PlusOne

Deliverable 2

Mentor: Ali Darejeh

Team: random_group

Stephen Bletsas (z5257110)
Faris Jalal (z5262405)
Vishnu Birudavolu (z5280782)
Rushil Pardasani (z5266512)
Thiagaut Valcke (z5282625)

Table of Contents

Introduction

In this report, we refine and formalise the software architecture of our mobile application and outline in detail how the different APIs are used in both the frontend and the backend. Some of the user stories were refined to fit better into the design of our application. Also, each user story is accompanied by a sequence diagram, presenting the flow of movements. Overall, as a team, we have progressed in ensuring that the application is technically viable and has a solid product-market fit.

Software Architecture

Diagram 1.1 represents the software architecture of the PlusOne mobile application which will be available on iOS and Android platforms.
It is split into two distinct sections, the frontend and the backend consisting of the selected mobile application stack, external data sources, programming platforms, languages and SDK's that will be implemented in the application.
A brief description of the frontend, backend and the advantages and disadvantages of their respective components can be found below the diagram.

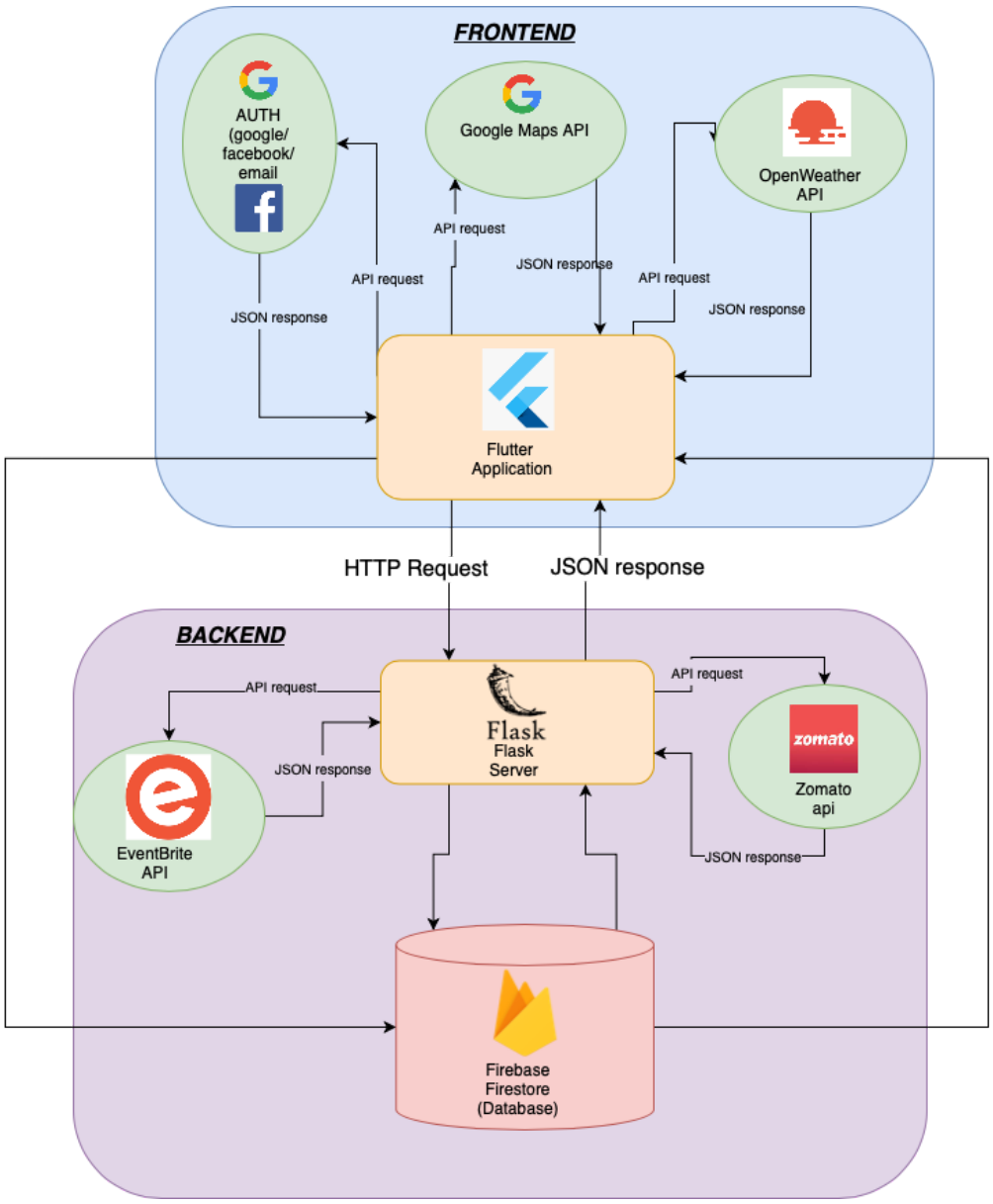


diagram 1.1

Frontend

Flutter:

It is an open-source SDK, developed by Google and written in Dart, C, and C++. We chose to use Flutter to develop our application as it allows us to create apps for Android and iOS devices with just one codebase written in Dart. Furthermore, flutter seamlessly incorporates many material design features making it easier for developers to create beautiful apps without spending unnecessary amounts of time on UI and styling. Flutter also incorporates ahead-of-time(AOT) compilation allowing apps to achieve higher performance. If we expand PlusOne to a web platform, the latest flutter update allows us to use the same codebase(with minimal changes) to deploy the application to the web too. With a vast developer community, excellent support, and regular updates, Flutter is the ideal platform to develop the app.

Facebook Login:

Signing up and logging in with Facebook has been included as it has a large user base and allows new users to create a new PlusOne account quickly and easily.

Google Login:

The option where a user can signup and login into the PlusOne app using their google account has been included as Google is very popular and has an extensive user base. It is also very secure as it uses the OAuth 2.0 API and integrates seamlessly with Flutter.

Google Maps API:

The Google Maps API is used to provide the user with location data like the geographical locations of particular events or venues and travel-related data like optimal routes from venue to venue. It also has integrated ride-share application support (Uber, Didi, Lime, etc) which allows users to travel between event venues with ease.
The choice to use the Google Maps API was obvious as it is an industry-standard in providing geographical location data with 99% coverage of the world, 25 million updates, and approximately 1 billion monthly active users.

OpenWeather API:

We will be using this API to obtain data about the weather at particular locations. We chose to use the OpenWeather API as it provides current weather data, forecasts, location-based weather data and is widely used in web and mobile applications to provide weather data.

Backend

Firebase Cloud Firestore:

Firestore is a fast NoSQL database designed for scaling with the powerful infrastructure of Google Cloud Platform offering automatic horizontal scaling (in and out) as per the application's load.

It offers strong reliability as it distributes data across multiple data centers in distinct regions.

It offers built-in offline support for mobile (iOS, Android) and uses local cache to serve and store data. Hence, an app built using Cloud Firestore remains responsive regardless of network latency or internet connectivity. The app can read, write, listen to, and query data even when the device is offline. If changes are made in the offline state, the same will be synced to the cloud when the client comes back online.

It requires less data flattening or denormalization as it is more structured and utilizes nesting of objects. Cloud Firestore supports indexed queries, thus it enables you to combine filtering and sorting on a property in a single query.

It offers atomic (all or nothing executes) write and transaction operations which also means that the transactions will keep repeating until completion, automatically.

All these advantages make Firestore an ideal database solution for a mobile application like PlusOne.

Flask:

Flask is a lightweight WSGI web development framework that is used with Python to build backend systems. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It is one of the most popular Python web application frameworks, and hence has a lot of readily available resources and documentation. It is lightweight, versatile, and can be used easily along with Firebase Cloud Firestore, our database solution. The backend Python server will need to handle API calls as well as process and store information in the database and hence Flask is the most suitable framework available at our disposal.

Zomato API:

The Zomato API allows developers to access a database of the newest and most exhaustive restaurant content to power applications with. It covers 1.5 million restaurants across 10,000 cities globally.The API allows us to list restaurants by parameters such as name, cuisine, location, ratings along with detailed information such as location coordinates, reviews, discounts, photos, menus. This allows PlusOne to have a wide variety of dinner options at the best rated restaurants across town. It also returns responses in JSON format,

which proves beneficial to us as it ties in with the NoSQL database which stores data in JSON-like documents and collections.

Eventbrite API:

Eventbrite is a popular event management and ticketing platform that allows users to find forthcoming local occasions and receive personalized recommendations. It is used by 700,000 event organizers to host 2 Million events per year. The API allows us to list events, ticket classes, discounts, attendees, venues along with detailed information such as location coordinates, reviews, organizers, formats, pricing, display settings, and media. This allows PlusOne to have a wide variety of event options across town. It too, like the Zomato API, returns responses in JSON format, which proves beneficial to us as it ties in with the NoSQL database which stores data in JSON-like documents and collections.

Developers can sign up for a referral program to earn a revenue share too, which is a possible form of revenue for PlusOne in the future.

Overall, these architectural choices ensure that our application is responsive and that all components can seamlessly integrate with each other.

User Stories

User Story 1	
Feature	Register a User
Story	As a first-time user, So that I can access and use the application, I want to be able to register for a profile account.
Case	Given that I am on the homepage, When I click on the 'Sign Up/Log In' button, Then I am redirected to a signup form where I can create an account via: · Connecting through my Google account · Connecting through Facebook · Creating a new account
Priority	1st Priority

User Story 2	
Feature	Creating a plan path and adding constraints
Story	As a user, So that I can organise an outing with friends,

	I want to be able to create a plan path of what is available given certain filters and updated covid restrictions.
Case	Given that I have signed up as a user, When I click on the 'create plan' button, Then I am redirected to the 'planner filter' page, When I fill in various details: · Event name · Start time · Budget · Type of event (date, drinks, chill, etc.) · Number of people, etc. Then I should see an auto-generated plan path with the ability to customize this recommendation, And be able to finalise the plan path.
Priority	1st Priority

User Story 3	
Feature	Editing the plan path after it has been created
Story	As a user, So that I can tweak a few changes in the plan path provided by the application, I want to be able to edit and update the plan path generated.

Case	Given that I am on the specific plan paths page, When I click on the 'Edit path' button, Then I can edit segments of the plan to suit my preferences.
Priority	1st Priority

User Story 4	
Feature	Displaying path details
Story	As a user, So that if I want an overview of a path, I want to be able to view details about the path and the events/venues it contains.
Case	Given that I am on the homepage or events page, When I click on a specific plan path, Then I can view details such as the weather, map and path plan, And When I click on an event or venue, Then I can view details about the venue/event also provided with a map.
Priority	2nd Priority

User Story 5	
Feature	Deleting a plan after it has been created

Story	As a user, So that in case of unforeseen circumstances that the event cannot be carried, I want to be able to delete the plan.
Case	Given that I am on the 'plan' page, When I click on the '-' icon, Then a message appears with a message to confirm the deletion of the event, When I click "Yes, I am sure", Then The plan is deleted, And it notifies everyone that the plan has been deleted.
Priority	2nd Priority

User Story 6	
Feature	Automated booking
Story	As a user, So that I can reserve a placement at a venue, I want to be able to book through the app without direct interaction with the event page/venue.
Case	Given that I am on the specific plan path page, When I click on the 'book' button, Then a selection of events that can be booked appears,

	When events are selected and confirmed, Then a booking for the selected venue/events is processed.
Priority	2nd Priority

User Story 7	
Feature	View event history
Story	As a user, So that I can compare details of plans and to look back on plans or maybe rebook a plan, I want to be able to view past events by order of date (recent to oldest).
Case	Given that I am on the homepage, When I click on the 'History' button, Then I am redirected to the 'History' page, And I am able to see the past history of events I attended with details.
Priority	3rd Priority

User Story 8	
Feature	Chat system

Story	As a user, So that I can organise and coordinate events without needing to access multiple social medias, I want to be able to chat with them within the app.
Case	Given that I have signed up as a user, And my friend has also signed up, When I click on the 'Chat' button, Then I am redirected to the 'Chat' page, When I click on the 'Create Chat' button, Then I am able to add a user/s via phone number, account name or email. And a new chat is created And messages can be sent
Priority	2nd Priority

User Story 9	
Feature	Notify group members
Story	As a group member, So that I can be informed of event time, location changes or any updates within the group, I want to be notified of updates through notifications on my phone.

Case	Given that I am a member of the group, When any changes or updates occur within a group or event, Then I receive a notification of the update so I can be informed.
Priority	2nd Priority

User Story 10	
Feature	Updating profile details/preferences
Story	As a user, So that friends can see what my interests and preferences are through my account, I want to be able to add/update my personal preferences within my profile.
Case	Given that I am on the 'Home' page, When I click on the 'profile' button/display picture, Then I am redirected to the 'My profile' page, When I click on the 'Preferences' button Then I am redirected to the 'preferences' page. And I am able to update my user preferences
Priority	2nd priority

User Story 11	
Feature	Notify plan path
Story	As a user, So that I can know when to leave one venue/activity on time to make it to the next one, I want to be sent notifications to my phone to alert me that I need to be at the next venue soon
Case	Given that I have already created a plan path, And the event is currently underway, When one event in the plan path is nearly finished, Then a notification will be sent to the phone to alert the user. (This will be the case for both normal and mystery plans)
Priority	2nd Priority

User Story 12	
Feature	Accessing public transport prices and timetable
Story	As a user, So that I can travel hassle-free to the event, I want to be notified with public transport prices and timetables when it is time to leave for the event.
Case	Given that I have already created a plan path,

	When it is time to leave for the event, Then a notification will be sent to the phone to alert the user with both cost and time estimates for public transport. (This will be the case for both normal and mystery plans)
Priority	2nd Priority

User Story 13	
Feature	Getting Uber prices and timetable
Story	As a user, So that I can travel hassle-free to the event, I want to be notified with Uber prices and timetables when it is time to leave for the event.
Case	Given that I have already created a plan path, When it is time to leave for the event, Then a notification will be sent to the phone to alert the user with both cost and time estimates for Uber. (This will be the case for both normal and mystery plans). When I click on the type of Uber I want, Then I get redirected to the Uber app.
Priority	2nd Priority

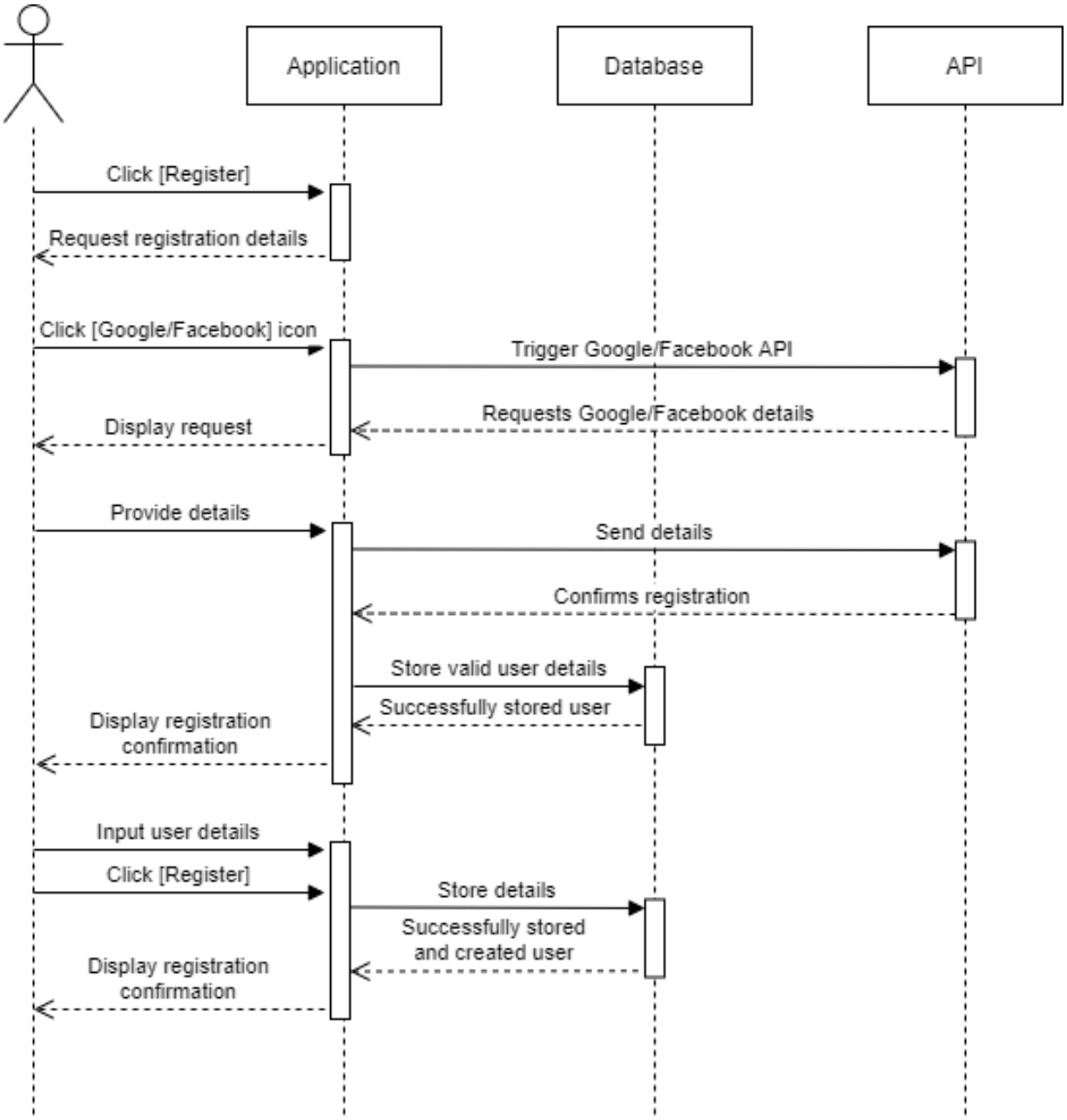
Sequence Diagrams

Application component: Refers to both frontend and backend of software architecture.

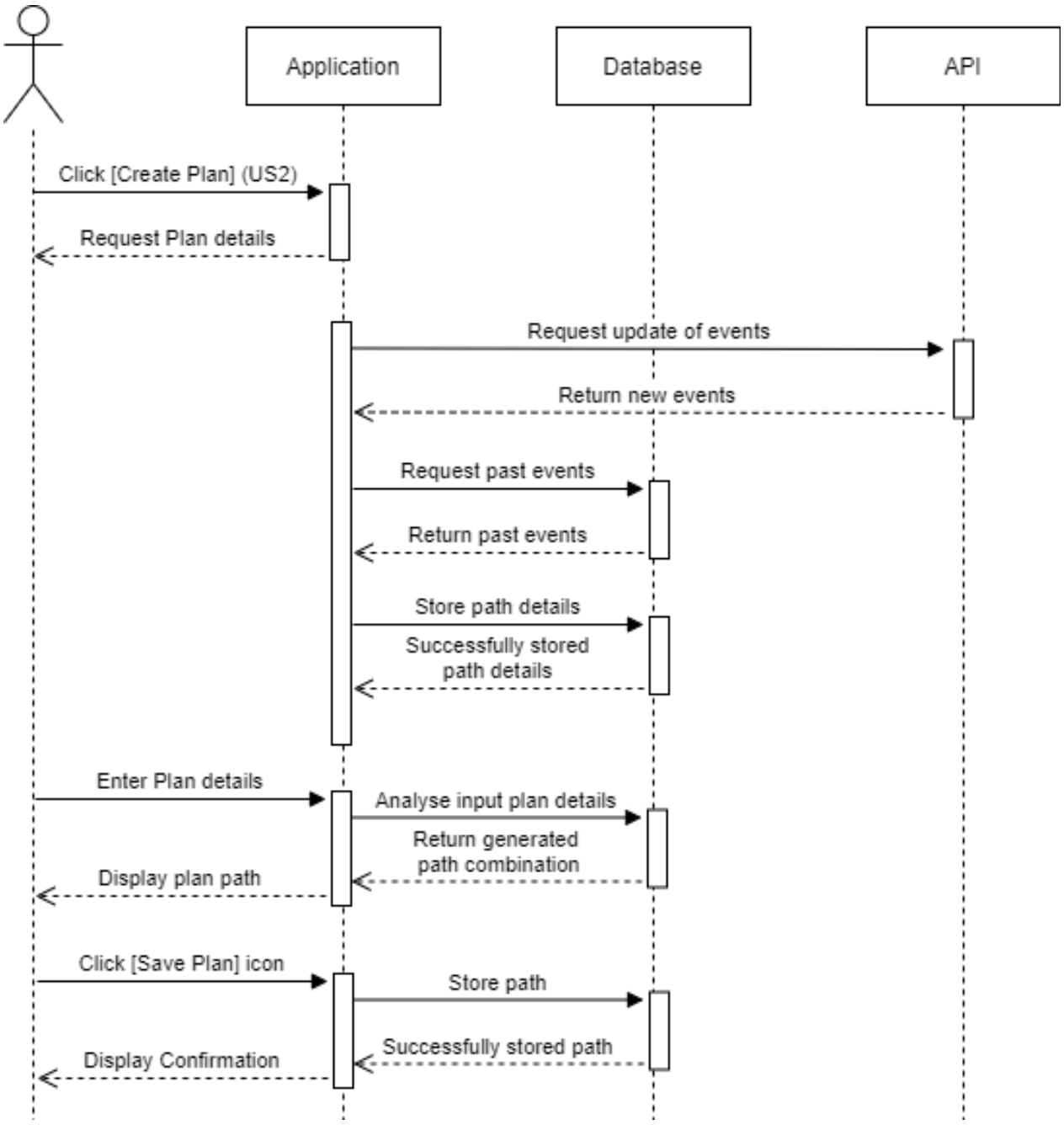
Database component: Refers to the Firebase Cloud Firestore database.

API component: Refers to the various API's implemented in the software architecture.

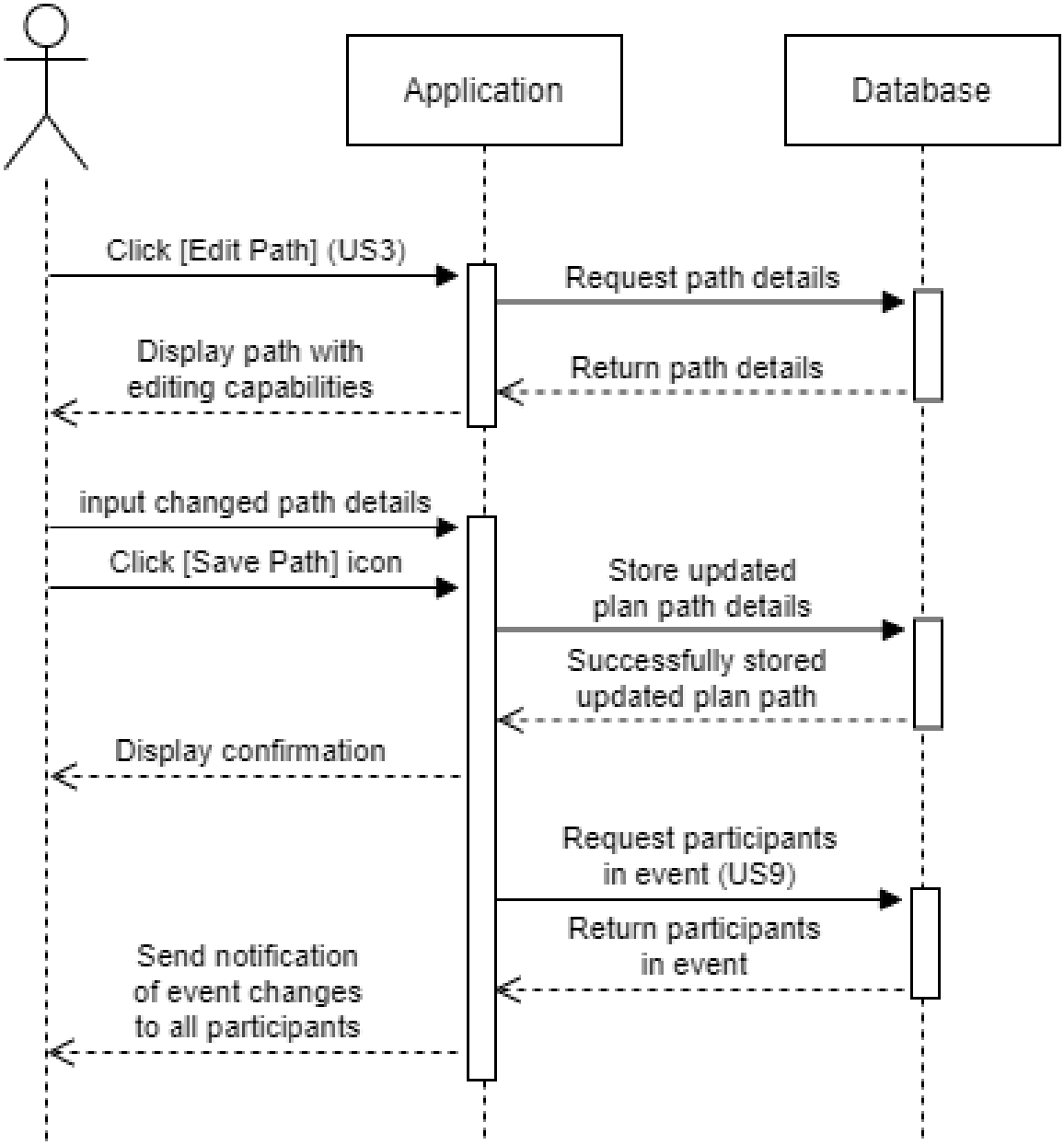
Epic story 1: As a first-time user, so that I can access and use the application, I want to be able to register for a profile account. (US1)



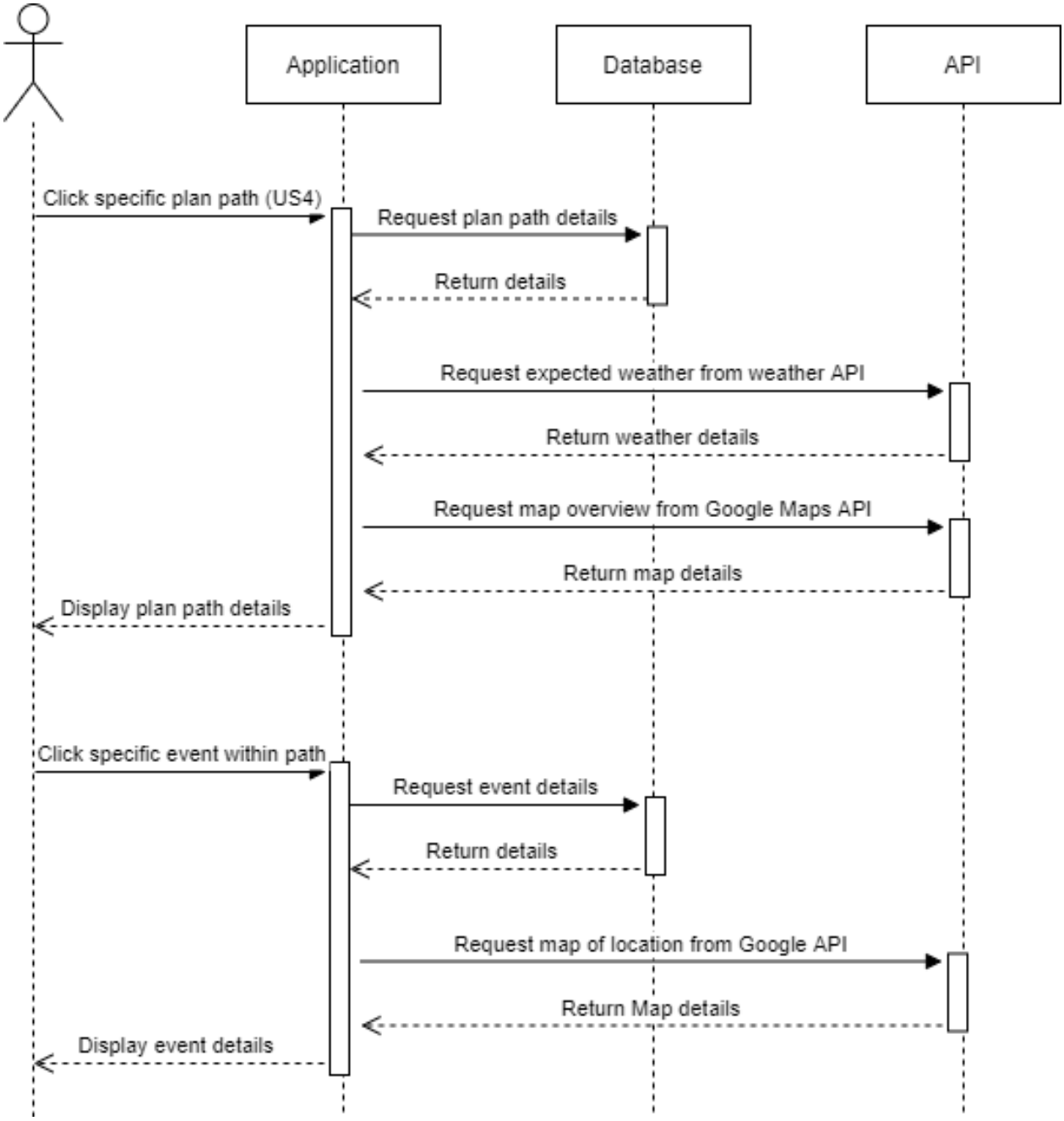
Epic story 2: As a user, I want to be able to create a plan, view plan details, book venues and also have the ability to edit and delete these plans notifying all event participants of any updates and changes. (US 2,3,4,5,6,9)



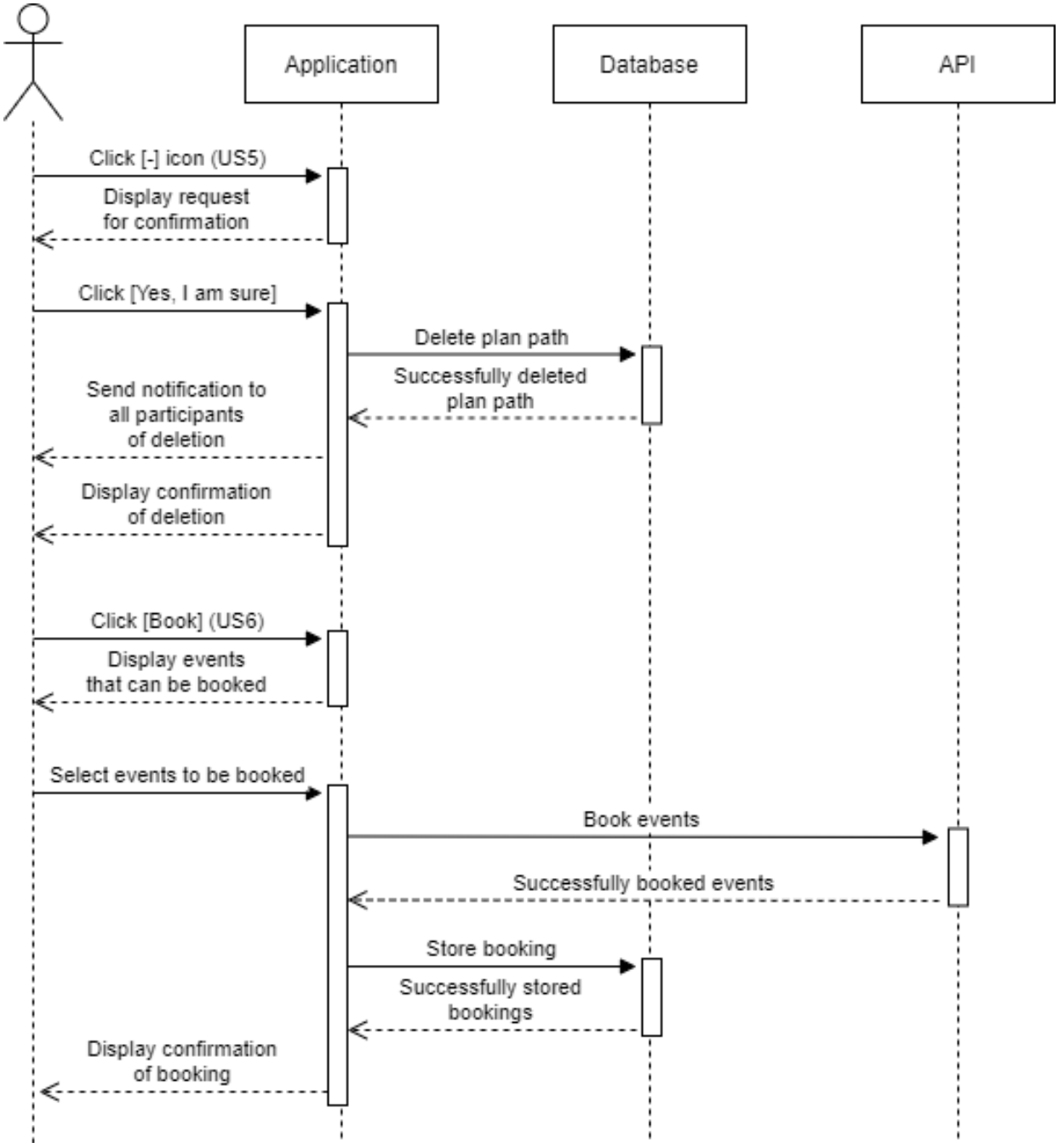
Epic story 2 (Cont.)



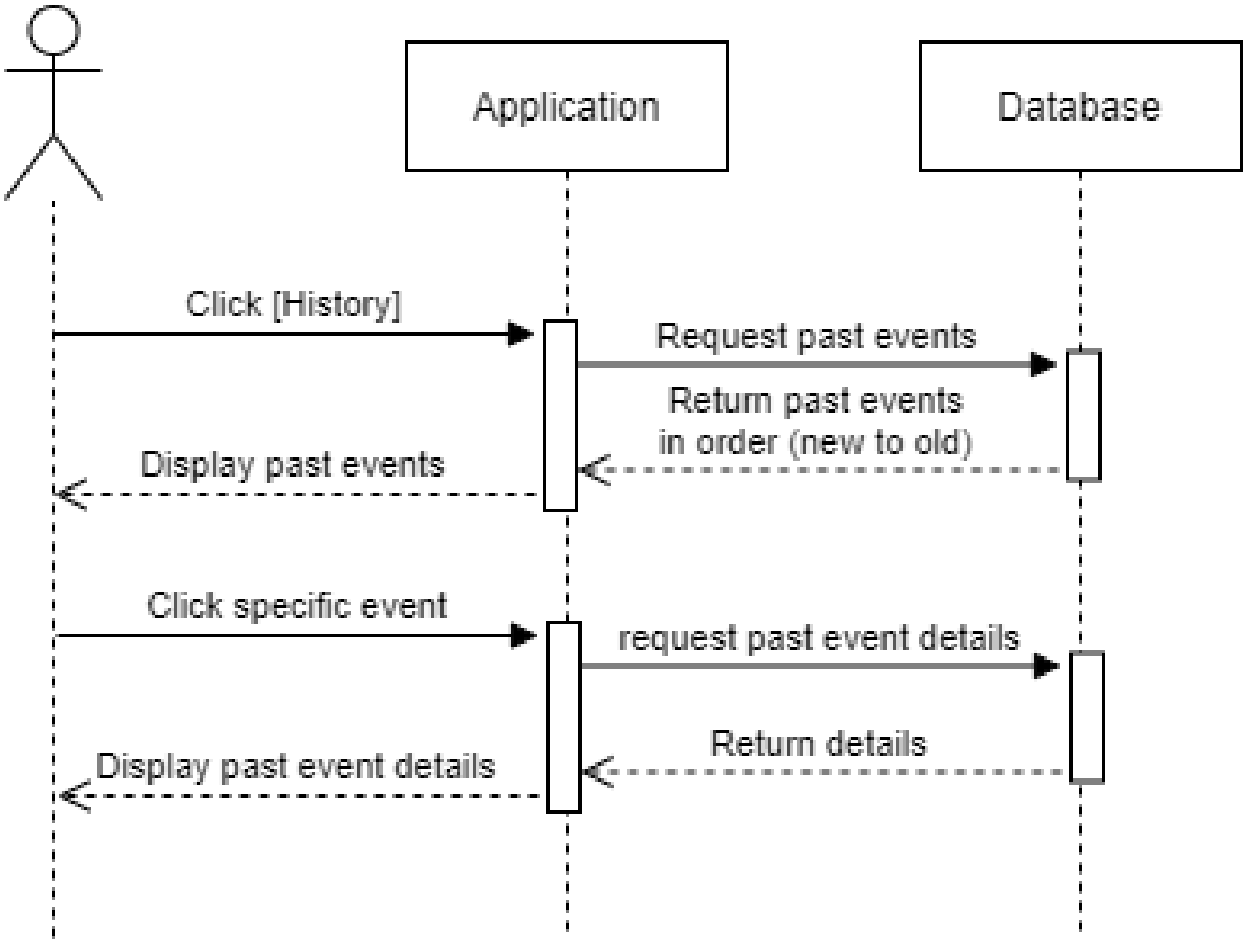
Epic Story 2 (Cont.)



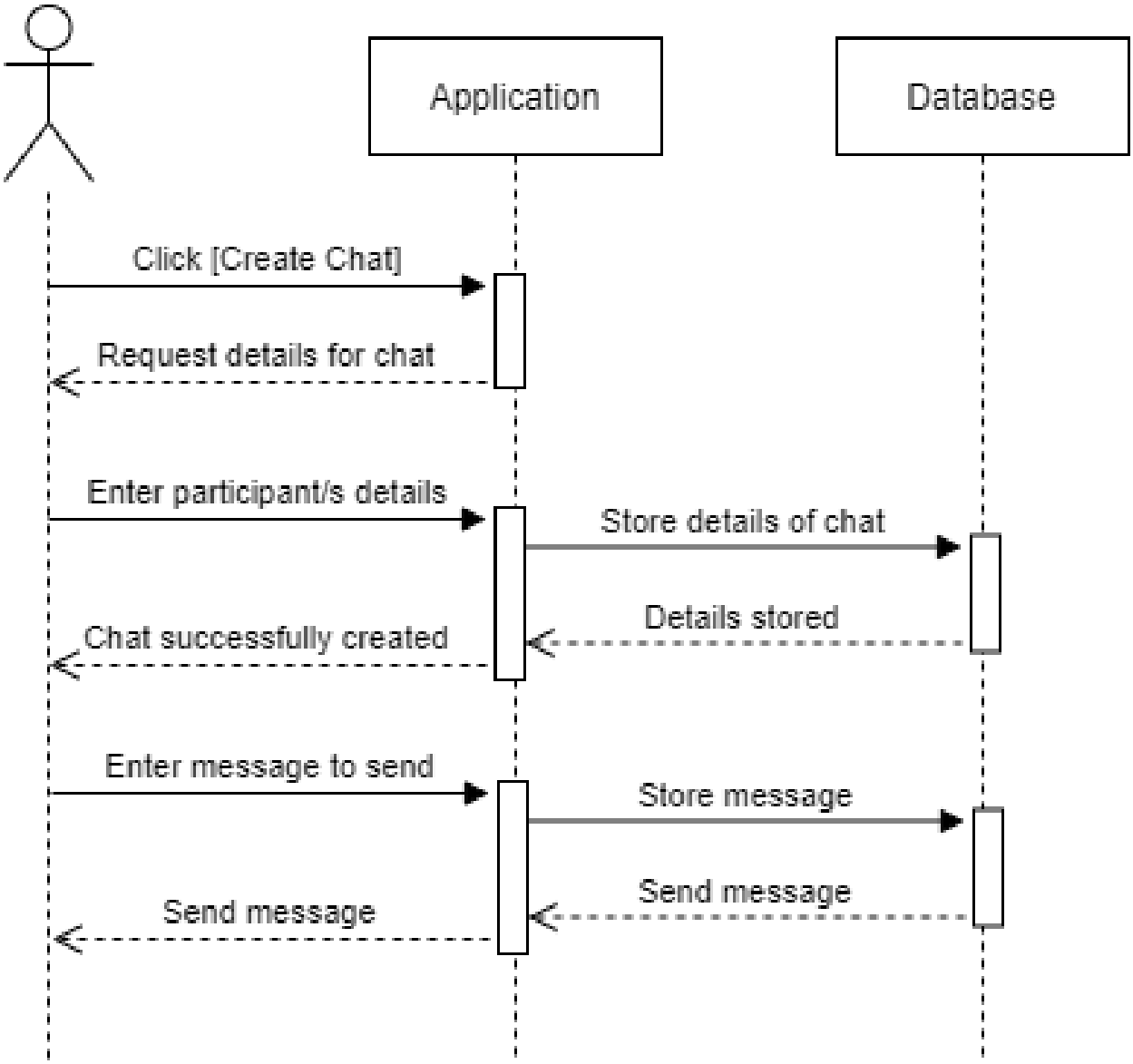
Epic story 2 (cont.)



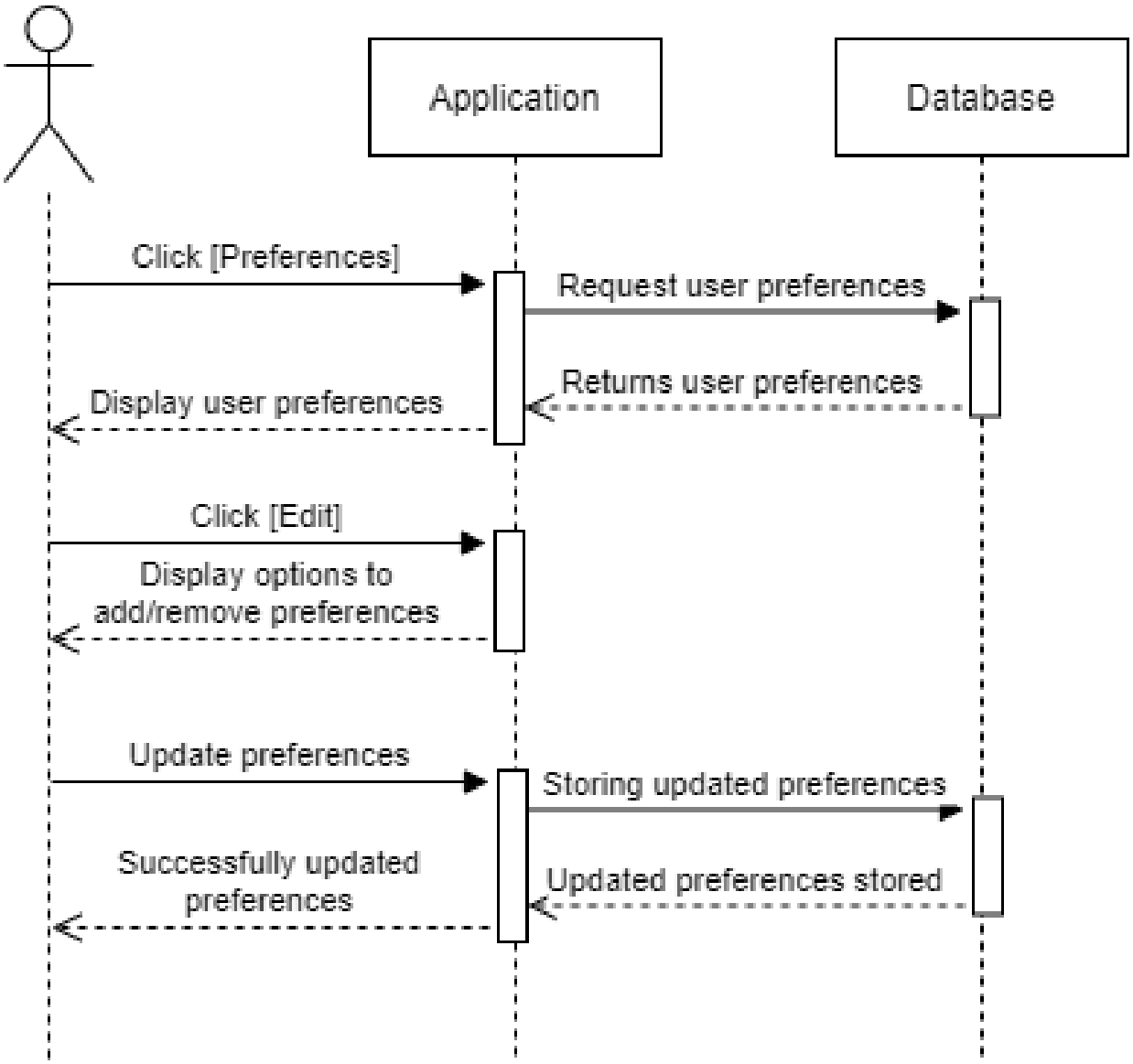
Epic Story 3: As a user, I want to be able to view past events by order of date so that I can compare details of plans and to look back on plans. (US7)



Epic Story 4: As a user, I want to be able to chat with other users within the app so that I can organise and coordinate events without needing to access multiple social media. (US8)



Epic Story 5: As a user, I want to be able to add/update my personal preferences within my profile so that friends can see what my interests and preferences are. (US10)



Epic Story 6: As a user, I want to be sent notifications to my phone to alert me that I need to be at the next event, also being notified of public transport times and Uber fares (US 11, 12, 13)