



ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

DEPARTMENT OF SOFTWARE ENGINEERING

Course: Mobile Application Design and Development

Project Title: EVENT MANAGEMENT SYSTEM

| Group Members | ID |
|-------------------------|--------------|
| 1,Fitsum Teka..... | UGR/25652/14 |
| 2.Azariel Tesfaye..... | UGR/25752/14 |
| 3.Yonas Teshome..... | UGR/25573/14 |
| 4.Hibrewerk Demile..... | UGR/26417/14 |
| 5.Robel Michael..... | UGR/23511/13 |
| 6.Muaz Jemal..... | UGR/22544/13 |

Submitted To Mr. Yared Tekalegn

1. Introduction

Welcome to our user-friendly Event Management System app, tailored to simplify event planning and enhance user control. Whether you're organizing a corporate seminar, a concert, or a local gathering, our platform puts powerful tools at your fingertips.

Using our app, you can easily create new events, manage every detail of events you've organized such as editing or deleting event information and conveniently view which events you've reserved spots for. This functionality ensures you stay organized and informed about your event engagements, all from your smartphone.

Each user enjoys full administrative rights over events they create, allowing for seamless updates to event specifics and management of attendee lists. Discover trending events showcased on our homepage, ensuring you're always in the know about popular activities in your community.

In today's digital age, our system facilitates a smooth transition from traditional event planning to a secure, digital platform. Your event data, from attendee details to event analytics, is handled securely and confidentially, ensuring peace of mind.

Our app includes robust login and signup authentication features to protect your personal information and event reservations. With these measures in place, you can confidently engage in event planning and participation, knowing your data is safe.

Whether you're organizing a small gathering or a large-scale event, our user-friendly tools empower you to create memorable experiences effortlessly. Embrace the future of event organization with our advanced Event Management System app.

2. Project Overview

2.1 Statement of the problem

The current manual event management system has done event planning and execution with its fair share of challenges and limitations. In an age where technology is advancing at an unprecedented pace, relying on outdated manual processes can slow down the efficiency and effectiveness of event management system. These problems include:

1. Time-consuming Processes: The manual system involves a lot of paperwork, phone calls, and physical coordination, making event planning and execution a time-consuming. This inefficiency can lead to delays and increasing workload for event organizers.

2. Lack of Real-time Information: Now a days the world is driven by data and real-time insights, the current manual system often fails to provide event organizers with the necessary

information they need to make decisions. This can cause the organizer to adapt to changing circumstances quickly.

3. Limited Scalability: As events grow in size and complexity, the manual system struggles to keep up with the demand. Scaling up operations often means more paperwork, more manual coordination, and an increased risk of errors.

4. Data Management and Analysis Limitations: Manual data entry and record-keeping make it difficult to gather, analyze, and derive meaningful insights from event-related data. This limitation leads to strategic planning and decision-making for future events.

2.2 Scope

The scope of the project includes a wide range of functionalities to address the challenges of the current manual event management system effectively. The new system will provide the following key functionalities:

- 1. User Registration and Login:** Users can register online and log in to the system, creating and managing their accounts.
- 2. Event Creation and Management:** Event organizers user can create, update, and manage events seamlessly, including event details, scheduling, and ticketing options.
- 3. Attendee Registration(Reservation):** Attendees can easily register for events, and see how many peoples will attend the event.
- 4. Feedback and Surveys&Reporting and Analytics:** Organizers can collect valuable feedback from attendees through surveys and feedback forms, helping to improve future events. And Transaction reports and analytics tools are available, enabling organizers to gain insights into event performance and attendee behavior. (any event organizer can get this information from us)
- 5. Payment Processing and Invoicing:** The system supports secure payment processing, allowing attendees to make payments for tickets, and it can generate invoices as needed. (this is a feature that will be added in the future)

3. User Requirement

1. User Sign Up and Login Authentication

- Requirements:
 - Simple and straightforward registration process.
 - Secure login authentication.
- Influence on Design and Functionality:
 - Minimalistic registration form for quick user onboarding.
 - Integration of robust security measures for user login.

2. Managing Events (Edit, Delete)

- Requirements:

- Ability to edit and delete events created by the user.
- Influence on Design and Functionality:
 - Dedicated event management interface for seamless editing and deletion of event details.

3. Creating Events

- Requirements:
 - Input fields for event details (location, date, guests, sponsors, event name, photo).
- Influence on Design and Functionality:
 - User-friendly form for creating events with clear sections for each detail.
 - Option to upload event photos directly from the user's device.

4. Viewing and Managing Reserved Events

- Requirements:
 - Ability to see which events the user has reserved spots for.
- Influence on Design and Functionality:
 - Clear display of reserved events in user profile or dashboard.
 - Option to manage reservations directly from the event details view.

5. Event Details and Reservation

- Requirements:
 - Detailed view of event information including current reservations.
 - Option to reserve a spot for events of interest.
- Influence on Design and Functionality:
 - Comprehensive event detail page showing attendee count and other relevant information.
 - Intuitive reservation process directly from the event details view.

6. User Experience and Interface Design Requirements:

➤ Intuitive and user-friendly interface. ➤ Fast and responsive app performance. ➤ Consistent design language and branding. Influence on Design and Functionality: ✓ Focus on a clean and intuitive UI/UX design. ✓ Optimization for fast load times and smooth navigation. ✓ Consistent use of branding elements throughout the app.

Summary:

The app focuses on providing users with robust features for event management, including secure sign-up and login authentication, seamless event creation and management (editing, deleting), detailed event viewing and reservation capabilities, and clear visibility into reserved events. Each feature is designed to enhance user experience and streamline the process of organizing and participating in events through a user-friendly interface.

4. Design Concepts

We have created a robust, scalable, and user-friendly application utilizing Flutter for the frontend, Appwriter for backend and database. And also, for designing the UI we use a Figma.

1. User Interface (UI):

- **Theme:** Dark theme with a friendly and inviting look.
 - **Color Scheme:** Use soothing yet vibrant colors that work well in a dark interface to enhance user engagement and readability.
- **Components:**
 - **Top Slide Image:** Featuring an image slider at the top of the home screen.
 - **Event Information:** Displaying location name and time of the event overlaid on the slide image.
- **Screens:**
 - **Home Screen:**
 - Dynamic feed of events tailored to the user's preferences.
 - Popular events with many participants displayed prominently.
 - **Event Detail Screen:**
 - Full details of the selected event, including description, time, location, and organizer information.
 - **Profile Screen:**
 - User information, Manage event that u create, and see the event you reserve to attend.
 - **Manage Events Screen:**
 - For event organizers: Manage events created by the user.
 - View and manage reservations and attendee lists.

Features:

- **Responsive Design:**
 - Ensures seamless experience across various devices and screen sizes.
- **Animations and Transitions:**
 - Smooth animations and transitions to enhance user interaction and engagement.
- **Custom Components:**
 - Use custom Flutter widgets to maintain consistency with your dark theme and enhance usability.

Functionality:

- **Event Management:**
 - Ability to create, edit, and manage events.

- View how many people reserve for that event.
- **User Interaction:**
 - Users can reserve spots to attend events.
 - Interactive elements like buttons and forms for event registration and management.

Additional Considerations:

- **Accessibility:** Ensure all elements are accessible in dark mode, with appropriate contrast and readability.
- **Performance:** Optimize for performance, especially with image handling and data retrieval for event listings.
- **Security:** Implement secure authentication and data management practices to protect user information and event data.

2, Backend Architecture Using Appwrite's

Authentication and User Management

- **Appwrite Authentication:**
 - Utilize Appwrite's authentication service for user signup and login.
 - Implement OAuth, email/password, or other authentication methods supported by Appwrite.
 - Securely manage user sessions and authentication tokens.

Data Storage

- **Appwrite Database:**
 - Store user profiles and preferences, including user information and activity.
 - Manage event details such as description, time, location, and participant lists in the Appwrite database.
 - Ensure data integrity and security through Appwrite's database security features.

File Storage

- **Appwrite Storage:**
 - Store event images and other multimedia assets in Appwrite's storage buckets.
 - Manage access controls and permissions for uploaded files.
 - Integrate storage handling within your application to upload, retrieve, and delete files as needed.

Backend APIs and Functions

- **Appwrite API Integration:**
 - Develop custom API endpoints to handle specific app functionalities.
 - Use Appwrite's server-side functions to perform tasks like event management (creation, editing, deletion), user interactions (reservation handling), and data retrieval.
 - Ensure API security and validation to protect against unauthorized access.

Event Management and Integration

- **Event Handling:**
 - Implement logic to manage events created by users, including CRUD operations.
 - Integrate event data with user profiles and preferences for personalized recommendations and notifications.
 - Use Appwrite's real-time capabilities (if available) for updating event details and participant lists in real-time.

5. Development Approach

To address the dynamic nature of the project and adapt to the changing requirements, an Agile software development approach is the optimal choice. Agile software development approach gives advantage in flexibility, collaboration, and iterative development. Here's how the Agile process will be structured:

- **Requirements Gathering and Backlog Creation:** The project will begin with the creation of a product, listing all desired features and functionalities.
- **Sprint Planning:** The project will be divided into short development cycles called sprints, typically lasting two to four weeks. Each sprint will focus on delivering a specific set of features.
- **Continuous Testing and Integration:** Testing will be an integral part of the development process, to ensure that each increment is thoroughly tested and meets quality standards.
- **Customer Feedback and Iteration:** At the end of each iteration, a review with stakeholders will gather feedback to update the Customer requirement Changes and improvements can be made based on this feedback.
- **Release and Deployment:** When sufficient features are developed and tested, they will be released to production, allowing incremental delivery of functionality.
- **Ongoing Improvement:** The Agile process encourages continuous improvement, both in the product and the development process itself.

6. Technological Stack

Building an event management system involves selecting a technological stack that can handle various aspects like frontend development, backend logic, database management, and

deployment. Here's a suitable technological stack for the event management system described:

Frontend Development

- **Framework:**
 - **Flutter:** For building cross-platform native applications with a single codebase.
- **Languages:**
 - **Dart:** Primary language for Flutter app development.
- **UI/UX Framework:**
 - **Material Design for Flutter:** Provides ready-to-use UI components that align with modern design principles.

Backend Development

- **Backend-as-a-Service (BaaS):**
 - **Appwrite:** Provides authentication, database, file storage, and server-side functions.
- **Database:**
 - **Appwrite Database:** Built-in NoSQL database provided by Appwrite.

Cloud Services

- **Cloud Storage:**
 - **Appwrite Storage:** For storing event images and other multimedia assets.

Additional Technologies

- **Version Control:**
 - **Git:** For version control and collaboration.
- **Development Environment:**
 - **Integrated Development Environment (IDE):**
 - **Visual Studio Code (VS Code)** with Dart and Flutter plugins.

Security Considerations

- **Authentication and Authorization:**
 - **Appwrite Authentication:** OAuth, email/password authentication mechanisms.

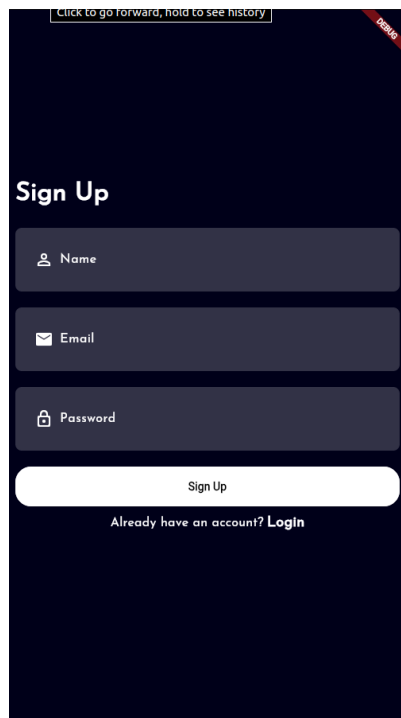
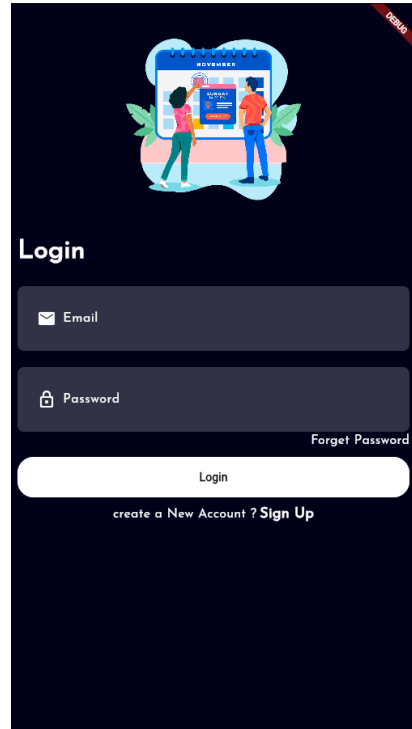
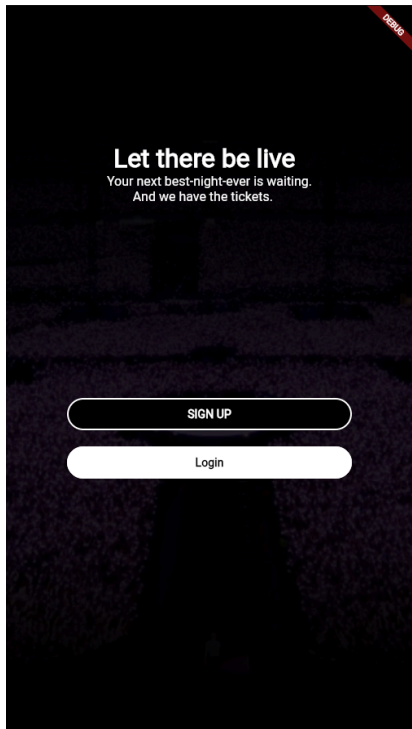
Performance Optimization

- **Caching Mechanisms (if applicable):**
 - Utilize caching for improving app performance, especially for frequently accessed data.

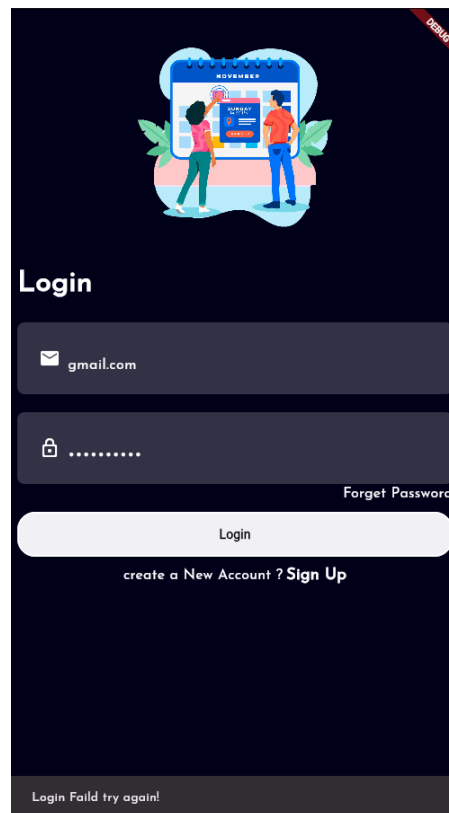
7. Implementation Details:

The app features and functionalities:

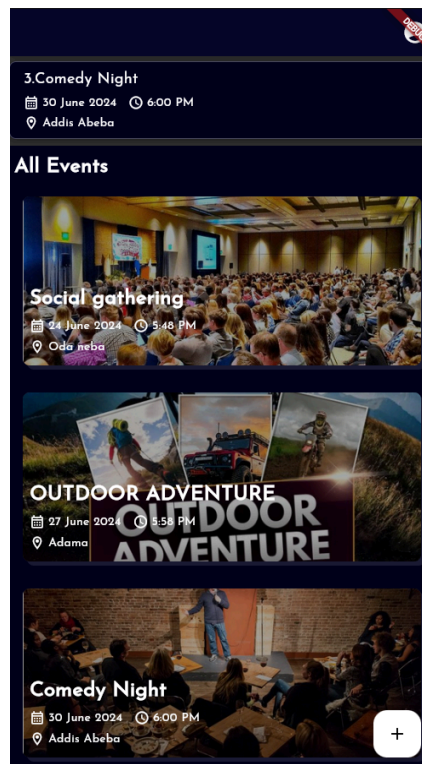
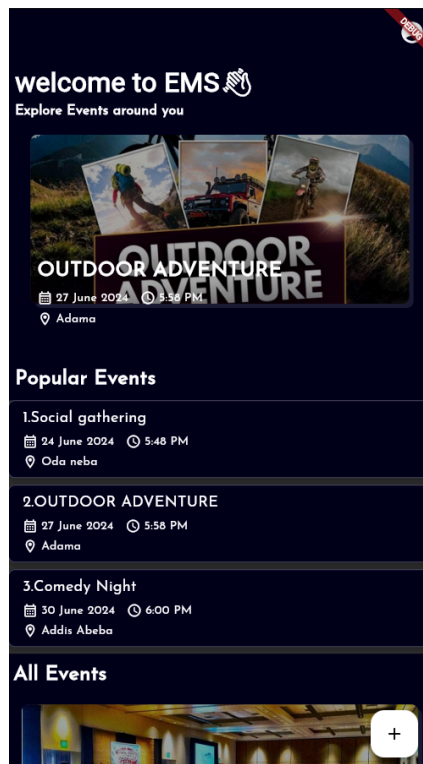
- ✓ customers should create an account to use functionalities of the app otherwise you cannot access anything in the app.
- ✓ customers can login with an account.



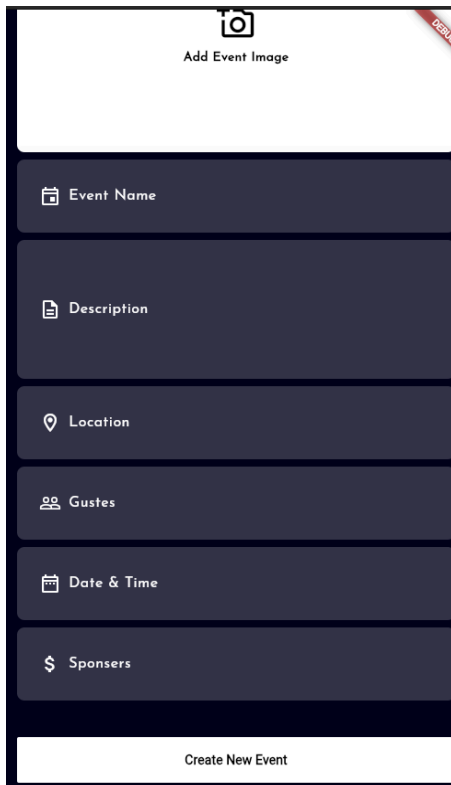
✓ If the customers username already taken when he signup and when username and password not correct in login page, the app shows message



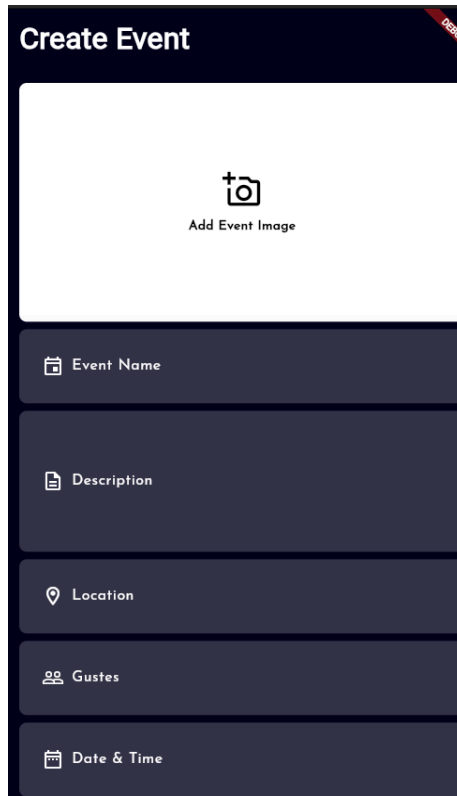
✓ After login a user see a home page with a photo of the event and also name and time another thing is that there is a popular events list that have most or many reservation



✓ the user can create an event and be the admin and manage that event he created.

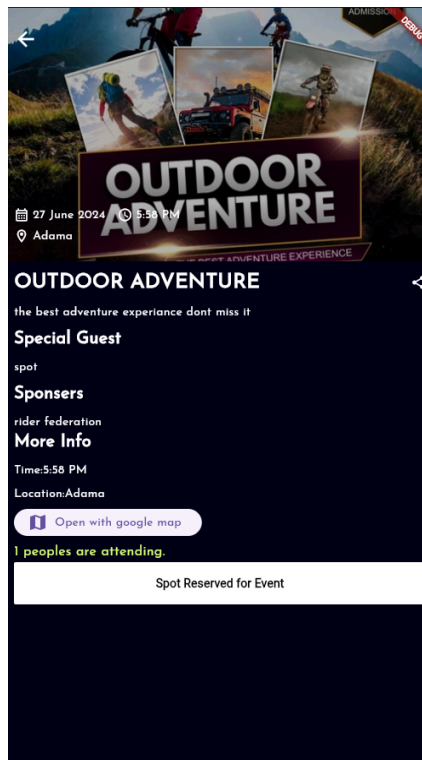


A mobile app interface for creating a new event. It features a dark blue header with a camera icon and the text "Add Event Image". Below the header is a large white rectangular area for the event image. The form consists of several dark blue input fields with white icons and labels: "Event Name" (calendar icon), "Description" (document icon), "Location" (location pin icon), "Gustes" (people icon), "Date & Time" (calendar icon), and "Sponsors" (dollar sign icon). At the bottom is a white button labeled "Create New Event". A red "demo" banner is in the top right corner.



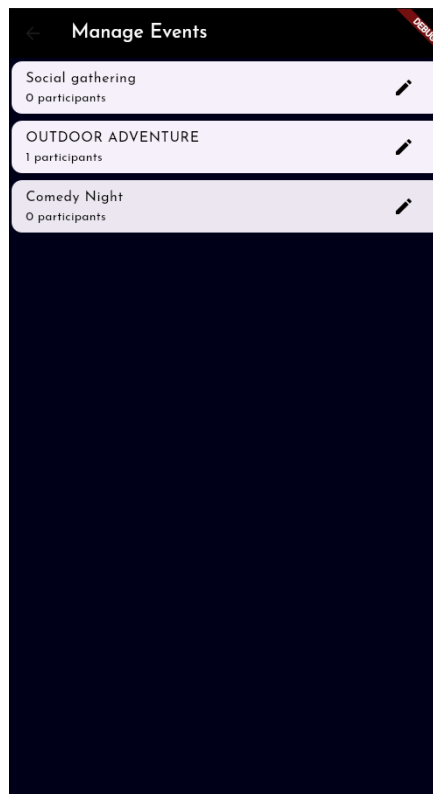
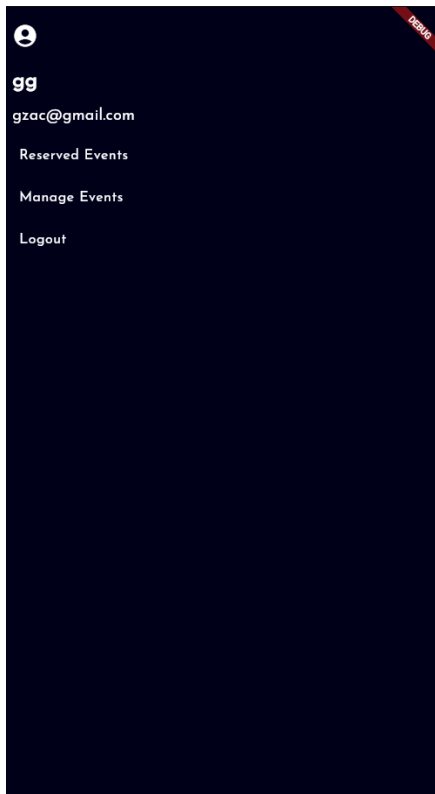
A mobile app interface for creating a new event. It features a dark blue header with the text "Create Event". Below the header is a large white rectangular area for the event image. The form consists of several dark blue input fields with white icons and labels: "Event Name" (calendar icon), "Description" (document icon), "Location" (location pin icon), "Gustes" (people icon), and "Date & Time" (calendar icon). A red "demo" banner is in the top right corner.

✓ the user can see more about the event and make a reservation on the page if he have an interest on that event .



A mobile app interface for an event details page. The header features a collage of outdoor adventure photos and the text "OUTDOOR ADVENTURE". Below the header, the event details are listed: "27 June 2024", "5:58 PM", and "Adama". The event title "OUTDOOR ADVENTURE" is prominently displayed. Below the title, the text "the best adventure experiance dont miss it" is shown. The "Special Guest" section lists "spot" and "Sponsors". The "More Info" section lists "rider federation", "Time:5:58 PM", and "Location:Adama". A button labeled "Open with google map" is present. Below the button, the text "1 peoples are attending." is shown. At the bottom, a white button labeled "Spot Reserved for Event" is displayed. A red "demo" banner is in the top right corner.

✓ the user can see event he reservation and can manage(edit or delete) the event that he create.



8, Testing and Quality Assurance Plan

1. Functional Testing

- **Event Creation and Management:**
 - **Test Scenario:** Create new events with different configurations (e.g., date, location, description).
 - **Expected Outcome:** Events are successfully created and displayed in the app. Verify that editing and deleting events function correctly.
- **Reservation Management:**
 - **Test Scenario:** Reserve spots for events, view reservations, and cancel reservations.
 - **Expected Outcome:** Reservations are accurately recorded and updated in real-time. Ensure notifications (if any) related to reservations work correctly.
- **User Authentication:**
 - **Test Scenario:** Test login and signup functionalities.
 - **Expected Outcome:** Users can create accounts, log in securely, and log out without issues. Verify password recovery/reset processes.

2. Usability Testing

- **User Interface (UI) Testing:**
 - **Test Scenario:** Navigate through the app as a new user.
 - **Expected Outcome:** The UI is intuitive and user-friendly. Ensure consistent design elements and smooth navigation between screens.

3. Performance Testing

- **Response Time Testing:**
 - **Test Scenario:** Measure the app's response times for various actions (e.g., event creation, loading attendee lists).
 - **Expected Outcome:** Ensure that response times are within acceptable limits to provide a smooth user experience.

4. Security Testing

- **Authentication and Authorization:**
 - **Test Scenario:** Attempt unauthorized access to restricted features.
 - **Expected Outcome:** Verify that only authorized users can access administrative functions and sensitive data.

5. Compatibility Testing

- **Device Compatibility:**
 - **Test Scenario**

6. Regression Testing

- **Test Scenario:** Perform regression tests after each app update or bug fix.
- **Expected Outcome:** Ensure that new changes do not introduce new bugs or affect existing functionalities negatively.

7. Documentation and Training

- **Documentation Review:**
 - Review and update user manuals, help guides, and API documentation.
 - Ensure they accurately reflect the current app features and functionalities.

8. User Acceptance Testing (UAT)

- **Test Scenario:** Conduct UAT with a select group of users or stakeholders.
- **Expected Outcome:** Gather feedback on usability, functionality, and overall user satisfaction. Address any issues or concerns raised during UAT.

8.2 Reflection on the Effectiveness of Our Testing Approach

The testing strategies employed for our Event Management System were pivotal in ensuring its functionality, performance, and reliability. Here's a reflection on the overall effectiveness:

- **Early Bug Detection:**
 - Unit testing played a crucial role in identifying and rectifying bugs early in the development cycle. This proactive approach significantly reduced the time and effort required for bug fixing later stages of development.
- **Seamless Integration:**
 - Integration tests were instrumental in validating that various components of the system, such as event creation, management, and user authentication, worked harmoniously together. This ensured a seamless user experience without disruptions or inconsistencies.
- **Optimized Performance:**
 - Performance testing allowed us to pinpoint and address potential bottlenecks in the system. By optimizing load times and enhancing responsiveness, we improved user satisfaction and engagement with the app, especially during peak usage periods.
- **User Satisfaction:**
 - Usability testing provided invaluable insights from real users, enabling us to refine the app's interface and functionality based on direct feedback. This iterative process enhanced the overall user experience, making the app more intuitive and user-friendly.

9. Future Enhancements

Payment Integration:

- **Chapa and Sntim Pay Integration:** Integrate local payment gateways such as Chapa or Sntim Pay to facilitate secure online payments for event registrations, ticket purchases, and other transactions.

Location Services:

- **Google Maps Integration:**
- **Event Venue Mapping:** Utilize Google Maps API to display interactive maps of event venues, including directions, nearby amenities, and parking information.
- **Features:**
 - Enable users to view event locations with satellite imagery, street view, and real-time traffic updates.
 - Provide geolocation services for attendees to navigate to event venues using their mobile devices.
 - Integrate event-specific markers, labels, and custom overlays to enhance navigation clarity.

Mobile App Enhancements:

- **Offline Mode:** Implement offline capabilities for users to access and manage event information without an internet connection.
- **Push Notifications:** Integrate push notifications to alert users about event updates, new reservations, or important announcements.

Event Creation and Management:

- **Customizable Templates:** Offer predefined event templates for common event types (e.g., conferences, workshops) to streamline the creation process.
- **Recurring Events:** Enable users to create recurring events with options for daily, weekly, monthly schedules.

Attendee Management:

- **Check-In System:** Develop a mobile check-in feature for event organizers to track attendee arrivals and manage event logistics in real-time.
- **Feedback Collection:** Integrate post-event surveys or feedback forms to gather attendee insights and improve future event planning.

User Experience Enhancements:

- **Personalized Dashboard:** Provide customizable dashboards where users can prioritize and track their upcoming events, reservations, and tasks.

Security and Privacy:

- **Two-Factor Authentication (2FA):** Strengthen account security with 2FA options to protect user accounts from unauthorized access.

Analytics and Reporting:

- **Event Analytics:** Introduce advanced analytics tools to provide organizers with insights into attendee demographics, engagement metrics, and event success indicators.
- **Custom Reports:** Enable users to generate custom reports based on specific event data criteria and export them for further analysis.

Continuous Improvement and Feedback:

- **User Feedback Mechanisms:** Establish channels for continuous user feedback and feature requests to prioritize enhancements based on user needs and preferences.
- **Agile Development Practices:** Adopt agile methodologies to facilitate iterative updates and quick deployment of new features and improvements.