TechMeet Documentation

Introduction

This documentation serves as the foundation for the development of TechMeet, a platform designed for discovering, sourcing, and booking technology-related events. It aims to help developers understand the overall architecture and design of the application, including its core entities, relationships, and key functionalities.

TechMeet empowers users—particularly those in the tech community—to explore upcoming events, save and manage their interests, and connect with others. The application integrates with third-party APIs to fetch events, supports user-specific tagging and categorisation, and facilitates event bookings and networking.

Core Entities

1. **Users**
   - Can register, authenticate (including via OAuth), and manage their profile
   - Can follow or connect with other users

2. **Events**
   - Sourced either manually or through external APIs (via the SourceAPI and OAuthConnection models)
   - Include fields like name, date, location, description, and tags

3. **Tags**
   - Represent user-defined or system-generated labels such as "AI", "Blockchain", or "Women in Tech"
   - Linked to events via a many-to-many relationship
   - Help organise events and support personalised discovery

4. **Categories**
   - Group related tags under broader umbrellas such as “Technology”, “Career”, or “Inclusion”
   - Allow users to filter events through category-tag-event hierarchies
   - Connected to tags through a many-to-many CategoryTag join table

5. **UserEvents**
   - A join table representing a user's relationship to specific events (e.g., saved, attending, completed)
   - Supports tracking progress and user-specific event interactions

6. **Connections**
   - Allow users to form networking relationships (pending, accepted, rejected)
   - Enhance the social aspect of the platform

7. **SourceAPI + OAuthConnection**
   - Handle the integration with external platforms like Eventbrite or Meetup
   - Store authentication tokens and metadata needed to fetch events

Package Diagram

The package diagram is a high-level overview of the application's layers and their relationships.

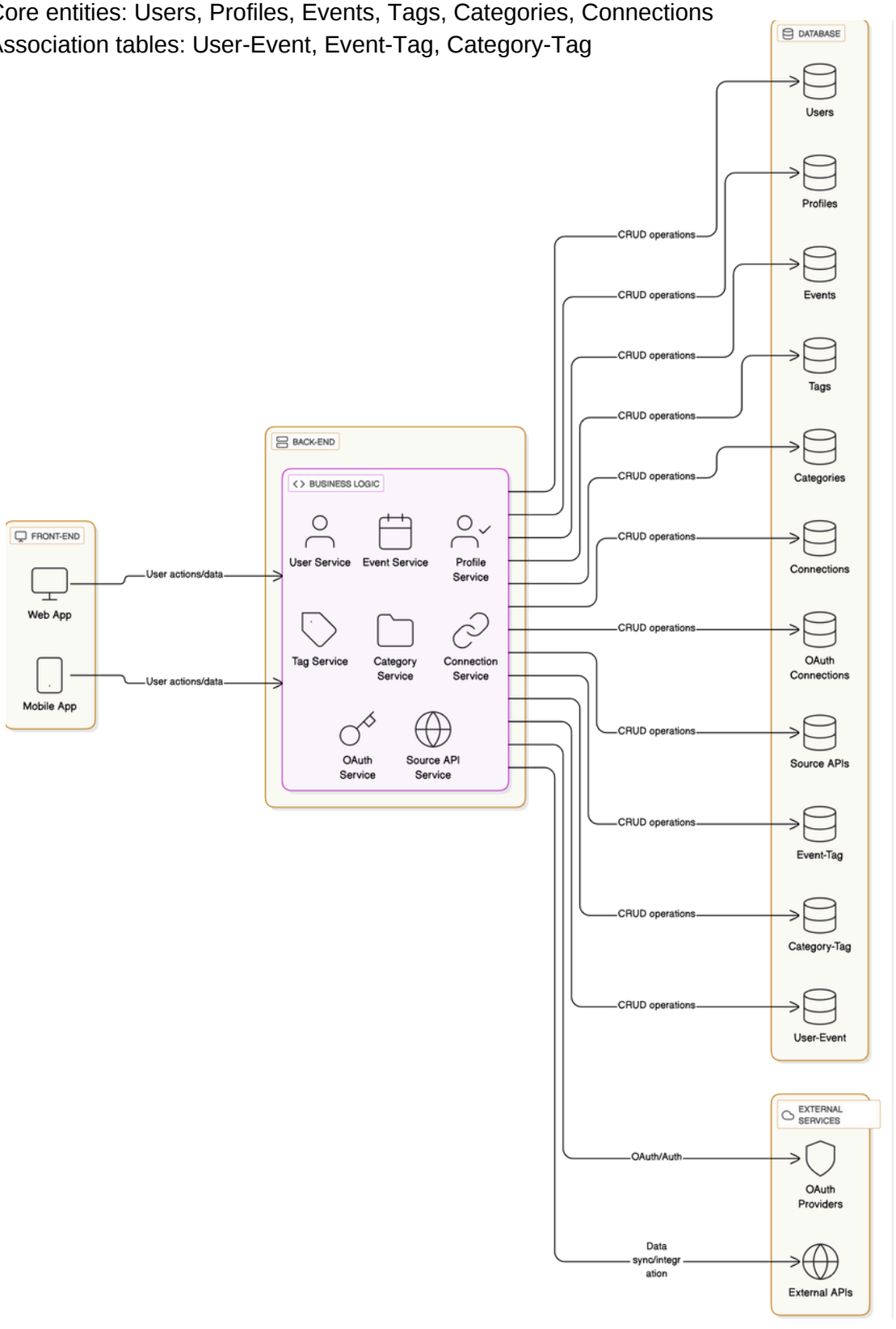
1. **Presentation Layer (Services):** This layer handles interaction between the user and the application. It includes the services and APIs that are exposed to the users via:
   - Web App and Mobile App as clients.
   - Backend-facing services such as

2. Business Logic Layer (Models): This layer contains the core application logic and model representations of system entities, including:

- User, Profile, Event, Tag, Category, and Connection
- Logical processing of OAuth workflows and API data ingestion

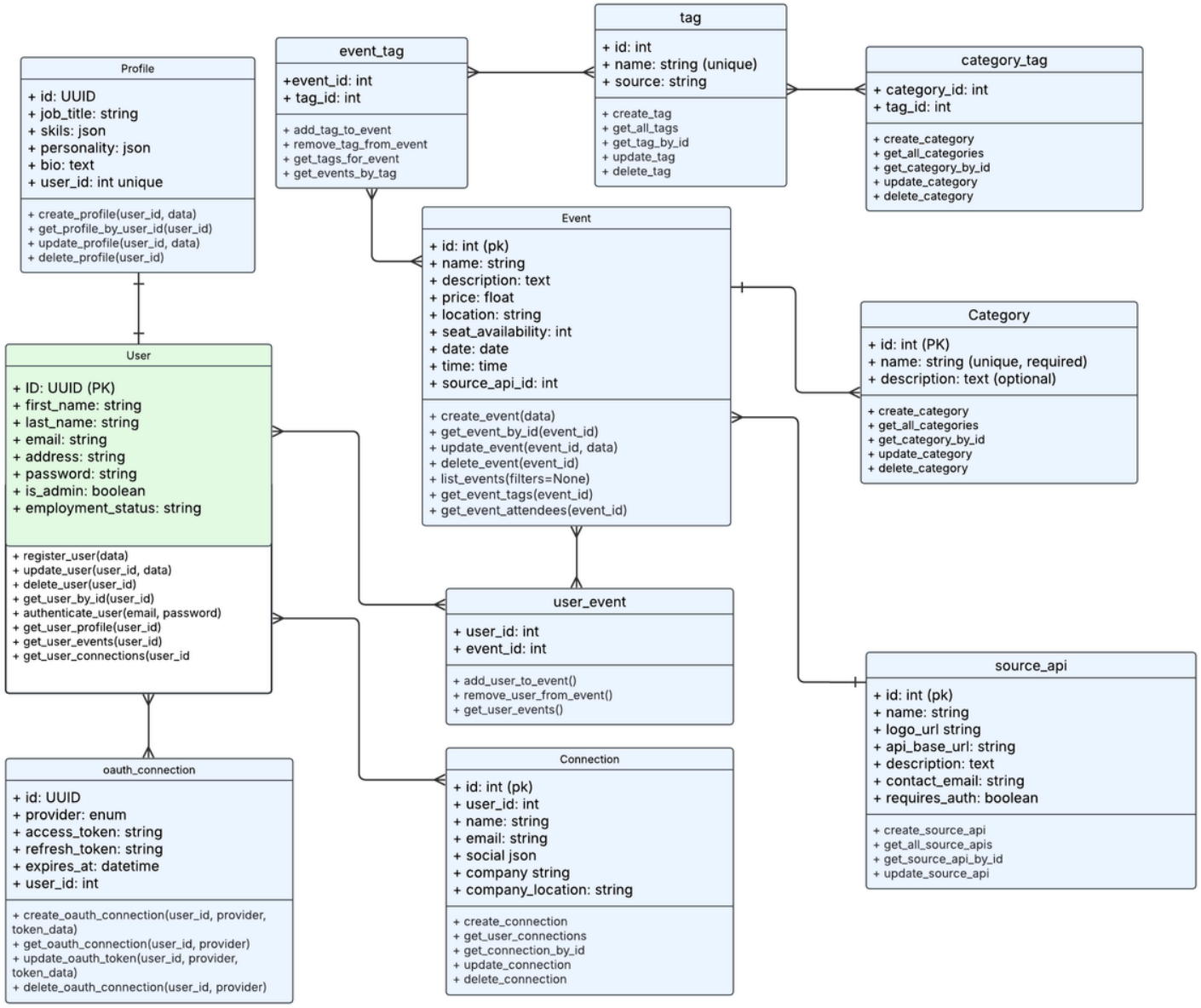
3. Persistence Layer (Database): This layer contains the core application logic and model representations of system entities, including:

- Core entities: Users, Profiles, Events, Tags, Categories, Connections
- Association tables: User-Event, Event-Tag, Category-Tag



Class Diagram

This class diagram provides a structured, scalable foundation for managing users, events, connections, and categorisation within TechMeet. It supports personalised discovery, external API integrations, and social interaction for the tech community.



Sequence Diagrams

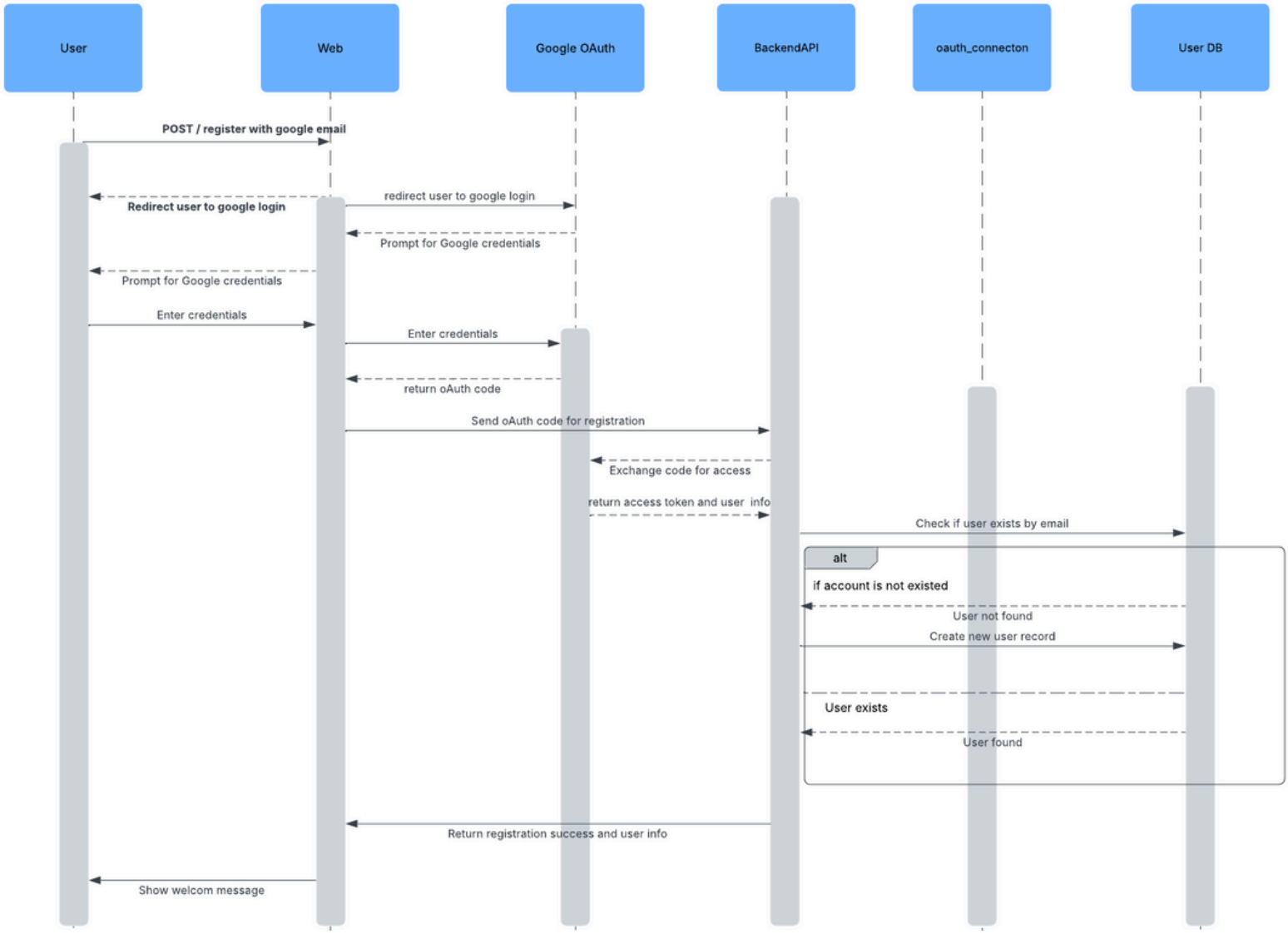
Sequence diagrams show the interaction between each of the application layers during the execution of core functionality.

A sequence diagram represents a timeline that flows from the top to the bottom. At the top of the diagram, the objects that are involved in the process are placed horizontally in the order they are called. An object's "lifeline" spans the height of the diagram.

Activation bars are placed on an object's lifeline to indicate when the object is active. Interactions between

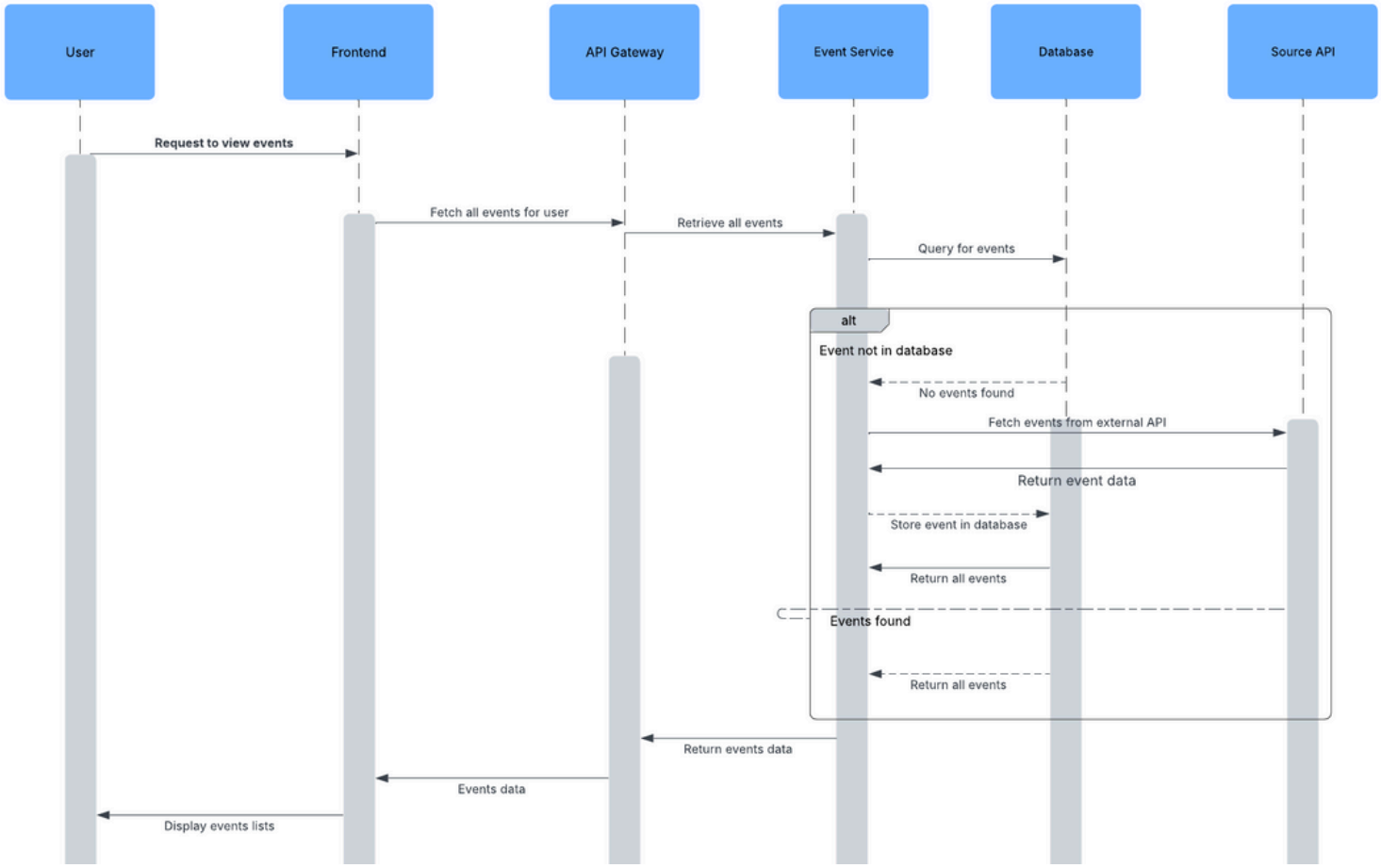
objects occur when one object sends a message to another, which is denoted by a horizontal arrow between the lifeline of two objects.

User Registrations



The diagram illustrates the OAuth registration flow for a new user, beginning with a redirect to Google for authentication. After the user grants permission, an authorisation code is returned, which is then exchanged for an access token. This token is used to retrieve the user's details from Google, which are subsequently stored in the application's database to complete the registration process.

Fetch events



In the Techmeet sequence diagram for fetching events, the process begins when a user sends a request to view events through the client-side interface. This request is forwarded by the frontend to the API Gateway, which then relays it to the Event Service. The Event Service queries the database for available events; if events are found, they are returned directly back through the API Gateway to the frontend, where they are displayed to the user. However, if no events are found in the database, the Event Service fetches data from the external Source API, stores the retrieved events in the database, and then re-queries the database to return the updated list of events to the user interface.