Islamic Event Listings**
  + Below the date display, users can view a list of upcoming Islamic events.
  + Each event entry includes:
    - **Event Name** (e.g., Eid al-Adha, Ashura)
    - **Hijri Date**

****

### **Ramadan**

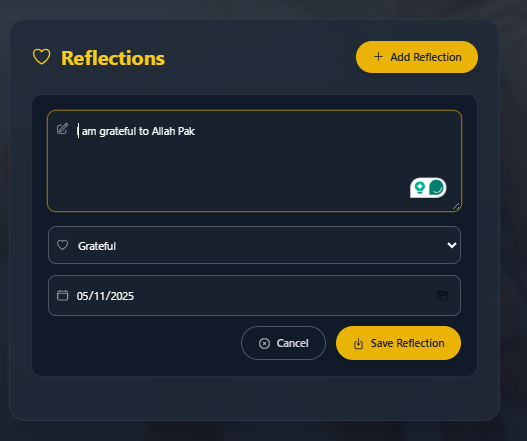
This module is designed to help users plan, track, and reflect on their spiritual growth during the month of Ramazan. It consists of two main sections: **Goals** and **Reflections**.

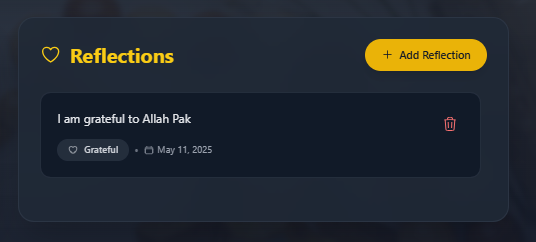
* **Add a Goal**
  + Click on the "Add Goal" tab.
  + Provide the following:
    - **Goal Name** (e.g., "Complete Juz 1", "Donate to charity")
    - **Description** (optional)
    - **Tag** (choose one):
      * Qur’an
      * Charity
      * Salah
      * Fasting
      * Dhikr
      * Other
    - **Target Date** – the intended completion date of the goal
* **Progress Tracking**
  + Goals are categorized as **Pending** or **Completed**.
  + A **Bar Chart** visualizes the distribution of completed vs. pending goals by category.
  + This helps users assess their consistency and achievement throughout the month.

****

****

* **Add a Reflection**
  + Navigate to the "Reflections" tab.
  + Provide the following:
    - **Reflection Title**
    - **Reflection Content** (free text)
    - **Tag** (choose one):
      * Grateful
      * Spiritual
      * Reflective
      * Inspired
      * Peaceful
      * Other
  + Reflections can be viewed in a list format, organized by tag and timestamp, encouraging ongoing self-awareness and contemplation.





## Database Schema

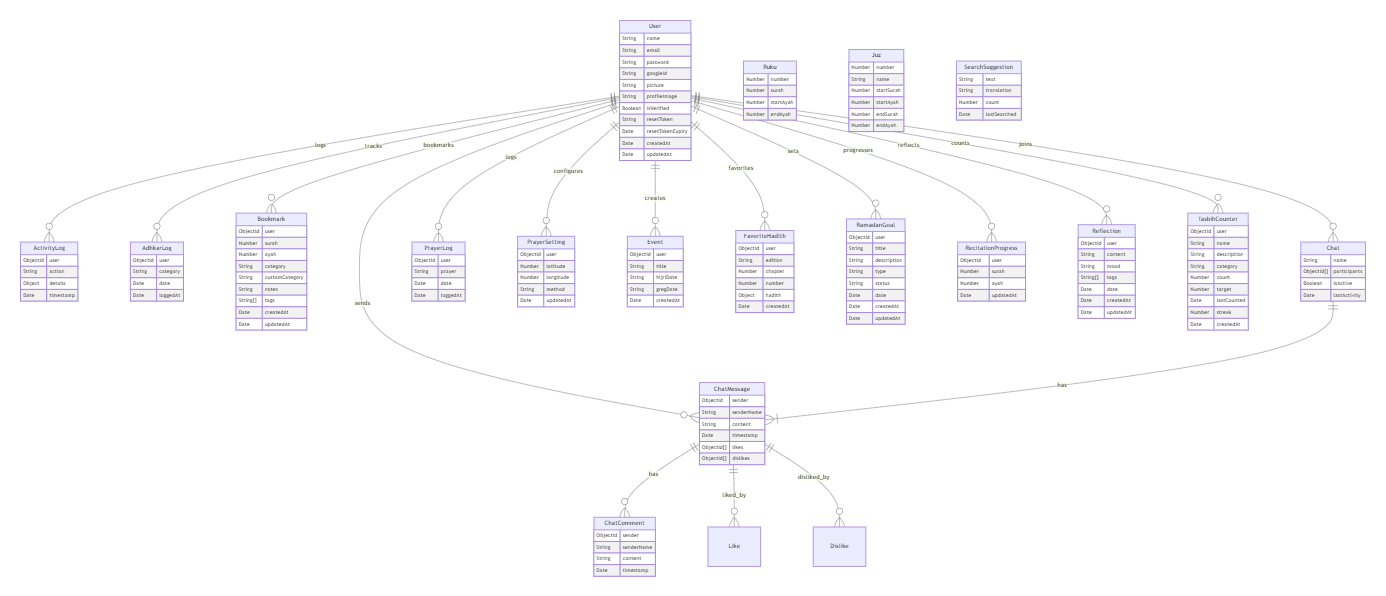
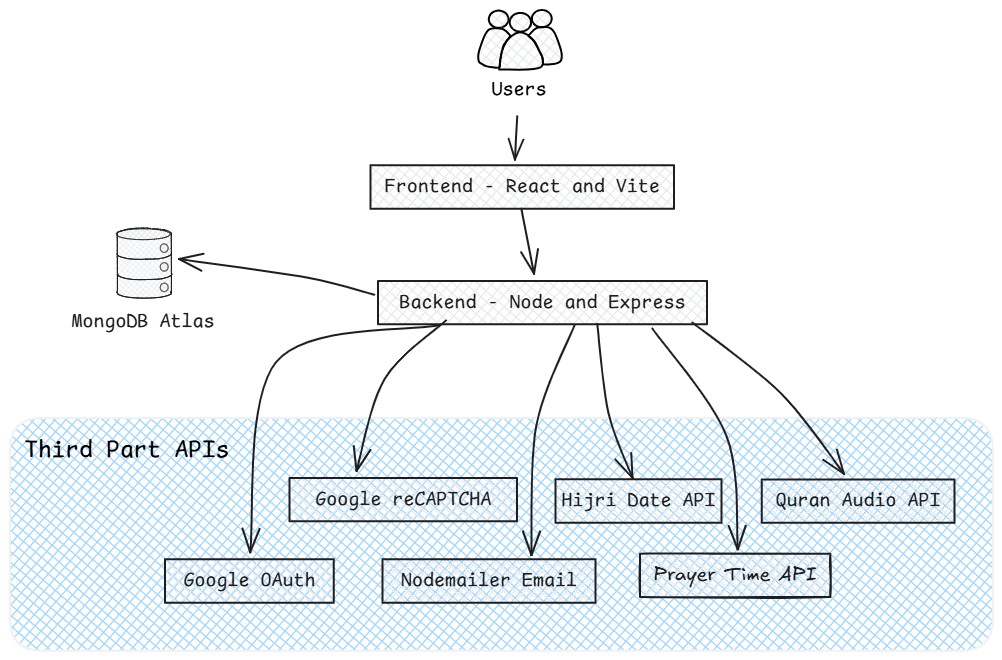


Image added in the Project folder for better visibility

## Architecture Diagram



## ****Advantages****

* The project can grow easily as more users join, thanks to the scalable MERN stack.
* The code is organized in a way that makes it easy to update or fix parts without affecting everything else.
* Features like the Namaz tracker, Tasbih counter, and community forum respond instantly, making the app feel fast and interactive.
* The frontend uses reusable React components, and the backend has clean API routes, making the code easier to maintain.
* It supports smooth deployment with modern tools, so updates and changes can be made without much hassle.

## ****Testing and Validation****

* **Manual Testing**  
  Core functionalities were manually tested for reliability, accuracy, and cross-browser compatibility, including goal creation, Tasbih toggling, and forum interactions.

## ****Deployment (Planned)****

* **Frontend:** Netlify or Vercel — for optimized builds, auto-deployment, and custom domains.
* **Backend:** Heroku or Render — for scalable hosting and environment configuration.
* **Database:** MongoDB Atlas — for secure, cloud-based NoSQL storage with authentication and analytics support.