Requirements Gathering

Project Name: Event Mate – Android Event Countdown App

Delivery Date:

# 1. Stakeholder Analysis

| Stakeholder | Role/Interest | Impact on Project |
| --- | --- | --- |
| End Users (Individuals) | People who will use the app to track personal events (birthdays, meetings, deadlines, etc.). | Their needs define core features, UX/UI, and notifications. |
| Project Owner / Client (Digital Egypt Pioneers Initiative) | The organization requesting and supporting the project. | Provides project requirements, feedback, and evaluation. |
| Development Team | Mobile developers working on the application using Kotlin. | Responsible for the implementation, testing, and delivery. |
| Supervisor / Instructor | Oversees the project development process and provides feedback. | Evaluates progress and ensures project meets the objectives. |

# 2. User Stories & Use Cases

## User Stories (Agile Format):

* As a user, I want to add a new event with a date and time, so that I can track important deadlines.
* As a user, I want to see a countdown timer for each event, so that I know how much time is left.
* As a user, I want to receive notifications before an event happens, so that I don’t miss it.
* As a user, I want to edit or delete events if my plans change.
* As a user, I want to switch between dark mode and light mode, to improve visibility depending on the time of day.
* As a user, I want to choose my preferred language, to make the app easier to use.

## Use Case Example (Add Event):

Use Case Name: Add New Event

Actor: End User

Description: The user adds a new event by entering event details such as name, date, time, and optional notes.

Preconditions: The user has successfully logged into the app (if applicable).

Postconditions: A new event is saved, and the countdown timer starts.

Main Flow: 1. User clicks on 'Add Event' button.  
2. User enters event details (name, date, time, description).  
3. User sets notification preferences (time of reminder).  
4. User saves the event.  
5. Event appears on the home screen with a countdown timer.

# 3. Functional Requirements

* The app shall allow the user to add new events with date, time, and description.
* The app shall display a real-time countdown timer for each event.
* The app shall allow the user to edit or delete existing events.
* The app shall send push notifications/reminders before the event time.
* The app shall allow the user to customize notification timing (e.g., 1 day before, 1 hour before).
* The app shall support Dark Mode and Light Mode.
* The app shall support multiple languages (English and Arabic).
* The app shall persist event data locally (using Room Database or SharedPreferences).
* The app shall provide a simple and intuitive user interface.
* The app shall operate offline (except for updates and cloud backups if added later).

# 4. Non-functional Requirements

| Requirement | Description |
| --- | --- |
| Performance | The app should load screens and perform actions (add/edit/delete events) within 2 seconds. |
| Scalability | Designed to handle unlimited events without performance degradation (efficient data handling). |
| Security | Secure local data storage, ensuring user privacy (data is not shared externally). |
| Usability | The app shall be easy to navigate, with clear instructions and an intuitive UI. |
| Compatibility | Compatible with Android 8.0 (API 26) and above. |
| Reliability | Notifications and reminders should be 95% accurate, triggering at the correct time. |
| Accessibility | Support for Dark Mode, Large Fonts, and Multi-language options for wider accessibility. |
| Battery Efficiency | Optimize background services (notifications) to minimize battery consumption. |