Best Bike Paths - Hu Peng, Zhang Zihao

POLITECNICO
MILANO 1863

Requirement Analysis and Specification Document

Deliverable:	RASD
Title:	Requirement Analysis and Verification Document
Authors:	Hu Peng, Zhang Zihao
Version:	1.0
Date:	19-December-2025
Download page:	https://github.com/euorrl/HuZhang
Copyright:	Copyright © 2025, Hu Peng, Zhang Zihao – All rights reserved

Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose

The increasing attention to sustainable mobility and healthy lifestyles has made cycling an important means of transportation in urban environments. However, cyclists often face difficulties in identifying safe, efficient, and well-maintained bike paths, especially in unfamiliar areas. Information about bike paths is frequently scattered, outdated, or unavailable, making it hard for users to choose routes that best fit their needs and preferences.

The purpose of Best Bike Paths (BBP) is to provide a system that supports cyclists in discovering, evaluating, and selecting bike-friendly paths by collecting and organizing information about cycling routes. BBP allows users to record their trips, share information about bike paths, and access aggregated data provided by the community and by automatic data collection mechanisms.

Through BBP, users should be able to gain a clear understanding of the available bike paths between an origin and a destination, compare alternative routes, and choose the most suitable one based on path conditions and overall effectiveness. By fostering information sharing and transparency among cyclists, BBP aims to improve the overall cycling experience and promote safer and more informed routing decisions.

1.1.1 Goals

G1: Cyclists are supported in recording their personal bike trips, allowing them to keep track of their cycling activities and access basic statistics related to their rides.
G2: Users are enabled to manually insert information about bike paths, including path conditions and relevant observations, and to decide whether such information can be made publishable to the community.
G3: The system supports the visualization of possible bike paths between a user-specified origin and destination, helping cyclists explore available routes and select suitable paths.

1.2 Scope

This section defines the scope of the Best Bike Paths (BBP) system by outlining the boundaries of the functionalities provided by the application. The system focuses on supporting cyclists in collecting, sharing, and exploring information related to bike paths. In particular, BBP supports:

- the recording of personal bike trips and the visualization of basic trip information;
- the manual insertion of publishable information about bike paths, such as path conditions and relevant observations;
- the visualization of possible bike paths between a user-defined origin and destination.

The system does not aim to control real-world cycling activities, road infrastructure, or traffic conditions, but rather to collect and organize information provided by users and make it available for consultation.

1.2.1 World Phenomena

WP1: Cyclists travel along bike paths in urban or suburban environments.
WP2: Cyclists choose routes between an origin and a destination based on personal preferences such as safety, distance, and comfort.
WP3: Cyclists experience different path conditions during their rides, such as good pavement, obstacles, potholes, or poor maintenance.

WP4: Cyclists generate movement data while riding, including position, speed, and changes in motion.
WP5: Cyclists form personal evaluations of bike paths based on their riding experience.

1.2.2 Shared Phenomena

SP1: Cyclists record their personal bike trips using the BBP system.
SP2: The system acquires trip-related data from users, such as GPS traces and basic movement information.
SP3: Cyclists manually insert information about bike paths, including path conditions and observations.
SP4: Cyclists choose whether the information they provide can be made publishable to the community.
SP5: Cyclists request the visualization of possible bike paths between a specified origin and destination.
SP6: The system visualizes alternative bike paths and presents them to the user.
SP7: Cyclists consult information provided by the community to support their route selection decisions.

1.3 Definitions, Acronyms, Abbreviations

Definitions

- **Bike Path**: A designated route or road segment intended to be used by cyclists, where bike traffic is supported or prioritized. A bike path may vary in terms of surface quality, safety, and maintenance conditions.
- **Trip**: A cycling activity performed by a cyclist, consisting of a sequence of movements between an origin and a destination, recorded over a period of time.
- **Path Condition**: Information describing the state of a bike path, such as pavement quality, presence of obstacles, potholes, or maintenance requirements.
- **Personal Trip Recording**: The process through which a cyclist records their own bike trip using the BBP system, allowing the collection of basic trip-related data and statistics.
- **Publishable Information**: Information about bike paths provided by cyclists that can be made visible to other users of the system, subject to the data owner's decision.
- **Community Information**: Aggregated information about bike paths collected from multiple cyclists and made available by the BBP system for consultation.
- **Route Visualization**: The presentation by the system of one or more possible bike paths between a user-defined origin and destination on a map.

Acronyms

- **BBP**: Best Bike Paths.
- **GPS**: Global Positioning System.
- **UML**: Unified Modeling Language.
- **UI**: User Interface.
- **API**: Application Programming Interface.

Abbreviations

- **G**: Goal.
- **D**: Domain Assumption.
- **R**: Functional Requirement.
- **P**: Property to Verify.
- **WP**: World Phenomena.
- **SP**: Shared Phenomena.

1.4 Revision History

Version	Date	Description
1.0	19-December-2025	Initial version of the document

1.5 Reference Documents

The assignment for this document and all the information included herein refer to the following documentation:

- The specification of the 2025–2026 Requirement Engineering and Design Project for the *Software Engineering II* course, Computer Science and Engineering, Politecnico di Milano.
- The slides and teaching material available on the WeBeep platform for the *Software Engineering II* course.
- *Software Engineering II* course, Computer Science and Engineering, Requirement Analysis and Specification Document, Students & Companies, 2024–2025.

1.6 Document Structure

This document is structured as follows:

- **Introduction**: presents an overview of the Best Bike Paths (BBP) system, including the purpose of the document, the goals of the project, its scope, key definitions, and reference documents.
- **Overall Description**: provides a general description of the BBP system, outlining its product perspective, main functionalities, user characteristics, assumptions, dependencies, and constraints.
- **Specific Requirements**: details the functional and non-functional requirements of the system, including external interface requirements, functional requirements, performance requirements, design constraints, and software system attributes.
- **Formal Analysis Using Alloy**: presents a formal model of selected aspects of the BBP domain, highlighting key assumptions and verifying the consistency of the model through Alloy analysis.
- **Effort Spent**: reports the amount of time spent by each group member in the preparation of this document.
- **References**: lists the documents and tools that were referenced or used during the development of this document.
- **Use Of AI Tool**: Described the usage of AI tools.

2 Overall Description

2.1 Product perspective

The Best Bike Paths (BBP) system is a software application aimed at supporting cyclists in collecting and consulting information about bike paths. BBP is designed as an information-support system that helps users explore available routes and share their personal experience, without providing real-time navigation or guaranteeing optimal routing decisions.

BBP interacts with the real world through cyclists and their mobile devices. During bike trips, users may record basic movement and location data, which can be stored and later consulted through the system. In addition, users can explicitly provide information about bike paths based on their personal experience, such as qualitative evaluations and observations.

The system distinguishes between registered users and unregistered users. Registered users can actively interact with the system by recording trips and inserting information about bike paths, while unregistered users can only consult information that has been made available to the community. Information provided by users can be marked as publishable at the discretion of the data owner.

BBP may rely on external services, such as mapping services for route visualization and weather services for enriching trip information when available. The system does not manage physical road infrastructure, traffic conditions, or real-time navigation, but focuses on organizing and presenting information to support users in their route selection.

2.1.1 Scenarios

Scenario 1: Recording a Personal Bike Trip (Registered User)

A registered cyclist uses the BBP application while riding in an urban area. Before starting the trip, the user opens the application and enables the trip recording functionality. During the ride, the system collects basic movement and location data generated by the user's mobile device. Once the ride is completed, the user stops the recording and the trip is stored in the user's personal history, where it can later be consulted together with basic trip statistics.

Scenario 2: Manual Insertion of Bike Path Information (Registered User)

A registered cyclist encounters a bike path with poor maintenance conditions, such as potholes or obstacles. After completing the ride, the user accesses BBP and manually inserts information about the affected bike path, specifying the relevant streets and providing qualitative observations. The user decides to mark this information as publishable, making it available to the community for consultation by other cyclists.

Scenario 3: Exploring Bike Paths Between an Origin and a Destination (All Users)

A cyclist plans to reach a destination in an unfamiliar area. The user opens BBP and specifies an origin and a destination. The system visualizes one or more possible bike paths on a map, based on the information currently available. The cyclist compares the proposed paths and selects the one that best fits personal preferences and needs.

Scenario 4: Consulting Community Information (All Users)

A cyclist wants to obtain information about bike paths in a given area. The user accesses BBP without logging in and specifies an origin and a destination. The system visualizes available bike paths and shows only the information that has been made public by registered users. The cyclist uses the community-provided information to support route selection, without being able to record trips or insert new path information.

2.1.2 Domain Class Diagrams

The domain class diagram of the Best Bike Paths (BBP) system is shown in Figure 1. The diagram models the main conceptual entities of the system and their relationships, focusing on the persistent information managed by BBP, while abstracting away user interface and control aspects.

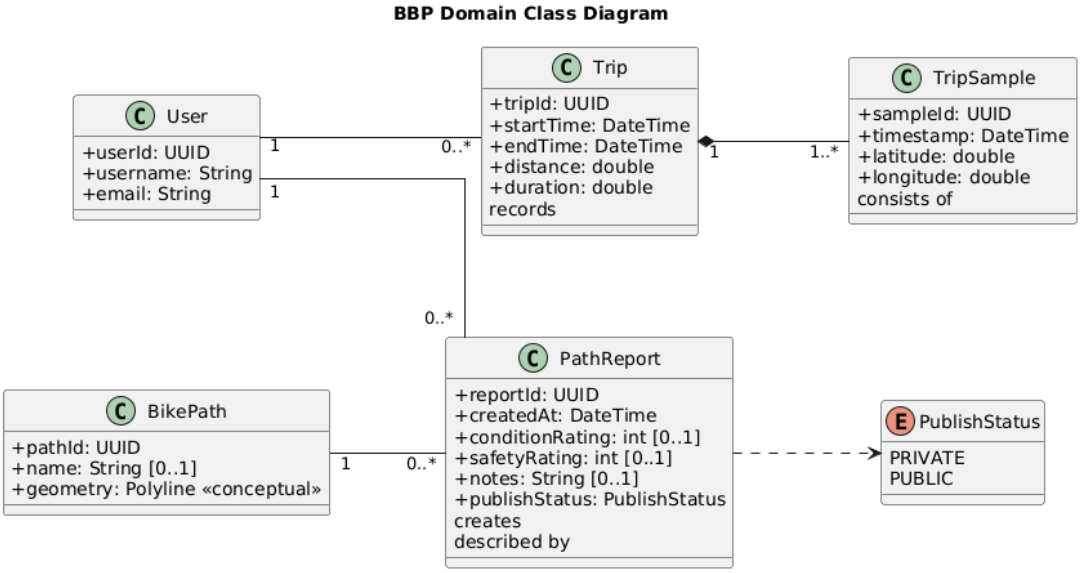

Figure 1: BBP Domain Class Diagram

At the core of the domain model is the *User* class, which represents a registered cyclist. A user is identified by a unique identifier and basic account information. Registered users can record personal bike trips and create reports describing bike paths.

Bike trips are modeled through the *Trip* class, which represents a single recorded cycling activity. Each trip is associated with exactly one user and stores basic temporal and quantitative information, such as start and end time, total distance, and duration. Each trip aggregates one or more *TripSample* instances, which represent individual location samples collected during the ride. This composition relationship reflects the fact that trip samples cannot exist independently of a trip and are removed when the corresponding trip is deleted.

Bike paths are represented by the *BikePath* class, which models a conceptual path that can be evaluated and consulted by users. A bike path is identified by a unique identifier and may optionally have a name, together with a geometric representation used for visualization purposes. The domain model does not assume real-time navigation capabilities, but treats bike paths as informational entities.

User-provided information about bike paths is captured by the *PathReport* class. A path report represents qualitative information created by a registered user to describe a specific bike path. Each path report is created by exactly one user and refers to exactly one bike path, while a bike path may be described by multiple reports originating from different users. Path reports may include optional ratings, textual notes, and a creation timestamp.

The visibility of a path report is modeled through the *PublishStatus* enumeration, which distinguishes between private and public information. This allows users to decide whether a report should remain part of their personal records or be made available to the community. Only path reports marked as public are accessible to unregistered users during route exploration and consultation activities.

Overall, the domain class diagram captures the essential concepts required to support trip recording, manual insertion of bike path information, and consultation of community data. The model emphasizes clear ownership of data, controlled information sharing, and a separation between personal and public information, in accordance with the system requirements.

2.1.3 State Diagrams

BBP User Session State Machine

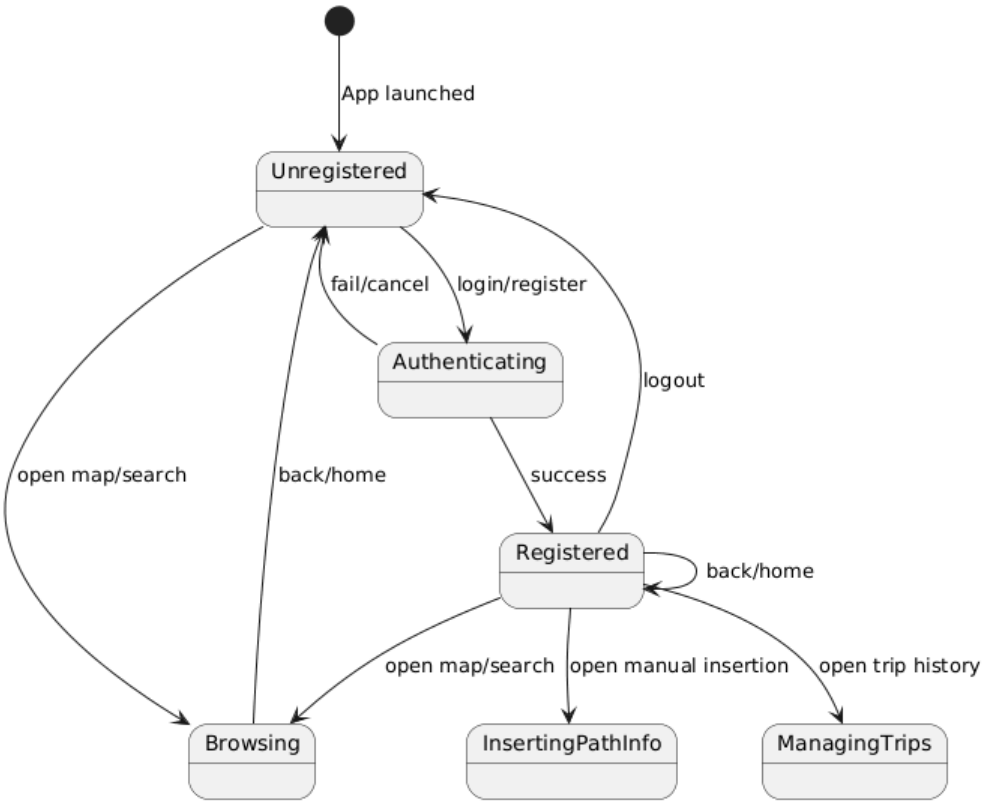

Figure 2: BBP User Session State Machine

Figure 2 illustrates the user session state machine of the BBP system. The diagram models the evolution of a user's session from the application launch to different interaction states, distinguishing between unregistered and registered users.

When the application is launched, the user initially enters the *Unregistered* state, from which basic browsing functionalities such as map exploration and route search are available. Unregistered users may initiate the authentication process by logging in or registering, leading to the *Authenticating* state. A successful authentication transitions the user to the *Registered* state, while failures or cancellations return the user to the unregistered state.

Registered users are granted access to additional functionalities, including the management of personal trip records and the manual insertion of bike path information. The state machine also models the possibility for registered users to log out, returning to the unregistered state. This diagram focuses on the lifecycle of user sessions and the availability of system functionalities based on the user's authentication status, while more detailed functional behaviors (e.g., trip recording) are modeled in dedicated state machines.

Trip Recording State Machine

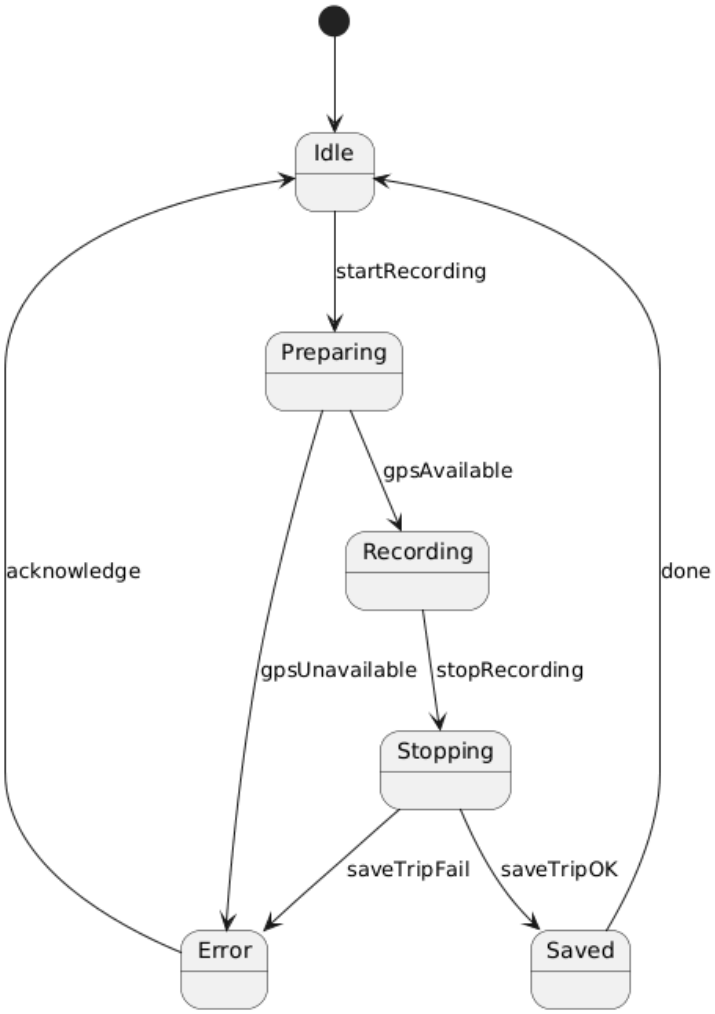

Figure 3: Trip Recording State Machine

As shown in Figure 3, the bike trip recording process in the BBP system is modeled through a dedicated state machine that describes the lifecycle of a single recording session. This state machine focuses on the functional behavior related to the acquisition and persistence of trip data and is modeled independently from user session management aspects.

Initially, the system is in the *Idle* state, indicating that no trip is currently being recorded. When a registered user initiates a recording session, the system transitions to the *Preparing* state, during which preliminary checks are performed, such as verifying the availability of GPS data. If the preparation phase is successful, the system enters the *Recording* state and starts collecting movement and location data generated during the ride. If the required resources are unavailable, the system transitions to an *Error* state.

While in the *Recording* state, the system continuously acquires trip-related data until the user explicitly stops the recording. The system then moves to the *Stopping* state, where the collected data is processed and saved. A successful save operation leads to the *Saved* state, whereas failures during data persistence result in an *Error* state. From both the *Saved* and *Error* states, user acknowledgment causes the system to return to the *Idle* state, thus completing the lifecycle of the trip recording process.

Manual Insertion Of Bike Path Information

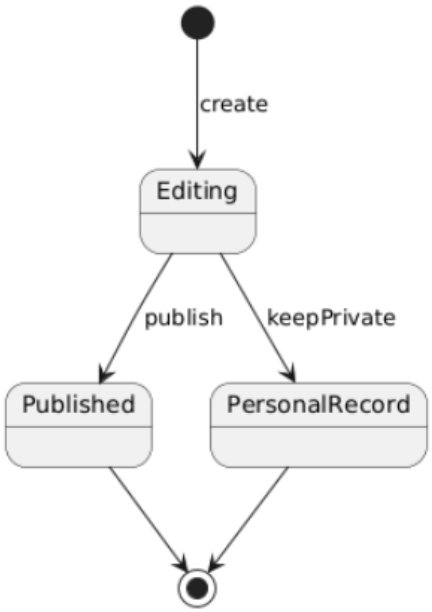

Figure 4: Manual Insertion Of Bike Path Information

Figure 4 illustrates the state machine for the manual insertion of bike path information in the BBP system. After creation, the user enters the *Editing* state, where path-related information is manually provided.

Once the editing activity is completed, the user decides whether to publish the information. If the user chooses to publish it, the information becomes publicly available to the community. Otherwise, if the user decides to keep it private, the information is stored as part of the user's personal cycling records and is not shared with other users.

Route Exploration

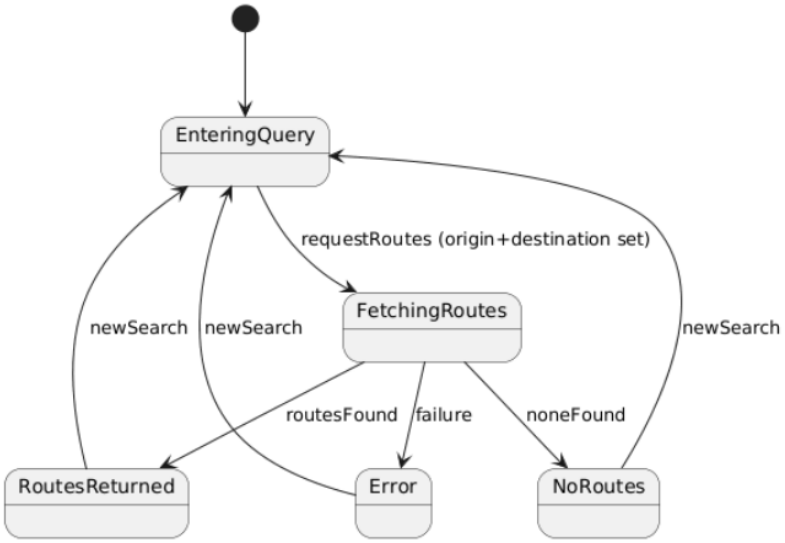

Figure 5: Route Exploration

As shown in Figure 5, the state machine describing the route exploration process in the BBP system. The interaction starts in the *EnteringQuery* state, where the user specifies an origin and a destination for the route search.

Once both endpoints are set, the user can issue a route request, triggering a transition to the *FetchingRoutes* state, in which the system retrieves the available bike paths between the selected locations. The outcome of this operation can be one of three cases: at least one path is found (*RoutesReturned*), no suitable path is available (*NoRoutes*), or an error occurs during the request (*Error*). Regardless of the outcome, the user can start a new search at any time, which returns the system to the *EnteringQuery* state.

2.2 Product functions

At a high level, the BBP system offers the following main functionalities:

- **User registration and authentication**: users can register an account and log in to access functionalities reserved to registered users.
- **Trip recording**: registered users can record their personal bike trips, allowing the system to store basic movement and location data.
- **Trip visualization**: registered users can visualize recorded trips and consult basic trip-related information.
- **Manual insertion of bike path information**: registered users can manually insert information about bike paths, including qualitative evaluations and observations, and decide whether such information should be made publishable.
- **Consultation of community information**: all users can consult information about bike paths that has been made available by other users.
- **Route exploration**: all users can request the visualization of possible bike paths between a specified origin and destination and explore alternative routes provided by the system.

These functionalities define the main services offered by BBP. More detailed functional requirements are specified in Section 3.2.

2.3 User characteristics

BBP is intended to be used by cyclists with heterogeneous backgrounds and levels of technical expertise.

- **Registered users** are cyclists who create an account to access the full set of system functionalities. They are expected to have basic familiarity with mobile applications and map-based interfaces. Registered users may be interested in recording their bike trips and sharing information about bike paths.
- **Unregistered users** access BBP without creating an account. Their interaction with the system is limited to consulting available information and visualizing routes.

The system does not require advanced technical skills and is designed to be usable by a broad range of cyclists.

2.4 Assumptions, dependencies and constraints

Regulatory policies

The BBP system collects and processes personal and location-related data generated by cyclists during bike trips. Such data must be handled in compliance with applicable data protection regulations, including the General Data Protection Regulation (GDPR). Location data and personal trip records must be stored securely and must not be shared with other users without the explicit consent of the data owner. The collected data must be used exclusively for informational and research purposes related to the BBP system and must not be exploited for commercial purposes.

Domain Assumptions

D1: Users have access to mobile devices capable of collecting basic location and movement data during bike trips.
D2: Users provide truthful and reasonably accurate information when recording trips or manually inserting bike path evaluations.
D3: GPS and sensor data collected during trips may be affected by environmental conditions and device limitations.
D4: Users are willing to share qualitative feedback about bike paths based on their personal experience.
D5: External mapping services used by BBP provide sufficiently accurate and up-to-date geographic information.
D6: Network connectivity may be intermittent during bike trips, and the system tolerates temporary unavailability of online services.
D7: The information provided by users may be subjective and incomplete, and the system does not guarantee the optimality of suggested routes.

Dependencies

- BBP relies on external mapping services to visualize bike paths.
- BBP may integrate external weather services to enrich trip information when such services are available.

Constraints

- The system must comply with applicable data protection regulations, including GDPR, when storing and processing personal and location data.
- BBP does not guarantee real-time data availability due to network limitations and data uncertainty.
- The system does not control physical road infrastructure, traffic conditions, or real-time navigation behavior.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The Best Bike Paths (BBP) system provides a web-based user interface designed to be simple, intuitive, and accessible to cyclists with heterogeneous technical backgrounds. The interface focuses on supporting information consultation and basic data insertion tasks, without relying on complex real-time interactions or advanced visualization features.

The user interface is implemented as a single-page application (SPA) with a persistent top navigation bar that allows users to access the main functionalities of the system. Navigation between pages is performed without full page reloads, ensuring a smooth and responsive interaction experience.

Depending on the authentication status, different functionalities are made available. Unregistered users can explore routes and consult community-provided information, while registered users can additionally access personal trip records and manually insert information about bike paths. Functionalities that require authentication are not accessible to unregistered users and are clearly indicated in the navigation bar.

Home Page The Home page represents the entry point of the BBP system. It provides a brief introduction to the application and explains its main purpose, namely supporting cyclists in exploring bike paths and sharing personal experiences. From this page, users can navigate to the main consultation functionalities without authentication.

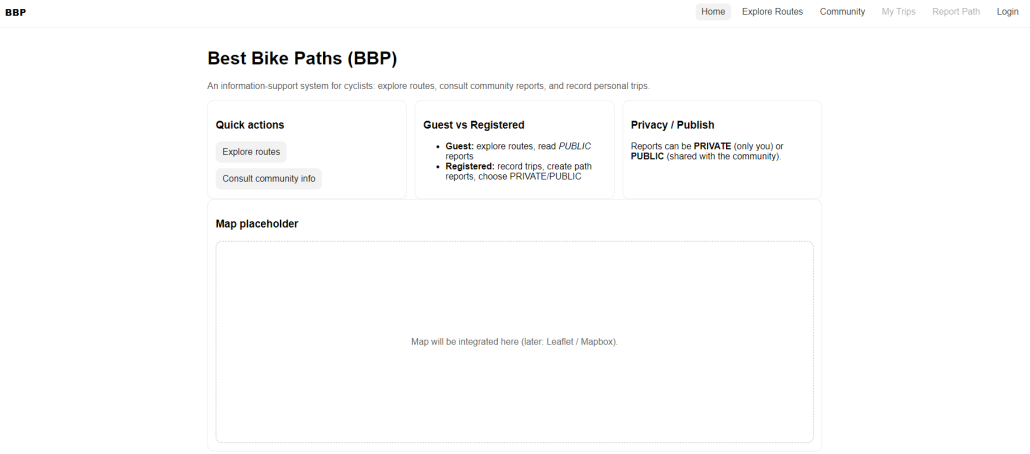

Figure 6: BBP Home Page

Explore Routes Page The Explore Routes page allows users to specify an origin and a destination and visualize possible bike paths on a map. This functionality is available to both registered and unregistered users. The interface is centered around a simple form for entering locations and a map-based visualization area used to present the retrieved routes.

Community Page The Community page allows all users to consult information about bike paths that has been made public by registered users. The interface presents qualitative evaluations and observations associated with bike paths, supporting users in making informed route selection decisions based on community feedback.

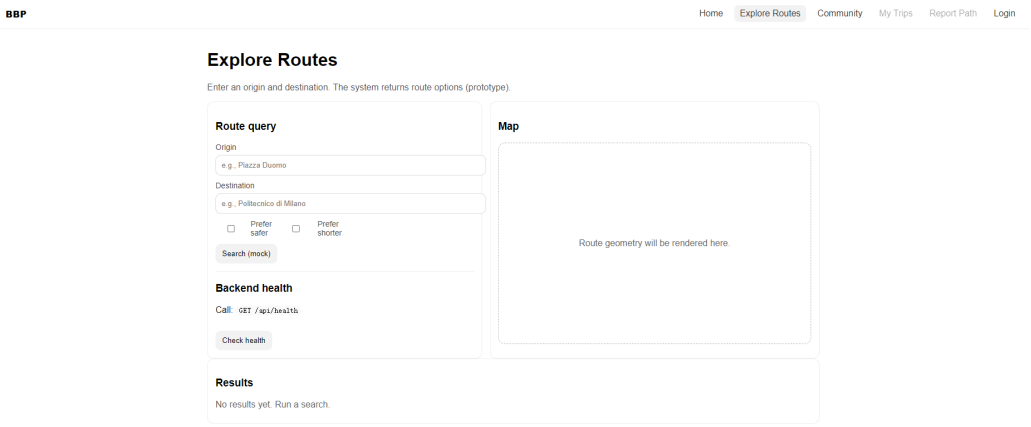

Figure 7: Explore Routes Interface

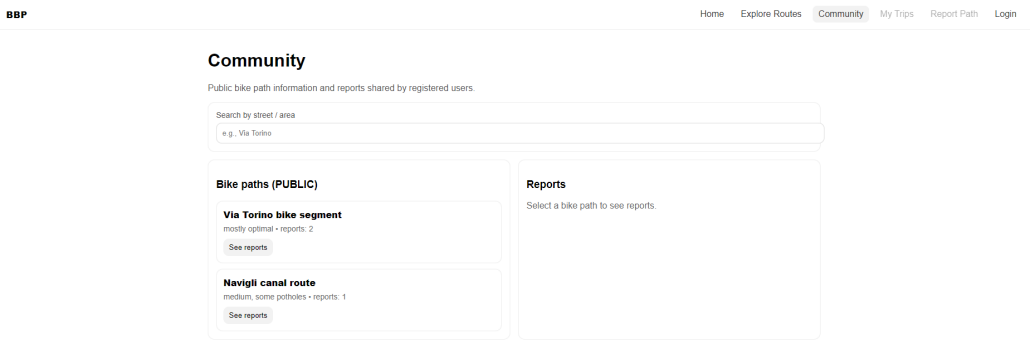

Figure 8: Community Information Page

Login Page The Login page enables users to authenticate and gain access to functionalities reserved to registered users. The interface relies on a minimal form-based design, requiring only basic credentials. After a successful login, additional pages that are not accessible to unregistered users become available.

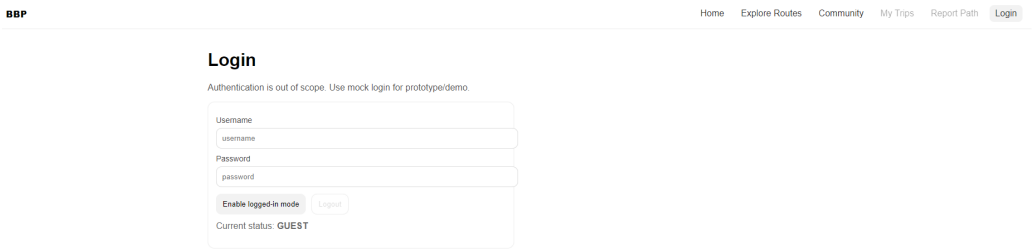

Figure 9: User Login Page

My Trips Page The My Trips page is accessible only to registered users. It allows authenticated users to consult their previously recorded bike trips and view basic trip-related information. Unregistered users cannot access this page, and any attempt to do so requires prior authentication.

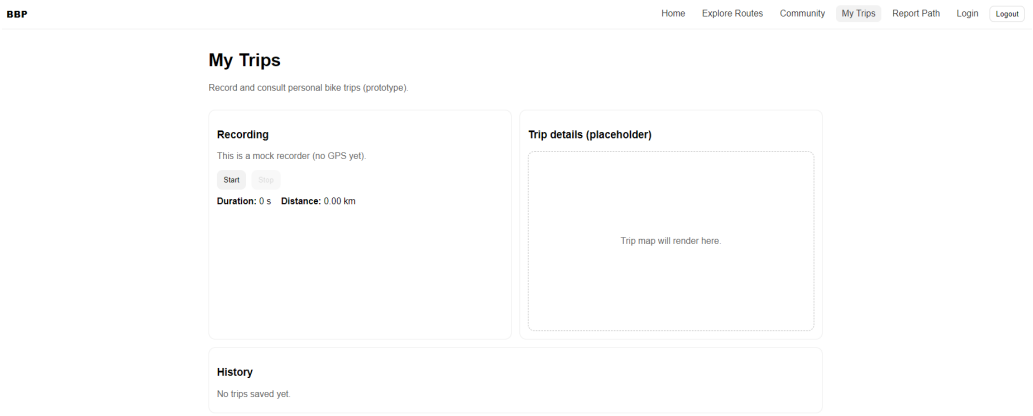

Figure 10: My Trips Page

Report Path Page The Report Path page allows registered users to manually insert information about bike paths, including qualitative evaluations and observations. Users can also decide whether the inserted information should be kept private or made public. This page is not accessible to unregistered users and requires authentication before use.

Overall, the user interface of BBP is designed to clearly separate public and restricted functionalities, enforce access control at the interface level, and provide a coherent interaction experience aligned with the system requirements.

3.1.2 Hardware Interfaces

BBP interacts with hardware components indirectly through user devices. In particular, the system may access location and basic movement data provided by mobile devices used by registered users.

The system does not directly control hardware components and does not require specialized hardware beyond standard consumer mobile devices and personal computers.

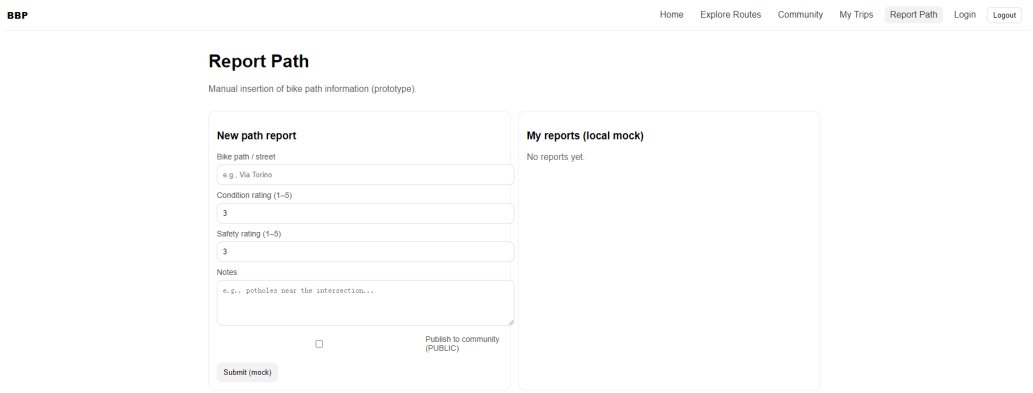

Figure 11: Manual Insertion of Bike Path Information

3.1.3 Software Interfaces

BBP interfaces with external software services to support its functionalities. In particular, the system may rely on:

- external mapping services for the visualization of geographic information and routes;
- external weather services to enrich trip data when available.

BBP also interfaces with a backend database system to store and retrieve user data, trip records, and information about bike paths.

3.1.4 Communication Interfaces

BBP uses standard web-based communication mechanisms to exchange data between clients and the backend system. Communication is based on HTTP and follows a request-response model.

No real-time communication or continuous data streaming is required for the operation of the system.

3.2 Functional Requirements

User registration and authentication

R1: The system shall allow users to register an account by providing basic personal information.
R2: The system shall allow registered users to authenticate using their credentials.
R3: The system shall distinguish between registered and unregistered users
R4: The system shall restrict trip recording and manual path insertion to registered users only.
R5: The system shall allow registered users to log out.

Trip recording

R6: The system shall allow registered users to start a bike trip recording.
R7: The system shall collect basic location and movement data during a trip.
R8: The system shall allow users to stop an ongoing trip recording.
R9: The system shall store recorded trips in the user's personal history.

Trip visualization

R10: The system shall allow registered users to visualize recorded trips on a map.
R11: The system shall provide basic statistics for each recorded trip.
R12: The system shall allow users to browse their trip history.

Manual insertion of bike path information

R13: The system shall allow registered users to manually insert information about bike paths.
R14: The system shall allow users to associate qualitative evaluations and observations with a bike path.
R15: The system shall allow users to decide whether inserted information is publishable.
R16: The system shall store non-published information as private user data.

Consultation of community information

R17: The system shall allow all users, including unregistered users, to consult bike path information that has been marked as publishable.
R18: The system shall clearly distinguish between published community information and private user data.

Route exploration

R19: The system shall allow users to specify an origin and a destination.
R20: The system shall visualize available bike paths between the specified points.
R21: The system shall present route information based on community-provided data.

3.2.1 Use Case Diagram

The main interactions between users and the BBP system are represented through a use case diagram. The diagram highlights the core functionalities available to registered and unregistered users and their relationships with the system.

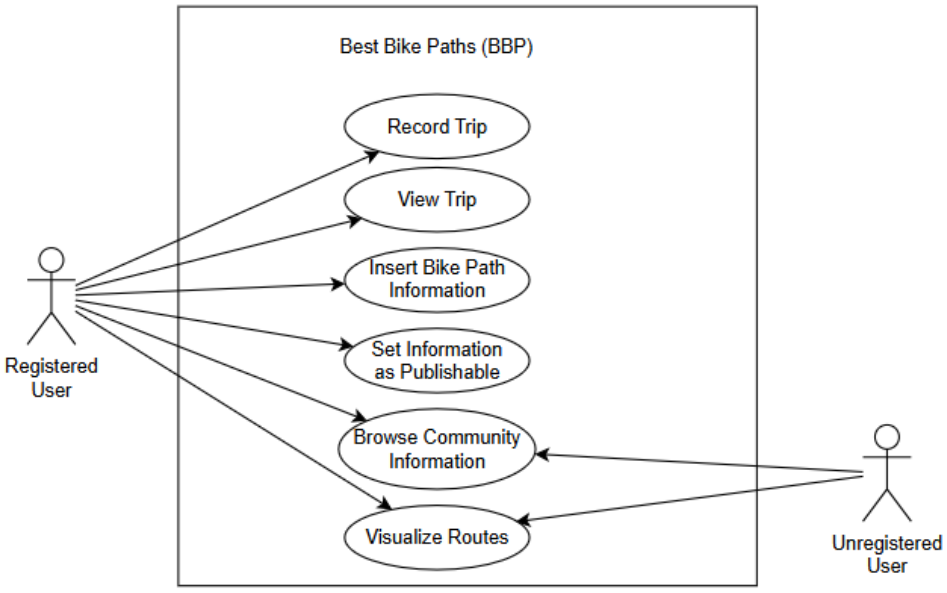

Figure 12: Use case diagram of the BBP system

Use Case Descriptions

The following use cases describe the main interactions between users and the BBP system. Each use case is associated with one or more functional requirements defined above.

3.2.2 Use Cases

UC1 – Record Trip

Name	Record Trip
Actors	Registered User
Entry Condition	The user is authenticated in the BBP system and has granted the application permission to access location data.
Event Flow	(a) The user selects the option to start recording a new bike trip. (b) The system starts collecting basic trip data (e.g., location and movement data over time). (c) The user ends the trip recording. (d) The system stores the trip together with the collected data in the user's personal history.
Exit Condition	A new bike trip is successfully stored in the system and associated with the user's account.
Exception	If location data cannot be collected, the system notifies the user and allows the user to stop the recording without storing incomplete trip data.

Table 1: Use Case Specification – Trip Recording

UC2 – View Trip

Name	**View Trip**
Actors	Registered user
Entry Condition	The user is registered and authenticated in the BBP system.
Event Flow	(a) The user requests to access the list of recorded trips. (b) The system displays the list of trips available to the user. (c) The user selects a specific trip from the list. (d) The system displays basic information about the selected trip and a visualization of the recorded path.
Exit Condition	The selected trip information and its visualization are successfully displayed to the user.
Exception	If no recorded trips are available, the system displays an empty list or a message indicating that no trips exist.

Table 2: Use Case Specification – View Trip

Name	**Manual insertion of bike path information**
Actors	Registered user
Entry Condition	The user is registered and authenticated in the BBP system.
Event Flow	(a) The user selects the option to insert information about a bike path. (b) The user provides the required information (e.g., path description and qualitative evaluation). (c) The system asks the user whether the provided information should be made publishable. (d) The user selects whether the information is publishable or not. (e) The system stores the inserted information together with the selected visibility option.
Exit Condition	A new bike path information entry is successfully stored in the system and marked as publishable or non-publishable according to the user's decision.
Exception	If mandatory information is missing, the system notifies the user and requests a correction before storing the entry.

Table 3: Use Case Specification – Manual insertion of bike path information

Name	**Browse Community Information**
Actors	Unregistered user, Registered user
Entry Condition	The user has access to the BBP system.
Event Flow	(a) The user accesses the community information section. (b) The user optionally applies simple filters. (c) The system displays available publishable community information. (d) The user selects an entry to view its details. (e) The system displays the details of the selected entry.
Exit Condition	Community information is successfully displayed to the user.
Exception	If no publishable community information is available, the system informs the user.

Table 4: Use Case Specification – Browse Community Information

Name	**Visualize Routes between Origin and Destination**
Actors	Unregistered user, Registered user
Entry Condition	The user has access to the BBP system.
Event Flow	(a) The user specifies an origin and a destination. (b) The user requests route visualization. (c) The system retrieves available information useful to determine candidate routes. (d) The system visualizes one or more possible routes on a map.
Exit Condition	One or more candidate routes are successfully visualized to the user.
Exception	If no route can be shown due to missing or insufficient information, the system informs the user.

Table 5: Use Case Specification – Visualize Routes between Origin and Destination

UC3 – Manual insertion of bike path information

UC4 – Browse Community Information

UC5 – Visualize Routes between Origin and Destination

3.2.3 Sequence Diagrams

Please note that the notifications here are modeled at a high level, without considering design and architectural aspects, and omitting synchronization details. They have been introduced just to represent the interaction with users.

UC1 – Record Trip

UC2 – View Trip

UC3 – Manual insertion of bike path information

UC4 – Browse Community Information

UC5 – Visualize Routes between Origin and Destination

3.2.4 Requirement Mapping

Requirements R1–R5 are supporting requirements related to user registration and authentication. They enable the correct identification of users and access control mechanisms required by Goals G1 and G2, but do not directly correspond to a specific system goal.

3.3 Performance Requirements

BBP is not subject to strict real-time performance constraints. The system is required to provide acceptable response times for common user operations such as data insertion and route visualization.

Performance requirements are defined qualitatively, as the system is intended to support informational use rather than time-critical decision making.

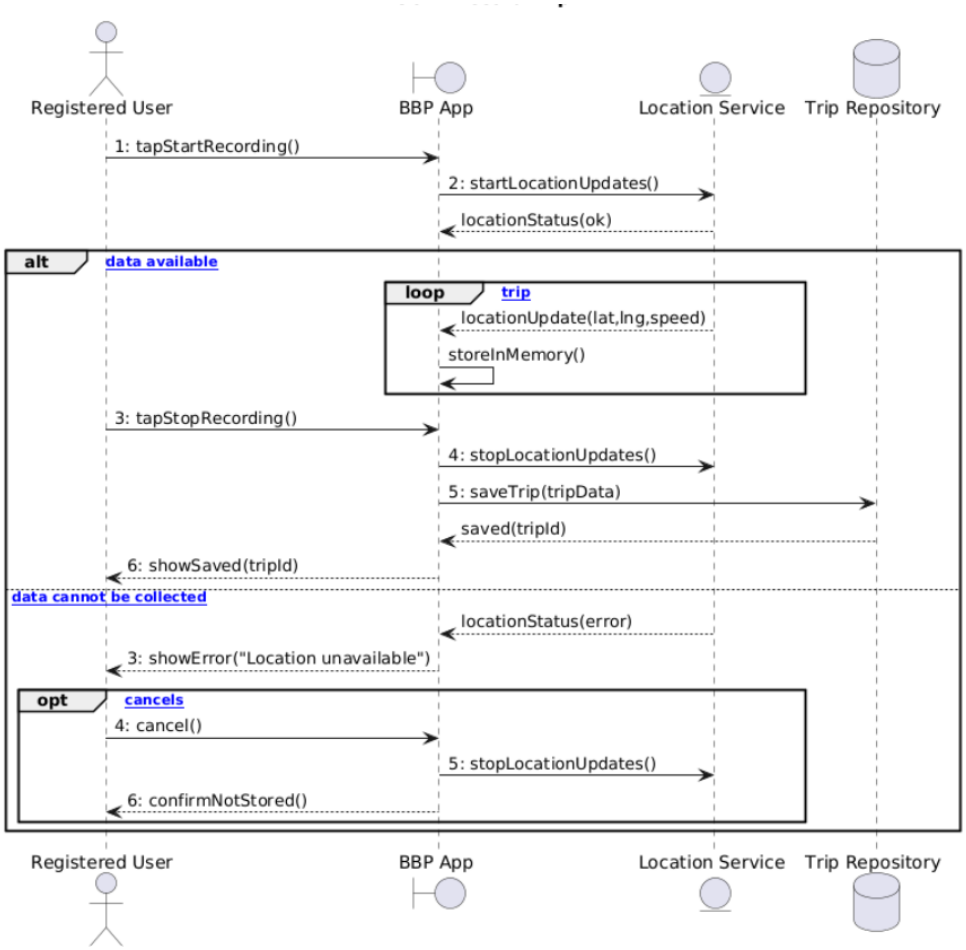

Figure 13: Sequence Diagram - UC1

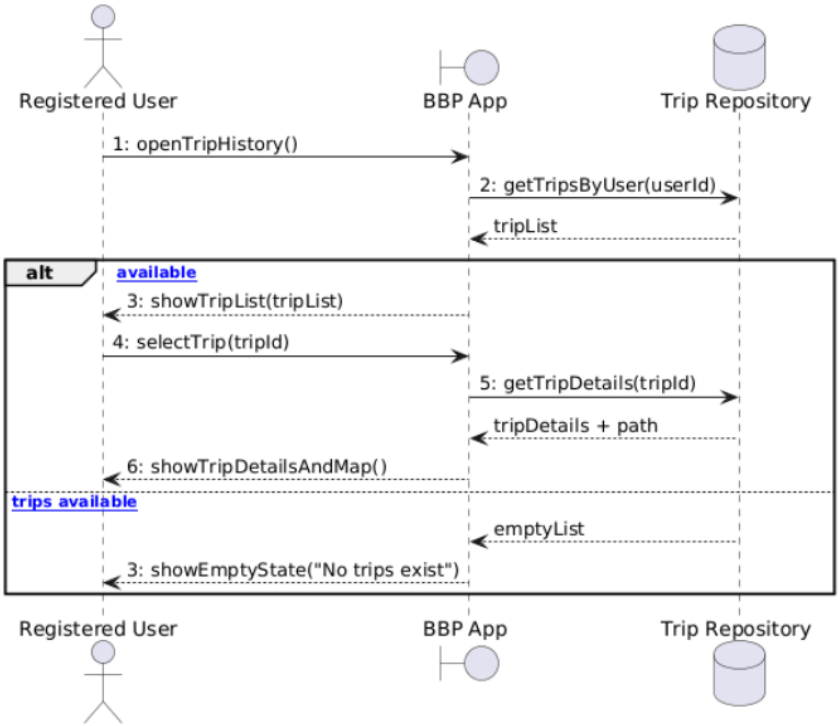

Figure 14: Sequence Diagram - UC2

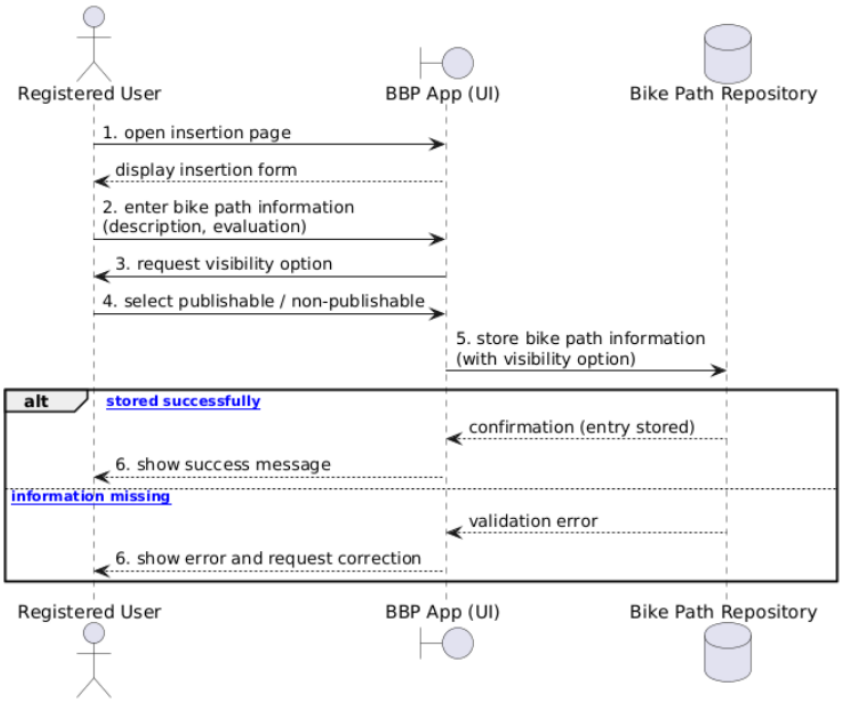

Figure 15: Sequence Diagram - UC3

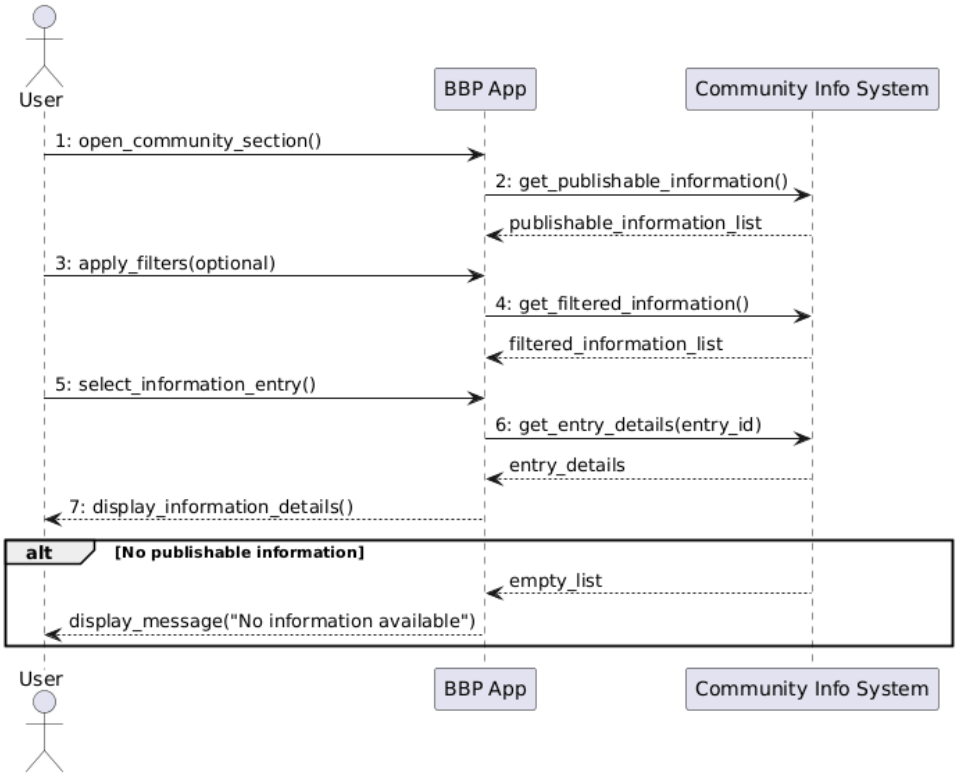

Figure 16: Sequence Diagram - UC4

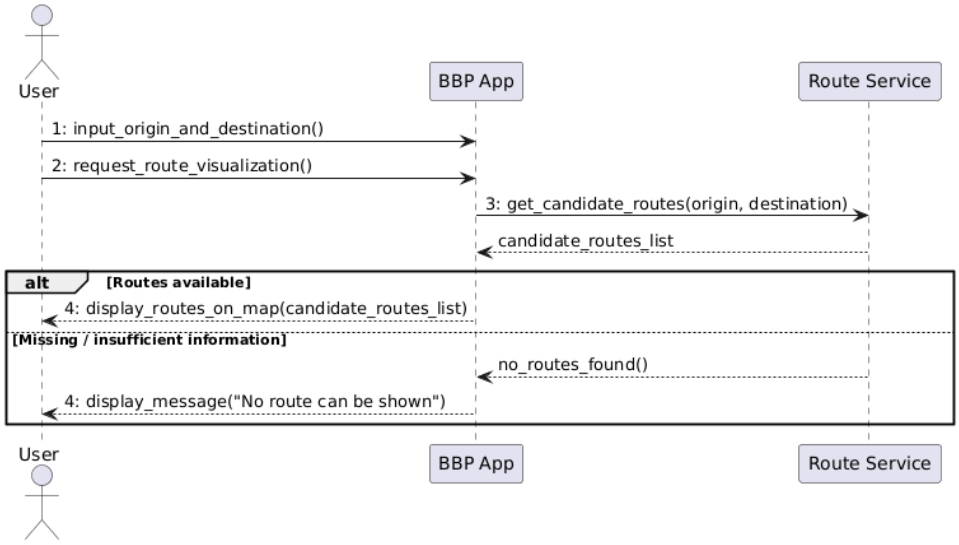

Figure 17: Sequence Diagram - UC5

G1: Cyclists are supported in recording their personal bike trips, allowing them to keep track of their cycling activities and access basic statistics related to their rides.
Related Requirements R6: The system shall allow registered users to start a bike trip recording. R7: The system shall collect basic location and movement data during a trip. R8: The system shall allow users to stop an ongoing trip recording. R9: The system shall store recorded trips in the user's personal history. R10: The system shall allow registered users to visualize recorded trips on a map. R11: The system shall provide basic statistics for each recorded trip. R12: The system shall allow users to browse their trip history.
Design Assumptions and Constraints D1: Users have access to mobile devices capable of collecting basic location and movement data during bike trips. D3: GPS and sensor data collected during trips may be affected by environmental conditions and device limitations. D6: Network connectivity may be intermittent during bike trips, and the system tolerates temporary unavailability of online services.

Table 6: Goals–Requirements–Design Assumptions Mapping for G1

G2: Users are enabled to manually insert information about bike paths, including path conditions and relevant observations, and to decide whether such information can be made publishable to the community.
Related Requirements
R13: The system shall allow registered users to manually insert information about bike paths.
R14: The system shall allow users to associate qualitative evaluations and observations with a bike path.
R15: The system shall allow users to decide whether inserted information is publishable.
R16: The system shall store non-published information as private user data.
R17: The system shall allow all users, including unregistered users, to consult bike path information that has been marked as publishable.
R18: The system shall clearly distinguish between published community information and private user data.
Design Assumptions and Constraints
D2: Users provide truthful and reasonably accurate information when manually inserting bike path evaluations.
D4: Users are willing to share qualitative feedback about bike paths based on their personal experience.
D7: The information provided by users may be subjective and incomplete.

Table 7: Goals–Requirements–Design Assumptions Mapping for G2

G3: The system supports the visualization of possible bike paths between a user-specified origin and destination, helping cyclists explore available routes and select suitable paths.
Related Requirements
R19: The system shall allow users to specify an origin and a destination.
R20: The system shall visualize available bike paths between the specified points.
R21: The system shall present route information based on community-provided data.
Design Assumptions and Constraints
D5: External mapping services used by BBP provide sufficiently accurate and up-to-date geographic information.
D7: The information provided by users may be subjective and incomplete, and the system does not guarantee the optimality of suggested routes.

Table 8: Goals–Requirements–Design Assumptions Mapping for G3

3.4 Design Constraints

3.4.1 Standards Compliance

BBP shall comply with applicable standards and regulations concerning data protection and privacy, particularly with respect to the handling of personal and location data.

3.4.2 Hardware Limitations

The system shall operate on standard consumer hardware, including personal computers and mobile devices. BBP shall not require specialized sensors or dedicated hardware components.

3.4.3 Any Other Constraint

Network connectivity may be intermittent, especially during bike trips. The system shall tolerate temporary unavailability of external services without compromising data integrity.

3.5 Software System Attributes

3.5.1 Reliability

BBP shall ensure that stored data is preserved consistently and is not lost in case of temporary system failures.

3.5.2 Availability

BBP shall be available for use during normal operating conditions, excluding scheduled maintenance and unforeseen outages.

3.5.3 Security

BBP shall protect user data from unauthorized access. Only registered users shall be allowed to insert or modify information, while consultation functionalities shall be accessible to all users.

3.5.4 Maintainability

BBP shall be designed in a modular way to facilitate maintenance and future extensions. Changes to one component of the system should not require major modifications to other components.

3.5.5 Portability

BBP shall be deployable on common operating systems and standard web environments without requiring platform-specific adaptations.

4 Formal Analysis Using Alloy

4.1 Verification Objectives

In this section, Alloy is used to formally analyze and validate key access-control and visibility properties of the BBP system. Since BBP manages both private user data (such as recorded trips and non-published bike path information) and community-visible information, it is crucial to ensure that requirements related to data ownership and publishability are consistently enforced and do not lead to unintended information disclosure.

4.2 Properties to Verify

The following properties capture the most critical access-control and visibility requirements of the BBP system and are selected for formal verification using Alloy.

P1 – Trip ownership restricted to registered users

Property: Unregistered users must never own recorded trips.
Motivation: Trips represent personal cycling activity data and must be protected from unauthorized creation or ownership. This property enforces the access-control requirement that trip recording is restricted to registered users.
Related requirements: R4, R6, R9 (supporting Goal G1).

P2 – Each trip has exactly one owner

Property. Every recorded trip is associated with exactly one registered user.
Motivation. A trip must be traceable to a unique user account; otherwise, personal history management and accountability would become ambiguous.
Related requirements. R9 (supporting Goal G1).

P3 – Only registered users can create bike path information

Property. Bike path information entries can only be created by registered users.
Motivation. Manual insertion of bike path information is an authenticated contribution activity. Restricting content creation to registered users prevents anonymous misuse and improves the reliability of community-provided data.
Related requirements. R4, R13 (supporting Goal G2).

P4 – Publishable information is visible to all users

Property. If a bike path information entry is marked as publishable, it must be visible to all users, including both registered and unregistered users.
Motivation. Publishable information is intended for community sharing and must therefore be accessible to every user of the system.
Related requirements. R17 (supporting Goal G2).

P5 – Non-published information is visible only to its creator

Property. If a bike path information entry is not marked as publishable, it must be visible only to its creator.
Motivation. This property enforces the privacy guarantees of the system by ensuring that non-published information is not disclosed to other users or to the community.
Related requirements. R16, R18 (supporting Goal G2).

4.3 Alloy Model

```
// ================================
// Best Bike Paths (BBP)
// Alloy Model for Formal Analysis
// ================================


// ---------- Users and roles ----------
abstract sig User {}
sig RegisteredUser extends User {}
sig UnregisteredUser extends User {}

// ---------- Trips ----------
sig Trip {
  owner: one RegisteredUser
}

// ---------- Bike path information ----------
abstract sig Bool {}
one sig True extends Bool {}
one sig False extends Bool {}

sig BikePathInfo {
  creator: one RegisteredUser,
  publishable: one Bool
}

// ---------- Visibility predicate ----------
fun canView[u: User]: set BikePathInfo {
  { i: BikePathInfo |
      i.publishable = True
      or (u in RegisteredUser and i.creator = u)
  }
}

// ---------- System invariants ----------
fact AccessControl {

  // Trips always belong to registered users
  all t: Trip | t.owner in RegisteredUser

  // Bike path information is created by registered users
  all i: BikePathInfo | i.creator in RegisteredUser

  // Visibility rules
  all i: BikePathInfo |
    (i.publishable = True =>
        all u: User | i in canView[u])
    and
    (i.publishable = False =>
        all u: User |
          (i in canView[u]) iff (u in RegisteredUser and u = i.creator)
    )
}
```

Listing 1: Alloy model for the BBP system

Example Instance Generation

```
pred demo {
  some RegisteredUser
  some UnregisteredUser
  // At least one trip (forces an owner in RegisteredUser)
  some Trip
  // At least one publishable entry and one private entry
  some iPub: BikePathInfo | iPub.publishable = True
  some iPriv: BikePathInfo | iPriv.publishable = False
}
run demo for 4
```

Listing 2: Predicate used to generate an illustrative instance

Instances

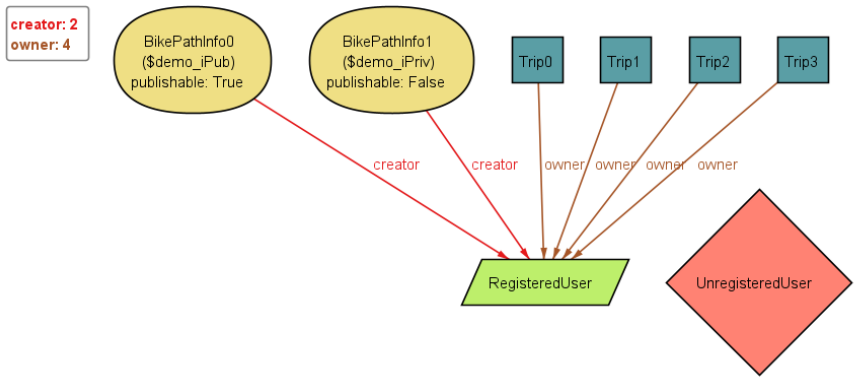

Figure 18: Example Instance - 1

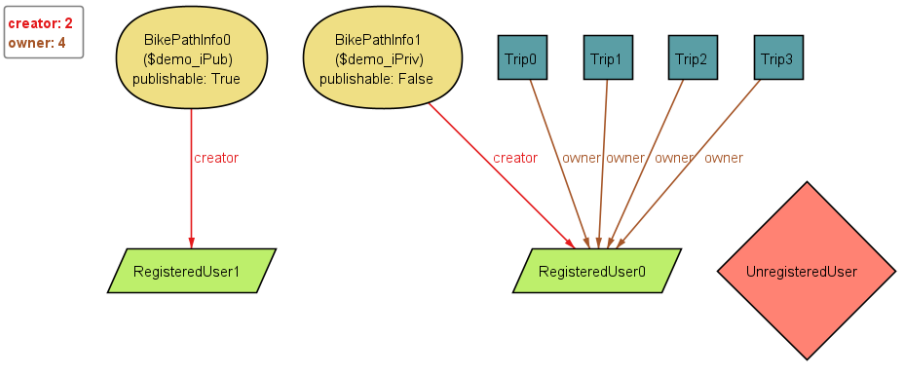

Figure 19: Example Instance - 2

4.4 Assertions and Checks

P1 – Trip ownership restricted to registered users

```
assert NoTripForUnregistered {
  no u: UnregisteredUser |
```



```
    some t: Trip | t.owner = u
}

check NoTripForUnregistered for 6
```

Listing 3: Assertion and check for Property P1

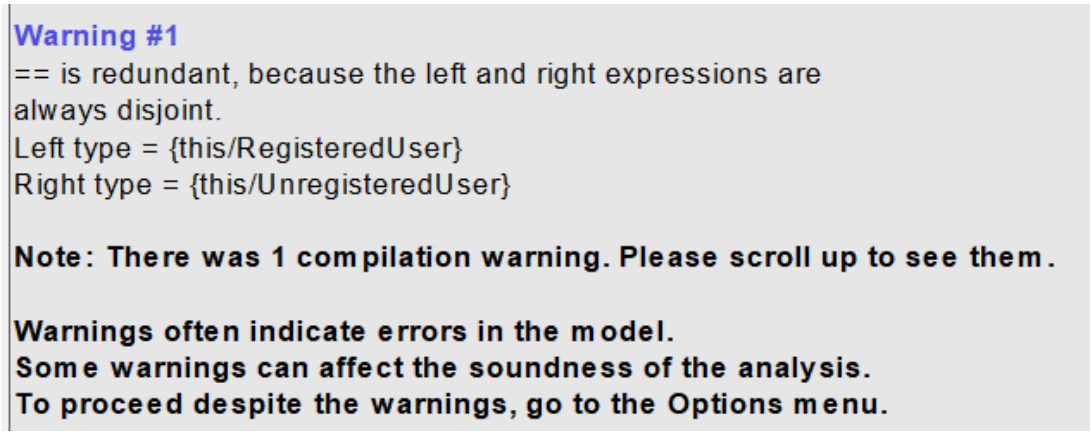

Figure 20: Alloy check result for Property P1.

As shown in Figure 20, Alloy did not generate any counterexample for Property P1 within the selected scope. The reported warning is caused by the static type constraints of the model, which already restrict trip ownership to registered users. Therefore, the warning does not affect the validity of the verification result.

P2 – Each trip has exactly one owner

```
assert UniqueTripOwner {
  all t: Trip | one t.owner
}

check UniqueTripOwner for 6
```

Listing 4: Assertion and check for Property P2

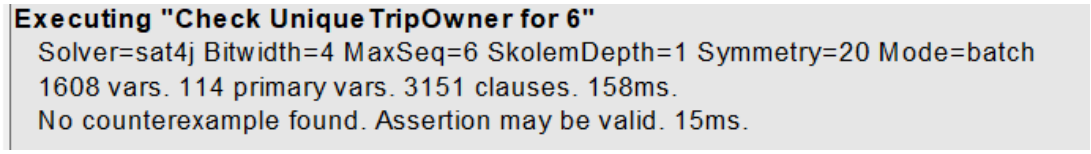

Figure 21: Alloy check result for Property P2.

P3 – Only registered users can create bike path information

```
assert OnlyRegisteredCanCreateInfo {
  all i: BikePathInfo | i.creator in RegisteredUser
}

check OnlyRegisteredCanCreateInfo for 6
```

Listing 5: Assertion and check for Property P3


```
Executing "Check OnlyRegisteredCanCreateInfo for 6"
  Solver=sat4j Bitwidth=4 MaxSeq=6 SkolemDepth=1 Symmetry=20 Mode=batch
  1604 vars. 114 primary vars. 3067 clauses. 38ms.
  No counterexample found. Assertion may be valid. 7ms.
```

Figure 22: Alloy check result for Property P3.

P4 – Publishable information is visible to all users

```
assert PublishableVisibleToAll {
  all i: BikePathInfo |
    i.publishable = True implies
      all u: User | i in canView[u]
}

check PublishableVisibleToAll for 6
```

Listing 6: Assertion and check for Property P4

```
Executing "Check PublishableVisibleToAll for 6"
  Solver=sat4j Bitwidth=4 MaxSeq=6 SkolemDepth=1 Symmetry=20 Mode=batch
  1739 vars. 120 primary vars. 3298 clauses. 26ms.
  No counterexample found. Assertion may be valid. 4ms.
```

Figure 23: Alloy check result for Property P4.

P5 – Non-published information is visible only to its creator

```
assert PrivateVisibleOnlyToCreator {
  all i: BikePathInfo |
    i.publishable = False implies
      all u: User |
        (i in canView[u]) iff
        (u in RegisteredUser and u = i.creator)
}

check PrivateVisibleOnlyToCreator for 6
```

Listing 7: Assertion and check for Property P5

```
Executing "Check PrivateVisibleOnlyToCreator for 6"
  Solver=sat4j Bitwidth=4 MaxSeq=6 SkolemDepth=1 Symmetry=20 Mode=batch
  1798 vars. 120 primary vars. 3717 clauses. 26ms.
  No counterexample found. Assertion may be valid. 5ms.
```

Figure 24: Alloy check result for Property P5.

5 Effort Spent

The following table displays the number of hours each group member spent on the various sections of the document. Please note that this division is only approximate, and each section still required the collaboration of all members.

Table 9: Distribution of effort among group members

Member	Chapter 1	Chapter 2	Chapter 3	Chapter 4
Hu Peng	2	12	9	6
Zhang Zihao	3	8	12	5

6 Use Of AI Tool

Generative AI tools were used during the preparation of this project as supporting instruments for well-defined and limited tasks. Their use did not replace the authors' reasoning or decision-making processes, and full responsibility for the content of this document remains with the authors. The following generative AI tool was used:

- **ChatGPT (OpenAI)**: used as a conversational assistant.

Inputs Provided

The inputs provided to the generative AI tool consisted of:

- Natural language prompts describing specific tasks, such as explaining formal specifications, refining technical text, translating technical content between languages, and generating draft versions of Alloy assertions;
- Excerpts from the project specification and intermediate versions of the document, provided solely for clarification, translation, and improvement purposes;
- Descriptions of intended system functionalities used to obtain guidance on the structure and interpretation of UML diagrams.

Outputs Obtained

The outputs obtained from the generative AI tool included:

- Suggestions for improving clarity, structure, and grammar of technical sections;
- Assistance with translating technical content while preserving its original meaning and terminology;
- Draft formulations of Alloy models, assertions, and explanatory text;
- Conceptual guidance on UML diagrams, including explanations of diagram elements and relationships, to support correct modeling decisions;
- High-level explanations of formal verification concepts to support understanding.

Verification and Integration

All outputs generated by the AI tool were carefully reviewed, verified, and, if necessary, modified by the authors before being integrated into the final document. Translated content was checked for consistency with the original technical meaning. UML diagrams were manually created and validated by the authors based on the project requirements. The correctness of Alloy models and assertions was independently validated using the Alloy Analyzer, and all technical decisions were made by the authors. Generated content was used only as a starting point and was adapted to ensure consistency with the project requirements and learning objectives.

References

