Meet up Application

Author: Nikhil Amin, Yash Revannavar, Mohit Kokate, Aditya Kumar, and Pramod Shirale

Revision: A

Document approval

| Name | Role | Signature/date |
| --- | --- | --- |
| Mr. Thomas Berger | Invigilator |  |

Previous Revisions:

Content

[1](#_heading=h.gjdgxs) Content 2

2 Change Log 3

3 [Purpose & Scope 3](#_heading=h.30j0zll)

4References 3

5 Users/actors 3

5[.1](#_heading=h.3znysh7) Guest User 3

5[.2](#_heading=h.2et92p0) Registered user 3

53 Host 3

5.4 Attendees 4

6System Overview and supported platforms 4

6[.1](#_heading=h.3dy6vkm) System Overview 4

6[.2](#_heading=h.1t3h5sf) Supported Platforms 4

7Requirements 5

7.1 Platform and Technologies 5

7.2 User Interface5

7.3 User Management5

7.4 Meetup Events5

7.5 Notifications and Reminders5

7.6 Meetup Groups 5

7.7 Meetup Reviews and Ratings 6

7.8 Security 6

7.9 Performance and Scalability 6

7.10 Integration 6

7.11 Testing and Quality Assurance 6

7.12 Deployment and Maintenance 6

8 [Requirement Tracing Matrix](#_heading=h.lnxbz9) 7

# 2 Change Log

| Revision | Description of changes |
| --- | --- |
| A | Initial version |
|  |  |

# 3 Purpose & Scope

The goal of this Technical Specification Document (TSD) is mapping product requirements to a technical specification document using a Tracing Matrix which can help to ensure that all the necessary features and functionalities are covered in the technical paper.

To ensure that the development team grasps understands and accurately translates the functional requirements of the application into a detailed technical design, the product requirement document (PRD) for a Meetup application is mapped into a technical specification document (TSD) using a tracing matrix. This procedure helps in bridging the gap between the technical implementation and the business or product needs.

The goal or scope of this mapping process is to translate the high-level product requirements given in the PRD into precise technical elements and functionality. The Tracing Matrix is a tool for establishing traceability between the technical requirements and the requirements for the product.

# 4 References

## [PRD-001 “Meet Up Application” Rev. A](https://docs.google.com/document/d/1Usw7hDBdg2BfkcIbzDQFJT1UioYPHSz6/edit?usp=drive_link&ouid=103163737203738927757&rtpof=true&sd=true)

# 5 User Actors

# 5.1 Guest User: The guest user is able to use the application without having been registered. Guest users should be able to explore the meetup through the application but will not be allowed to join the event until register. Services include: Changing the location, Searching for the events in the selected area and filtering out based on the days and time, Find/exploring groups.

5.2 Registered user: The registered user is registered in the Application with the email id and more services are offered if the user is signed in. In order to sign in to the application user has to log in with the registered email id and password.  
Services include: Changing/Browse the location, Searching for the events in the selected area and filtering out based on the days and time, Find/exploring groups to join, Browse previously attended events, Sharing events with other registered users, Make a group with other registered users, Able to access the “Notification tab”, Able to access the “Messages tab”, Attend the events, Host the events.

5.3 Host: The registered user can be the host to create an event or group and invite other registered users to join the event. Services include: Host an event, and Create a group.

5.4 Attendees: The registered user can be an attendee to attend an event and invite other registered users to join the event. Services include: attending an event, Creating a group, Share the event with other registered users.

# 6 System Overview and Supported Platforms

6.1 System Overview:   
Meetup is a social media platform for organizing and planning in-person or virtual activities, get-togethers, and events for people and communities with similar interests, hobbies, and jobs. Users must download the Meetup application from the Google Play Store or the Apple Application Store. The user/host must then log in to the application or website using the login/sign-up option. The user/host will be prompted to create their profile after successfully logging into the website or application. Based on the user's interests, the user will receive all events that are taking place near them and at the same time can become the host of events that the user needs to create. The user or host can also invite their friends to the event by sharing it on the website or application.

After registering for a specific event, the host or user can view all event details. The user can check the availability of seats by clicking on the booking option and he can see all the details of the visitors attending the event. If event seats require a payment method, the user will be redirected to the payment gateway and will need to make a payment to confirm their seat. If the payment is successful, a boarding pass or ticket for the event the user is attending will be issued.  
  
These are some commonly used interfaces, services, backend and frontend that could be employed in the Meetup application.   
Frontend(UI): HTML, CSS, JavaScript

Application Programming Interface (API): Email service provider, Search engine Elasticsearch, Google Maps API), Calendar library, Firebase Cloud, Messaging Notification

Backend: Node.js, Ruby on Rails, Django, Java Spring, ASP.NET, etc.

Databases: MySQL, PostgreSQL, MongoDB, Redis, etc.

Plan and client Encounter (UX): Custom plan and UX strategies particular to the improvement group. e.g. Figma, Sketch, Adobe XD, Zeplin, Marvel, etc

**6.2 Supported Platforms:**   
Operating systems:

* iOS
* Android
* Windows

Web Browsers:

* Google Chrome
* Mozilla Firefox
* Safari
* Microsoft Edge

# 7 Requirements

These technical specification requirements provide a broad overview of the various aspects that need to be considered when developing a Meetup application.

**7.1 Platform and Technologies:**  
TS-001: Specify the platform(s) the application will run on (e.g., web, mobile, both).  
TS-002: Identify the programming languages, frameworks, and libraries to be used.  
TS-003: Define the database technology and any specific requirements for data storage.

**7.2 User Interface:**  
TS-004: Design an intuitive and user-friendly interface for easy navigation and interaction.  
TS-005: Ensure the interface is responsive and compatible with different devices and screen sizes.  
TS-006: Implement a visually appealing design consistent with the application's branding.

**7.3 User Management:**  
TS-007: Develop user registration and login functionality with proper authentication and validation.  
TS-008: Implement user profile management features, allowing users to edit their information and upload profile pictures.  
TS-009: Enable social media integration for streamlined user registration and login.

**7.4 Meetup Events:**  
TS-010: Implement features for creating, editing, and deleting meetup events through backend API endpoints for creating meetup events.  
TS-011: Enable users to search for meetup events based on location, date, category, or keywords e.g. Implement search functionality using Elasticsearch for meetup events.  
TS-012: Implement RSVP functionality, allowing users to indicate their attendance for specific meetup events. So, develop frontend functionality for users to RSVP to meetup events.  
TS-013: Include features for users to view event details, such as date, time, location, and organizer information using frontend functionality to allow users to edit their meetup event details, etc

**7.5 Notifications and Reminders:**  
TS-014: Implement a notification system to send reminders and updates to users about upcoming meetup events. So, integrate push notification service to send reminders for upcoming meetups.  
TS-015: Enable users to customize their notification preferences, such as choosing email or push notifications.  
TS-016: Send automated notifications for event updates, cancellations, or changes.

**7.6 Meetup Groups:**  
TS-017: Develop features for users to create, join, and manage meetup groups. So, create backend API endpoints for users to join and manage meetup groups.  
TS-018: Implement validation checks to ensure meetup events do not exceed the maximum capacity. Enable group organizers to schedule and manage events within their groups.  
TS-018: Implement features for group discussions, member invitations, and group announcements.

**7.7 Meetup Reviews and Ratings:**  
TS-019: Allow users to rate and review meetup events they have attended. For this, a database schema and API endpoints serves for users to rate and review meetups.  
TS-020: Implement a rating system to collect and display average ratings for each meetup event.  
TS-021: Enable users to view and sort meetup events based on ratings and reviews.

**7.8 Security:**  
TS-022: Implement secure user authentication mechanisms, such as encryption and hashing of passwords. So, Implement this using OAuth 2.0 and JWT token.  
TS-023: Ensure secure communication between the application and users, using HTTPS or other secure protocols.  
TS-024: Apply proper authorization and access control to protect user data and prevent unauthorized access.  
  
**7.9 Performance and Scalability:**  
TS-025: Optimize the application's performance to ensure fast loading times and a smooth user experience.  
TS-026: Design the application to handle high user traffic and large amounts of data.  
TS-027: Implement caching mechanisms to improve performance and reduce server load.

**7.10 Integration:**   
TS-028: Integrate with mapping services to display event locations and provide directions. So, design a database schema to store location and time information for meetup events.  
TS-029: Enable social media sharing and integration for users to share events and invite friends.  
TS-030: Integrate with third-party services for features like payment processing or ticketing.

7.11 Testing and Quality Assurance:  
TS-031: Perform thorough testing of all features and functionalities to ensure proper functionality and usability.  
TS-032: Conduct security testing to identify and address any vulnerabilities.  
TS-033: Implement error handling and logging mechanisms to track and resolve issues.

**7.12 Deployment and Maintenance:**  
TS-034: Define the deployment process, including server configuration and hosting requirements.  
TS-035: Plan for regular maintenance and updates to ensure the application remains secure and up-to-date.  
TS-036: Implement monitoring and analytics tools to track application performance and user behavior.

# 8 Requirement Tracing Matrix

| **Product Requirement** | **Technical Requirements** |
| --- | --- |
| PRD-001.REQ-001: Users should be able to create a new meetup event.  PRD-001. REQ-002: Users should be able to RSVP to a meetup event.  PRD-001.REQ-003: Users should be able to search for meetup events.  PRD-001.REQ-004: Users should receive notifications for upcoming meetups.  PRD-001.REQ-005: Meetup details should include location and time.  PRD-001.REQ-006: Users should be able to join meetup groups.  PRD-001.REQ-007: Meetup events should have a maximum capacity.  PRD-001.REQ-008: Users should be able to edit their meetup event details.  PRD-001.REQ-009: Users should be able to rate and review meetups.  PRD-001.REQ-010: The application should have a user authentication system. | TS-010: Implement backend API endpoint for creating meetup events.  TS-012: Develop frontend functionality for users to RSVP to meetup events.  TS-011: Implement search functionality using Elasticsearch for meetup events.  TS-014: Integrate push notification service to send reminders for upcoming meetups.  TS-028: Design database schema to store location and time information for meetup events.  TS-017: Create backend API endpoints for users to join and manage meetup groups.  TS-018: Implement validation checks to ensure meetup events do not exceed the maximum capacity.  TS-013: Develop frontend functionality to allow users to edit their meetup event details.  TS-019: Create database schema and API endpoints for users to rate and review meetups.  TS-022: Implement user authentication using OAuth 2.0 and JWT tokens. |