POLITECNICO
MILANO 1863

COMPUTER SCIENCE AND ENGINEERING
SOFTWARE ENGINEERING II
2025 - 2026

RASD

Requirement Analysis and Specification Document

Best Bike Paths

Authors:
Leonardo Guglielmi, Francesco Lo Conte

Version:
1.0
(December 9, 2025)

Contents

1 Introduction

1.1 Purpose

The growing interest in cycling, whether as a recreational activity, a means of transportation, or a sport, brings with it a significant challenge: finding routes that are not only efficient, but also safe and well-maintained. Cyclists often lack reliable and up-to-date information on trail conditions, such as the presence of potholes, obstacles, or roads with little traffic. At the same time, many cyclists meticulously log their trips to monitor their performance, collecting valuable data that, however, remains siloed. This creates a gap where vital community knowledge about trail quality is not easily shared or accessible. "Best Bike Paths" (BBP) aims to provide a solution. Commissioned by a cyclists' association, BBP will be a software system designed to create and manage a community-driven inventory of cycling routes. The platform will help bridge this information gap by allowing registered users to track their trips while simultaneously submitting detailed information on the condition of their routes. Other users, registered or not, will then be able to use this collective data to find and display the best possible cycling routes between two points, ranked by a quality score.

1.1.1 Goals

- **G1:** A registered user wants to track their personal cycling activities and related performance statistics.
- **G2:** A registered user wants to contribute to the community inventory by sharing reliable information on the condition of the trails (e.g. quality, obstacles, potholes).
- **G3:** Any user (registered or not) wants to find and view the best cycling route between an origin and a destination, based on up-to-date and relevant data.
- **G4:** The cycling association aims to provide the community with a tool to create, consult, and maintain a reliable and centralized inventory of cycling routes.

1.2 Scope

The Best Bike Paths (BBP) system is designed to create, manage, and distribute a community-driven inventory of cycling paths, acting as a mediator between the physical conditions of the road network and the cyclists' need for safety. The scope of the application covers the entire lifecycle of path data: from its collection via mobile devices to its aggregation into a quality metric (Path Score) utilized for routing.

For this project, the following users interacting with the system have been identified:

- **Registered User**
- **Any User**

A Registered User will be able to use the application to log and store their trips, tracking their cycling activities and related statistics. When available, this data can be enriched with weather information retrieved from external services. Furthermore, this user is the primary contributor to the inventory. They can enter route information in two ways:

1. In **manual mode**, by actively specifying the route status (e.g., optimal, requires maintenance) and the presence of obstacles (e.g., potholes).
2. In **automatic mode**, by allowing the app to acquire data from GPS and mobile device sensors while cycling, in order to automatically detect potential problems such as potholes.

For automatically collected data, the system will ask the user to confirm or correct the information before making it available to the community. Once confirmed or manually entered, this information becomes publishable.

Any user, whether registered or not, can benefit from the collected information. The user can specify a starting point and a destination. The system leverages third-party mapping services to identify valid physical routes and overlays them with BBP's inventory data. If a route is present in the inventory, it is displayed with its Path Score; otherwise, it is displayed without it. If multiple routes exist, BBP will present them based on this score, calculated based on the route's status derived from user-confirmed data.

1.2.1 World phenomena

- **WP1:** A registered user starts a cycling activity.
- **WP2:** Any user searches for a cycling route between two places.
- **WP3:** A registered user contributes contribute to the BBP inventory.
- **WP4:** While pedaling a registred user ecounters some kind of problem on the route.

1.2.2 Shared phenomena

World controlled

- **SP_WC1:** The registered user launches starts to register the trip with the application.
- **SP_WC2:** The registered user stops to register the trip with the application.
- **SP_WC3:** The registered user opens the interface for manually entering route information.

- **SP_WC4:** The registered user enters the data (e.g. "optimal" status, "hole" presence) and sends the manual entry form.
- **SP_WC5:** The registered user selects a notification or confirmation request for automatically detected data.
- **SP_WC6:** The registered user confirms to validate automatically the detected data (e.g. a pothole).
- **SP_WC7:** The registered user deletes to invalidate an automatically detected data (false positive).
- **SP_WC8:** The registered user modifies an automatically detected piece of data (e.g. corrects the position of the hole on the map) and saves the change.
- **SP_WC9:** Any user enters a source and destination address.
- **SP_WC10:** Any user starts the route search.

Machine controlled

- **SP_MC1:** The system shows the statistics of the completed trip to the registered user.
- **SP_MC2:** The system shows the weather data associated with the trip to the registered user.
- **SP_MC3:** The system presents the Registered User with a confirmation request for automatically detected data.
- **SP_MC4:** The system shows the user a map with the cycling routes found between the origin and the destination.
- **SP_MC5:** The system displays the details of a route, including its score and confirmed obstacles.
- **SP_MC6:** The system displays an error message (e.g., "Weather service unavailable").

1.3 Definitions, Acronyms, Abbreviations

This section contains the definitions for people that may not know what a specific concept is, acronyms and abbreviations used throughout the document.

1.3.1 Definitions

- **Bike Path:** a route deemed suitable for cycling. This includes paths with a proper bike track or roads where cars are rare and speed limits are compatible with the average speed of a bike.

- **Trip:** a personal record of a user's cycling activity, stored by the system to track performance metrics like distance and speed.
- **Publishable Information:** data about a bike path (e.g., status, obstacles) that a registered user has either entered making it available to the wider community.
- **Path Score:** a metric computed by BBP to rank route options. It is based on the status of the path and its effectiveness in getting the user from their origin to their destination.
- **Obstacle:** any significant element or condition on a cycle path that may represent a danger or hindrance to the cyclist (e.g. pothole).

1.3.2 Acronyms

- **BBP:** Best Bike Paths.
- **GPS:** Global Positioning System.
- **API:** Application Programming Interface.

1.3.3 Abbreviations

- **G*:** Goal.
- **WP*:** World Phenomenon.
- **SP*:** Shared Phenomenon.
- **R*:** Requirement.
- **UC*:** Use Case.
- **D*:** Domain Assumption.

Note: asterisks are intended as a replacement for the number.

1.4 Revision history

- **Version 1.0 (17/11/2025)**

1.5 Reference documents

This document is based on the following materials:

- The specification of the RASD and DD assignment of the Software Engineering II course a.y. 2025/26.
- Course slides shared on WeBeep.
- Past Requirement Analysis and Specification Documents.

1.6 Document structure

1. **Introduction:** a brief description of the project. It contains the main goals and objectives that the final system wants to achieve.
2. **Overall description:** this section is a high-level representation of the system and of the interactions of the system with the other actors.
3. **Specific requirements:** a detailed list of all the requirements needed for the system to achieve the goals. It contains valuable information for developers.
4. **Formal analysis using Alloy:** a formal description of the model of the system with Alloy.
5. **Effort spent:** the time spent on each section of the document, for each member of the group.
6. **References:** reference to documents or tools used for writing this document..

2 Overhall description

2.1 Product perspective

2.1.1 Scenarios

[SC1] Registering a new account

User "Zoe" has just downloaded the BBP app in order to monitor her activities on the bicycle, and wants to create a profile. So she creates an account by entering her name, surname, email, birth date, gender, and accepting the privacy policy. Once her information is verified, she receives an email to confirm her mail address. She confirms it, and the account is succesfully created.

[SC2] Logging into account

Registered user "Monica" wants to enter in the BBP app with her account. She opens the BBP app, enters her email and password on the login screen, and submits that information. Then the account information is displayed to her, and she can use all app functionalities.

[SC3] Updating account information

Registered user "Giulio" noticed that he had selected the wrong birth date during account creation. He decides to fix it: he opens the BBP app, goes to the Profile section, and opens the edit screen. On this screen he changes the birth date with the correct one, he confirms the update and the app now displays the correct date.

[SC4] Resetting account password

Registered user "Vittorio" changed mobile device and installed the BBP app, but when he tried to log in, he realized he had forgotten his password. Then from the login page he clicks on the link to reset the password, which takes him to a form in which he enters the account email. After a few seconds, he receives an email which contains a link to reset the password. He opens it, fills out the form with the new password and submits it. Then he tries to log-in again in the app with the new password, successfully logging in.

[SC5] Account deletion

After months of inactivity, registered user "Mirko" decides he no longer wants to cycle and deletes his BBP account. He opens the BBP app, opens the Account section and from the options he selects that one to delete the account. He confirms to the app that he wants to delete his account, he receives an email containing a link to confirm his choice a second time. He opens it, reads the disclaimer and confirms that he wants to delete the account. After a few hours, he receives another email confirming account deletion.

[SC6] Intelligent route planning with successful match (Causal user)

Tourist ”Diana” wants to explore the city by bike but is concerned about traffic and poor roads. She accesses the BBP website without logging in and enters ”Hotel Plaza” as the origin and ”Museo della Scienza” as the destination, receiving two possible paths in response. Diana notices that the shortest route (3 km) has a low ”Path Score”, with several ”Pothole” icons along the way. The alternative, slightly longer route (3.5 km) has an excellent ”Path Score” and it's marked as having excellent conditions. Diana chooses the green route, starts the trip and follows the instructions.

[SC7] Intelligent route planning with unsuccessful match (Casual user)

Casual user ”Mirko” wants to find a route to reach out his friends by bicycle. He opens the BBP app and proceedes as a guest, then searches for a path but the app doesn't find a match. He selects the option to create a new path, selects one of the proposed alternatives and starts the trip.

[SC8] Intelligent route planning with successful match (Registred user)

Registered user ”Giorgio” is planning his daily cycling training ride. He opens the BBP app, logs in and searches for bike paths with a starting point near his home and a length of 30 km. He receives three paths: the first path has a high ”Path Score” but that passes by his ex-wife's house; the second path has a decent ”Path score” with no problem marked; the third one has low ”Path Score” with several potholes marked on the map. Given these options, he chooses the second one and starts the activity.

[SC9] Intelligent route planning with unsuccessful match (Registered user)

Registered user ”Sara” wants to reach her hometown pharmacy by bike. She opens the BBP app, searches for a path from her home to the pharmacy but no match is found. She then selects the option to create a new path, and starts the activity following one of the suggested paths.

[SC10] Starting trip activity

User ”Marco” has selected the path he wants to do by bike, starts the trip by selecting the relative option. By doing so, he's able to see the path he should follow and his position in real-time.

[SC11] Stopping and resuming trip activity

User ”Tony” started an activity, but in the middle of it, he encounters his old friend ”Lorenzo” and stops for a chat. He opens the activity screen and pauses it. Later, when he finished with his friends, he resumes the activity.

[SC12] Automatic activity monitoring and trip data enrichment

Registered user ”Alessandro” is preparing for his weekly training session. He wants to track his performance, including its correlation with weather conditions. He starts recording his activity allowing automatic collection of data for both check path and weather conditions tracking. Once he has finished his trip, he stops the recording and after a little bit he views the trip summary on the app: the path map; the total distance traveled; the average, maximum and minimum speed; maximum, average and minimum altitude; and the weather conditions.

[SC13] Route score assignment

Registered user ”Anna” started and completed her activity with the bicycle. Upon completion, she receives from the app the summary of the activity and a small form to sore the route. She selects the score she wants to give and submits it.

[SC14] Automatic path information update

Registered user ”Carlo” started an activity with automatic monitoring. Almost at the end of the ride, he rode over a pothole. When he arrives at work he checks the BBP app to see whether the pothole has been detected or not. He notices that there two potholes were detected: one approximately in the middle of the path, and another one near his work building. Since he didn’t encountered a pothole in the middle of the path, he selects it and discards it. He then selects the pothole near his work building, confirms it and adds an optional note to be more detailed.

[SC15] Manual path information update 1

Registered user ”Bianca” is riding a popular bike path when she notices that a stretch, previously marked as ”Optimal”, is now blocked by unreported construction. She decides to alert the community: she stops and reports the problem on the BBP app by specifying the bike path, the type of problem, the problem position. She also adds an optional textual note for more detailed, then submits the report, receiving an acknowledgement few seconds before.

[SC16] Manual path information update 2

Registered user ”Edoardo” is riding along a path where a pothole had been reported the previous week. He notices that the pothole has been fixed, so he selects the pothole icon on the map and switches its status to Resolved. After a few hours, he decides to check if the icon on the map has been removed, and finds out that the pothole mark disappeared.

[SC17] Historical performance analysis

Registered User "Alessandra", after months of using BBP, wants to analyze her performance progress. She opens the Trip History app section and looks at the list of all her saved trips. She filters the list by "Last month" and looks at the aggregated graph showing her average speed and the total distance traveled for that period. Then she searches for a specific activity she completed two months ago to check for improvements.

[SC18] Trip deletion

Registered user "Caterina" has an accident during her last recorded trip, and therefore the recorded performances are inaccurate. She opens the BBP app, goes to the Activity History section and searches for the trip she wants to delete. Once she finds it, she selects the option to delete it, she confirms that she wants to do that and then the trip is deleted.

2.1.2 Domain Class Diagram

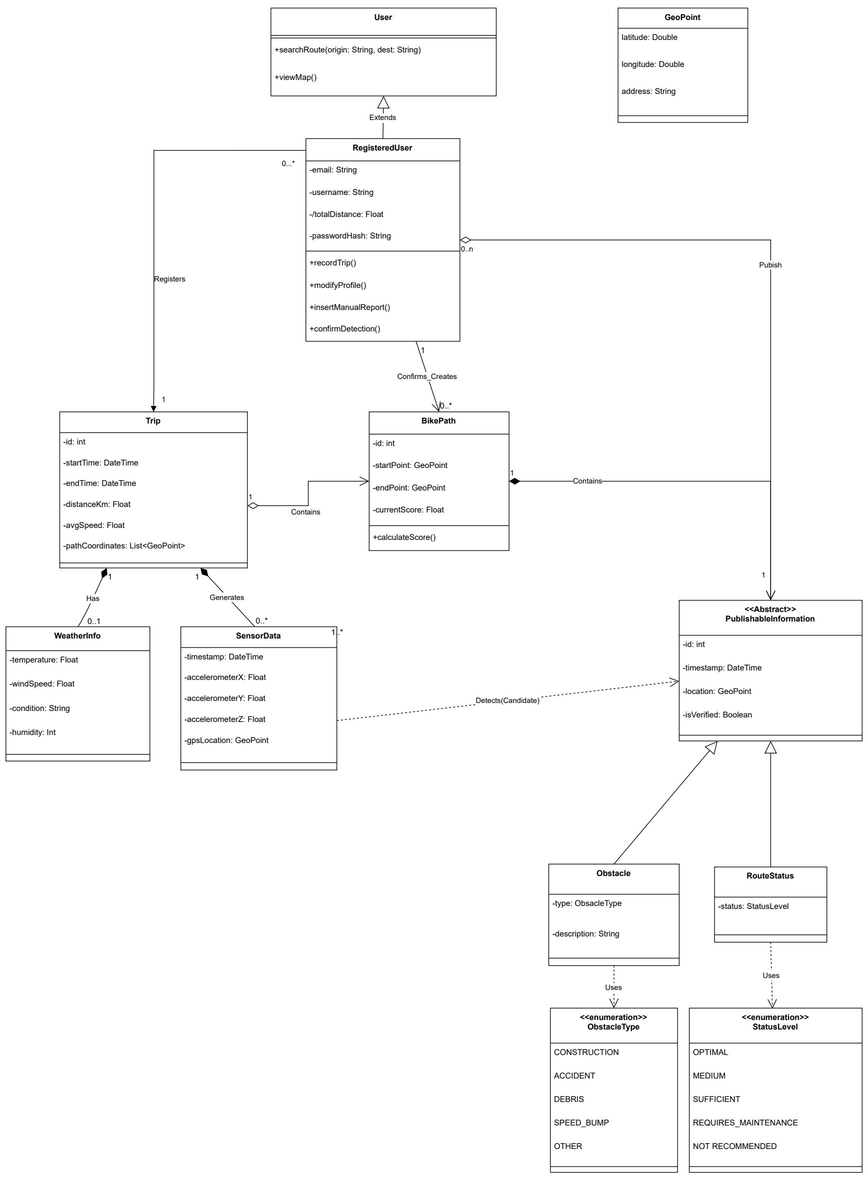

Figure 1: Domain Class Diagram of the BBP system

Figure 1 shows the domain class diagram. The main architectural choices are explained below:

- **User Generalization:** To avoid duplication and facilitate future scalability, the `User` superclass has been introduced. It encapsulates basic functionality accessible to everyone, such as route search and map viewing. The `RegisteredUser` class extends this foundation, adding authentication data and the main writing functionality: `recordTrip()`, `insertManualReport()`, and `confirmDetection()`. This structure allows for easy extension to future roles such as "Administrator" or "Moderator."
- **Information Abstraction and Scoring:** The abstract `PublishableInformation` class was created to logically group all alerts (whether `RouteStatus` or `Obstacle`). This polymorphic approach greatly simplifies the calculation of the Path Score: the system can iterate over a generic list of confirmed information associated with a trip to calculate its score, without having to use separate logic for each type of alert.
- **Sensor Scalability:** Although the assignment specifically mentions potholes, the model correctly links the raw `SensorData` data to the generic `Obstacle` class via the "Detects (Candidate)" dependency. This design ensures that the system can evolve to detect other types of anomalies in the future without changing the core data model.
- **Trip Composition and Data Lifecycle:** There is a composition relationship between `Trip` and its internal data: `WeatherInfo` and `SensorData`. This indicates that this data is closely tied to the trip lifecycle: if a user decides to delete a trip from their history, the associated weather data and raw sensor data will also be automatically removed, preventing data fragmentation and ensuring database cleanliness.

2.1.3 State Diagrams

User Session Lifecycle

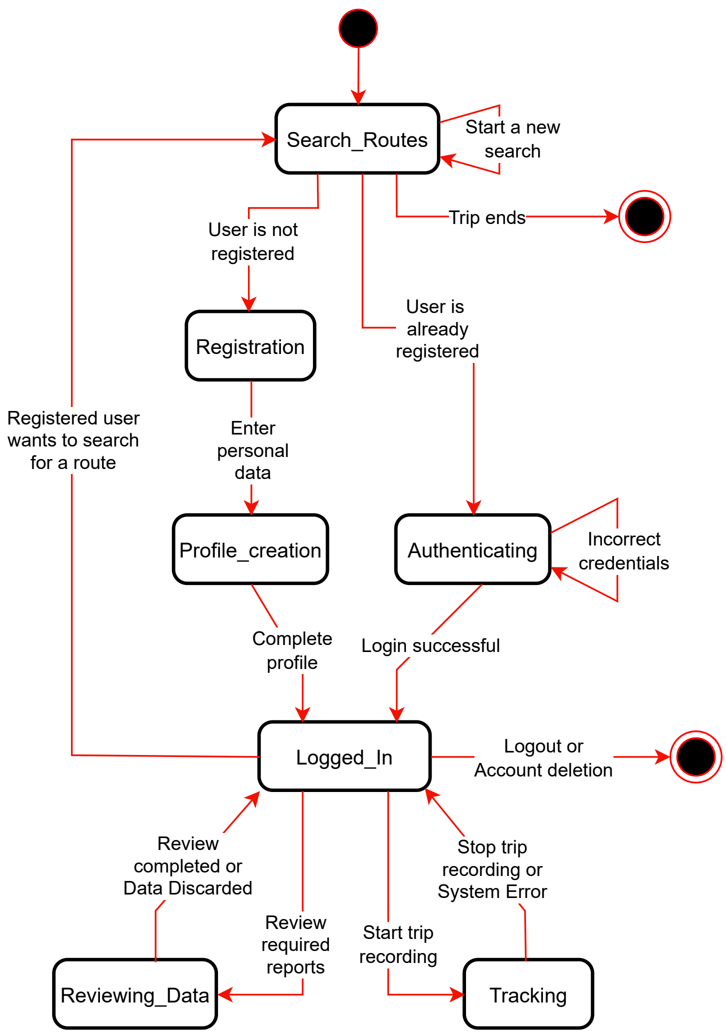

Figure 2: State diagram of a BBP system user's lifecycle

The finite state diagram in Figure 2 models the **user session lifecycle** within the BBP system, defining how the user transitions from the anonymous browsing state to the fully operational one. The system is designed to ensure that all basic functionality, such as route search and map viewing, is immediately accessible, with a single initial state that converges on `Search_Routes`, the universal entry point. From this anonymous browsing state, the user can choose to authenticate whether they are already logged in or not. Once the `Logged_In` state is reached, the user unlocks the contribution capabilities, which are critical to the system's value. This state serves as a hub, allowing the user to initiate trip tracking by moving to the `Tracking` state (when sensors are active) or to proceed to `Reviewing_Data`. Both contribution states are separated to reflect their high impact on resources (tracking) or data consistency (auditing). The session can end by exiting `Search_Routes` (for both anonymous and registered users) or by `Logout` from the operational state for the registered user.

Trip Lifecycle

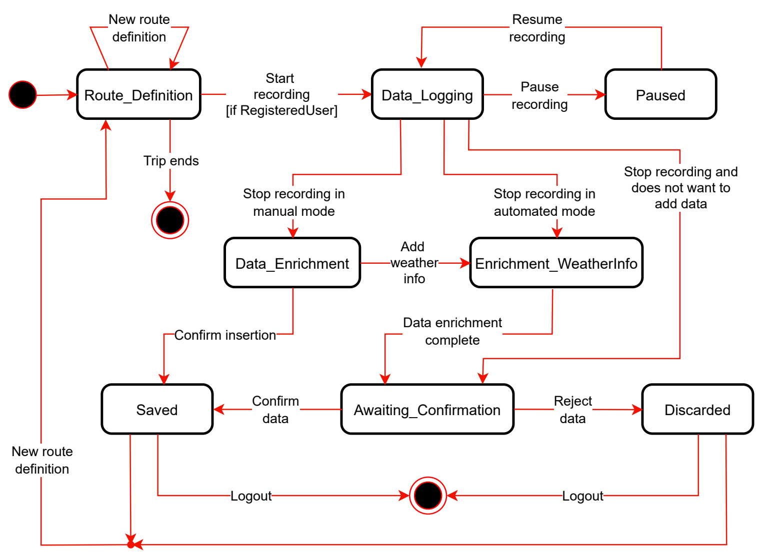

Figure 3: State Diagram of the Lifecycle of a Trip in the BBP System

The diagram in Figure 3 models the complete lifecycle of a `Trip`, from its inception to its final storage or discard. The process begins in the initial `Route_Definition` state, which represents the hub where a new route can be defined or an existing one can be used. The fundamental transition to data acquisition occurs only if the `[if RegisteredUser]` guard condition is satisfied, ensuring that only authenticated users can initiate tracking, based on the system's contribution requirements. Once in the `Data_Logging` state, the system actively logs raw sensor data (GPS, accelerometer) if in automatic mode. This state offers flexibility, allowing data acquisition to be paused and resumed via transitions. The system manages three distinct transitions when recording is stopped, resulting in separate processing paths:

- **Stop in manual mode**: This transition allows the user to actively add non-sensor data to the route.
- **Stop in automated mode**: Indicates that the route has ended, starting the automatic processing cycle.
- **Stop without data**: If the user does not wish to add any data, they go directly to the confirmation to save or delete the collected data (if collected).

The automated processing cycle begins with `Enrichment_WeatherInfo`, where the system enriches the trip with weather data retrieved from external services. Once enrichment is complete, the flow moves to `Awaiting_Confirmation`. This state is crucial for data quality: here, the user must decide whether to validate the anomalies detected by the sensors (e.g., potholes) or discard them. The cycle closes by returning to the `Route_Definition` state or definitively exiting the system, demonstrating how data only goes from ephemeral to persistent information through a rigorous validation process.

Data Lifecycle

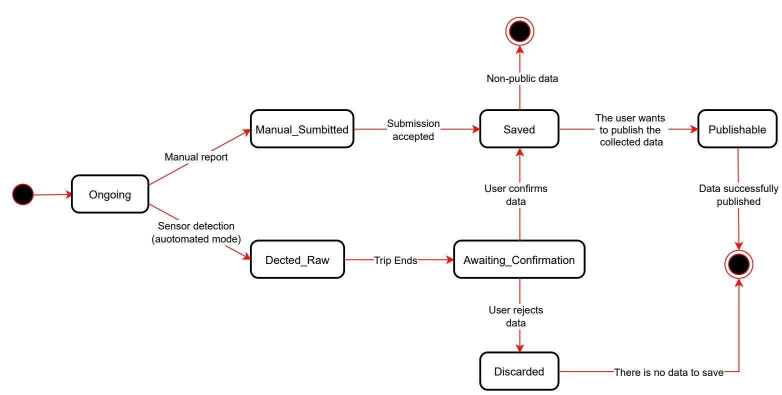

Figure 4: Data lifecycle state diagram in BBP system

The diagram in Figure 4 models the complete data lifecycle, from its origin to its final state. The process rigorously distinguishes data based on its source to direct it to the correct validation path. The flow forks immediately from the initial state:

- **Manual Path:** The user generates a `Manual report` that transitions to the `Manual_Submission` state. The data, being the result of an explicit action, is initially saved and can be published if the user wishes.

- **Automatic Path:** The data passively detected by the sensors transitions to the `Dected_Raw` state. This raw data must pass through the `Awaiting_Confirmation` state at the end of its journey.

The pending confirmation state is the critical checkpoint: the user is responsible for validating the discovery to allow it to move to `Publishable`, or discarding it, moving it to `Discarded`. Only data in the `Publishable` state is integrated and can influence the `Path Score`. The cycle ends with final publication or discard.

2.2 Product functions

Sign up & Login

This feature is the entry point for any user wishing to actively contribute to the inventory. A visitor can register by providing their information and credentials,and once the account is created the user can log in to access their reserved area, view their travel history, and use the tracking features. Without authentication, the user remains in "read-only" mode, without access or all the features expected of a registered user.

User Profile Management

Registered users have access to a dedicated section for managing their personal data. Here they can update their contact information and personal details, change their password, or delete their account. These actions ensure that the user maintains full control over their digital identity within the system.

Trip Recording

This is a core feature available exclusively to authenticated users. Users can start a recording session at the beginning of their activity. During the trip, the system tracks their geographic location via GPS in real time. Users have the flexibility to pause and resume recording (for example, during a rest stop). Upon completion, the trip is stored in the user's personal database.

Statistics Calculation and Data Enrichment

Upon completion of a trip, the system processes the raw data to provide detailed statistics, such as total distance traveled and average speed. Additionally, BBP automatically queries external services, if available, to retrieve weather information (temperature, wind, and weather conditions) for the area and time of the trip. This data is integrated into the trip record, providing the user with richer context for analyzing their performance.

Manual Data Entry

Registered users can actively contribute to the quality of the inventory by entering manual reports. Through a dedicated interface, users can specify the status of a road segment (e.g., "Optimal," "Requires Maintenance") or report the presence of specific obstacles. The system associates this information with the current GPS coordinates (or those selected on the map) and makes it immediately available to the community.

Automatic Detection via Sensors

If the registered user enables "Automatic Mode" while driving, the system uses the mobile device's accelerometer and gyroscope to monitor vibrations and sudden movements. Internal algorithms analyze this data to identify potential road surface anomalies, such as potholes. This process occurs in the background so as not to distract the user while driving.

Review and Confirmation of Detections

To ensure data reliability and filter out false positives, automatic detections are not published immediately. At the end of the journey, the system presents the user with a list of detected anomalies. The registered user must explicitly confirm the presence of the obstacle (validation) or discard the detection (if incorrect). Only confirmed data is promoted to publishable information. The published route data is then used to calculate the Path Score.

Route Search

This feature is accessible to all users, regardless of registration. The user enters a starting point and a destination in the search interface. The system processes the request and calculates one or more possible cycling routes connecting the two points, if data for such routes exists in the inventory.

Display and Path Score

The routes found are displayed on an interactive map. For each route, the system calculates and displays a `Path Score`. This summary score aggregates information about the route's status and the presence of confirmed obstacles, allowing the user to quickly assess not only the distance, but also the safety and quality of the proposed route.

2.3 User Characteristics

This section describes the general characteristics of users who interact with the BBP system. There are two main categories of users: Registered Users (the active contributors) and General Users (the passive users).

2.3.1 Registered Users

The Registered User represents the core of the BBP ecosystem. This profile typically corresponds to a regular cyclist (commuter or recreational) who wishes to monitor their performance and actively contribute to community safety.

Profile and Skills

The user must have a personal account with login credentials. It is assumed that they have moderate familiarity with the use of smartphones and GPS technology. Since the app is used in mobile contexts, the user requires a clear interface that minimizes distractions.

Needs and Interactions:

- **Tracking:** The user wants to track their trips to analyze statistics such as speed and distance, contextualized with weather data if available.
- **Active Contribution:** The user wants to report obstacles or assess road conditions to help other cyclists. They can do this manually or by activating automatic mode.
- **Validation:** The user is responsible for data quality. The system relies on them to confirm or discard automatic sensor detections (e.g., potholes) at the end of the trip, ensuring that only truthful information influences the Path Score.
- **Privacy:** The user wants sensitive data (such as personal travel history) to remain private, while agreeing to share anonymized road condition data publicly.
- **Trip planning**: The user needs to access to an updated archive of paths in order to plan its cycling activity; so it needs to find the most efficient or the more intriguing path from its starting point up to its destination, but avoiding those paths having some problem; it's also not interested in paths with low score, Since they won't match its expectancies.

2.3.2 Generic User

The Generic User includes anyone who accesses the platform without authenticating. This profile includes tourists, occasional cyclists, or route planners who need quick and reliable information without the commitment of registration.

Profile and Skills

They do not have a persistent profile in the system. Minimum proficiency in using digital maps and web/mobile interfaces is required. Interaction is sporadic and aimed at an immediate goal: reaching a destination.

Needs and Interactions:

- **Safety and Planning:** The primary need is to find the safest or most efficient route between two points. The user relies on the system to avoid poor or dangerous roads.
- **Immediacy:** They want to view routes and their Path Score immediately. It's not interested in contributing data or saving history, but only in consuming aggregated information generated by the community.
- **Reliability:** It expects the obstacle reports (e.g., potholes) displayed on the map to be up-to-date and verified, so it can plan its trip with confidence.

2.4 Assumptions, dependecies and constraints

2.4.1 Domain Assumptions

The following assumptions describe real-world conditions that the system considers true and necessary for the correct functioning of the intended features:

- **D1 - Hardware Equipment:** It is assumed that the user's mobile device is equipped with functioning and calibrated hardware, specifically: GPS receiver, accelerometer, and gyroscope.
- **D2 - Accuracy of user registration data:** It is assumed that the information entered by users during registration phase is correct and truthful.
- **D3 - Accuracy of route feedbacks:** It is assumed that the user's feedbacks about routes problems (either manual or automatically detected) are correct and truthful.
- **D4 - Accuracy of Basemaps:** It is assumed that third-party mapping services provide a correct topological representation of reality, that is if a road exists on the map then it's assumed that physically exists and that it'is drivable safely by bycicles (unless otherwise reported on BBP).
- **D5 - GPS Signal Availability:** It is assumed that, for most of the duration of an outdoor trip, satellite coverage is sufficient to ensure useful location accuracy.
- **D6 - Distinguishable Movement Patterns:** It is assumed that the physical characteristics of cycling are sufficiently distinct from those of other modes of transport or walking in order to allow classification algorithms to operate with an acceptable level of accuracy.

2.4.2 System Dependencies

The BBP system is not an island; it relies on external services to provide added value. Failure of these services degrades the system's functionality as follows:

- **External Weather Service:** BBP depends on third-party APIs to retrieve weather data (temperature, wind). If this service is unavailable, the system will continue to record trips, but the "Weather Enrichment" feature will not be performed, and the trips will be saved without this metadata.
- **Mapping Services:** Route visualization and address geocoding depend on external map providers. If these are unavailable, the "Route Search" and "Map View" features will be compromised.

2.4.3 System Constraints

- **GDPR and Privacy:** Since the system tracks users' physical movements (sensitive data), the management, storage, and sharing of GPS data must strictly comply with the GDPR regulation. Personal travel data must not be accessible to other users without explicit consent.
- **Energy Consumption:** The automatic detection algorithm must be optimized to avoid draining the mobile device's battery quickly, ensuring coverage of medium-duration trips (e.g., 2-3 hours).
- **Intermittent Connectivity:** Since cycling routes can pass through areas with poor network coverage, the mobile application must be able to store sensor data locally and synchronize it with the server as soon as the connection is re-established.

3 Specific requirements

3.1 External interface requirements

3.1.1 User interfaces

This section presents mockups of the BBP mobile application's user interface. The images illustrate the main interaction flows defined in the scenarios, demonstrating how the system meets usability and functionality requirements.

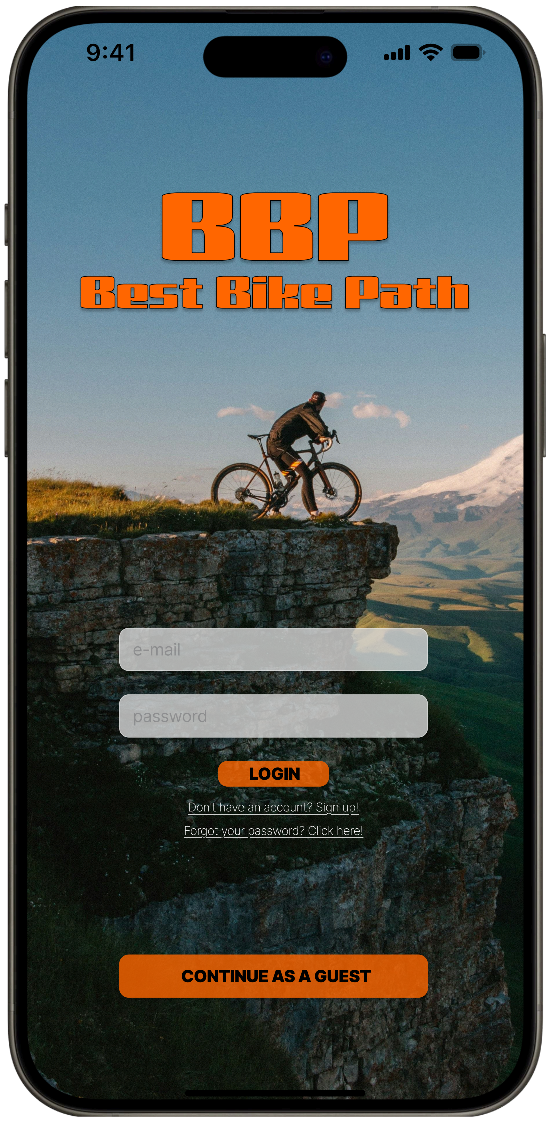

Figure 5: Login and Registration Screen

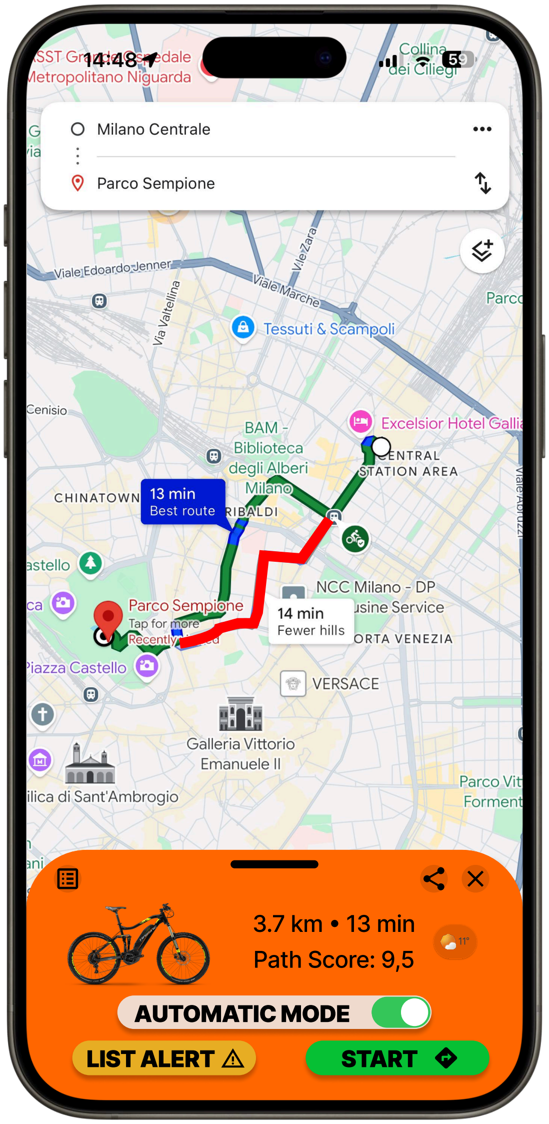

Figure 6: Route Selection Screen

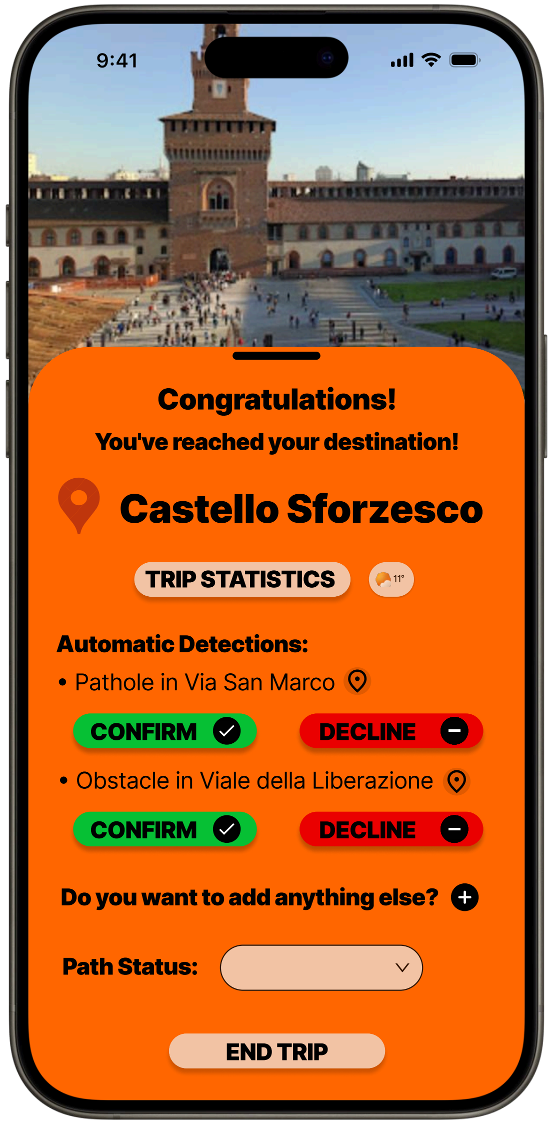

Figure 7: Post-Trip Confirmation Screen

Figure 8: Trip History Screen

3.1.2 Hardware interfaces

Since BBP is a mobile application focused on automatic tracking and detection, hardware interfaces are critical to the system's operation.

- **GPS:** The system requires access to the mobile device's GPS receiver to track the user's location in real time during travel and to geolocate alerts.

- **Inertial Sensors:** For the "Automatic Mode" feature, the application needs to interface directly with the device's motion sensors to detect vibrations and road surface anomalies.

3.1.3 Software interfaces

The system interacts with external software components to enhance its functionality.

- **External Weather Service API:** The system interfaces with a weather data provider to retrieve historical weather conditions for the time and location of the completed trip.
- **Mapping Service API:** The application uses mapping services for map rendering, route calculation, and address geocoding.
- **Mobile OS APIs:** The app interacts with native Android and iOS APIs for managing permissions and push notifications.

3.1.4 Communication interfaces

- **Network Protocols:** All communications between the mobile application and the backend server are via the **HTTPS** protocol to ensure the security and encryption of data in transit, especially for authentication information and sensitive location data.
- **Network Connectivity:** The device must have a network interface (4G/5G/Wi-Fi) to send data to the server and download maps.

3.2 Functional requirements

Authentication and Account Management

- [**R1**] The system shall allow any user to create an account.
- [**R2**] The system shall allow registered user to log in using their credentials.
- [**R3**] The system shall allow registered user to update their personal profile information.
- [**R4**] The system shall allow registered user to delete their account.

Trip Recording and Monitoring

- [**R5**] The system shall allow registered user to start the recording of a new trip.
- [**R6**] The system shall allow registered user to pause and resume the recording of an active trip.
- [**R7**] The system shall allow registered user to save a trip.

- **[R8]** During the recording, the system shall track the user's position and its performance statistics.
- **[R9]** Upon completion of a trip, the system shall automatically retrieve weather data from an external service, if available, and associate it with the saved trip.

Data Contribution and Governance

- **[R10]** The system shall allow registered user to insert manual reports regarding the status of a path.
- **[R11]** The system shall allow registered user to submit feedback regarding the Path Status.
- **[R12]** The system shall allow registered user to insert manual reports regarding problems on the path.
- **[R13]** The system shall allow registered user to enable automatic detection for a trip.
- **[R14]** When automatic detection is active, the system shall analyze data from the device's sensors to detect potential anomalies.
- **[R15]** The system shall present the list of automatically detected anomalies to the registered user at the end of the recorded trip for review.
- **[R16]** The system shall allow the registered user to confirm or discard a detected anomaly.

Path Planning and Visualization

- **[R17]** The system shall compute valid cycling routes between a specified starting point and a destination based on the available physical road network.
- **[R18]** The system shall compute and visualize one or more valid routes between the specified points on a map.
- **[R19]** The system shall compute a Path Score for each route derived from available inventory data
- **[R20]** The system shall display confirmed obstacles on the map with visual markers.
- **[R21]** The system shall allow the user to filter the search on Path properties.

Trip History

- **[R22]** The system shall allow registered user to view the list of its past trips.
- **[R23]** The system shall allow registered user to view the details of a specific past trip, including the route on the map, statistics, and weather data (if they exist).
- **[R24]** The system shall allow registered users to delete a specific trip from their history.
- **[R25]** The system shall allow the user to search a specific trip in its history.
- **[R26]** The system shall allow the user to filter the view of its history.

3.2.1 Use Case Diagram

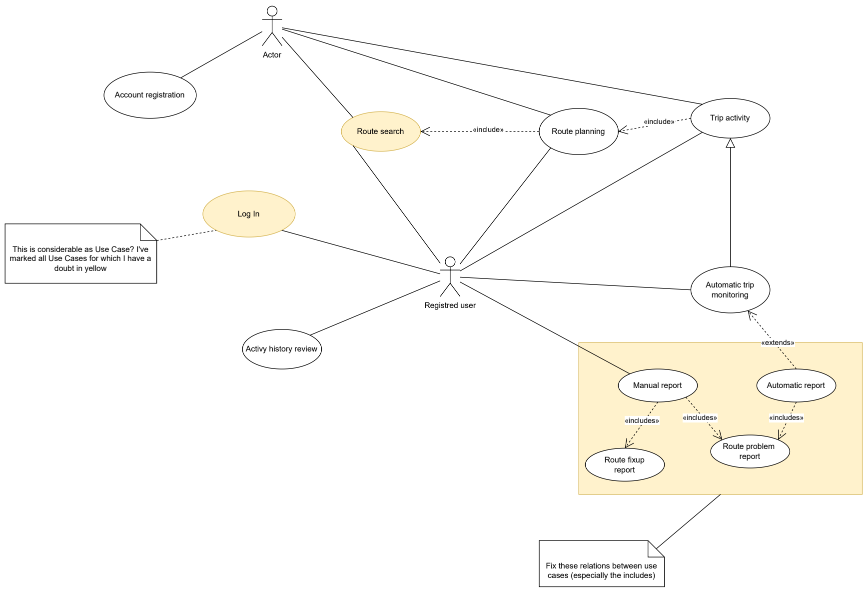

Figure 9: Use Case Diagram

3.2.2 Use cases

[UC1] Account creation

Name	**Account creation**
Actors	User
Entry Condition	True
Event Flow	1. The person downloads the BBP, opens it and starts the creation of an account 2. The system asks to the user to fill out a form with the following personal information: name, surname, birth date, geneder, email, password; the systems also asks to accept the privacy terms 3. The person fills out the questionare and submits it 4. The system verifies that the email address is valid and that it's not already been used, then it sends a verification email containing a link to verify the email address, which expires in 1 hour 5. The person receives the email and opens the link to confirm the email 6. The system sends an acknowledgement of successful account creation
Exit Condition	The account is successfully created
Exception	• Email address is not valid, therefore a warnigs is displayed in point *2.* and the form can't be submitted. • Email inserted during registration has been already used, therefore the account is not created and in the point *4.* instead of a link an informative message is sent. • The user doesn't open the confirmation link within an hour, therefore the account creation procedure is aborted and the submited data cancelled.

Table 1: Refers to SC1. We highlight that the same person can create multiple accounts.

[UC2] User login

Name	**User login**
Actors	Registered user
Entry Condition	The user isn't logged in
Event Flow	1. The user opens the BBP app and the login form is shown to it 2. The user types email and password, then submits the form 3. The system receives the user's login info, checks that the information sent corresponds to an existing account 4. The system retrives the information to build the homepage for that account and sends it
Exit Condition	The user logs into its account
Exception	• The user sends an email not related to any account, therefore the system sends an error message to the user • The user sends a password that doesn't match with the one on the system for that account, therefore the system sends an errore message to the user

Table 2: Refers to SC2.

[UC3] Account information update

Name	**Account information update**
Actors	Registered user
Entry Condition	True
Event Flow	1. The user opens his profile page and selects the option to modify the account 2. The user selects the attribute to modify, types the new value and sends it 3. The system modifies the attributes and sends an acknowledgment
Exit Condition	The account profile is updated
Exception	1. The user tries to modify email address and password, therefore it's immediately blocked

Table 3: Refers to SC3

[UC4] Password account reset

Name	**Password account reset**
Actors	Registered user
Entry Condition	The user can't log into its account
Event Flow	1. The user clicks the "Reset Password" option in the login page and sends a reset password request 2. The system receives the request and sends a form in which the user must specify the account email 3. The user receives the form, enters the account email and submits it 4. The system receives the account email, checks whether an account with that email exists or not, and if it finds it sends an email to that address with a link to reset the password that expires in 1 hour 5. The user receives the email, opens the link, types the new password two times and sends the new password 6. The system receives the new password, updates the account password and sends an update confirmation 7. The user receives the update confirmation
Exit Condition	The user can log-in with the new password
Exception	• The user submits an invalid email address, therefore an error message would be sent to it • The user submits the email address of another account, therefore he won't receive the email to reset the password • The user doesn't open the link within one hour, therefore the password wouldn't be changed • The user doesn't submits the new password after opening the link within one hour, therefore the password wouldn't be changed

Table 4: Refers to SC4.

[UC5] Account deletion

Name	**Account deletion**
Actors	Registered user
Entry Condition	True
Event Flow	1. The user opens the Profile page, opens the options and selects to delete the account 2. The system receives the request and sends a confirmation message on the app 3. The user sends the confirmation 4. The system receives the confirmation and sends an email containing a link to delete the account, which expires in 1 hour 5. The user receives the email, opens the link and sends the second confirm. 6. The system receives the second confirmation and processes the account deletion request 7. Once the system has deleted the account, sends to the ex-account mail address a deletion message confirmation
Exit Condition	The account is deleted
Exception	• The user didn't open the link after one hour, therefore the link expries • The user didn't confirm the second time, therefore after one hour the request expires

Table 5: Refers to SC5

[UC6] Route planning with route match

<table>
<tr><td>Name</td><td>Route planning with route match</td></tr>
<tr><td>Actors</td><td>User</td></tr>
<tr><td>Entry Condition</td><td>True</td></tr>
<tr><td>Event Flow</td><td>The user opens the Search page and inserts the starting point and the destination
The system retrieves from its archive all paths near the starting point specified by the user and that lead toward the destination, ordered by ”Path Score” and sends them to the user
The user explores the choices given by the system and selects one of them
The user starts the trip activity for the selected path, see UC8, UC:9.</td></tr>
<tr><td>Exit Condition</td><td>A path is displayed to the user</td></tr>
<tr><td>Exception</td><td>No path between starting point and destination is found, see UC7</td></tr>
</table>

Table 6: Refers to SC6, SC8

[UC7] Route planning without route match

<table>
<tr><td>Name</td><td>Route planning without route match</td></tr>
<tr><td>Actors</td><td>User, External Mapping Service</td></tr>
<tr><td>Entry Condition</td><td>True</td></tr>
<tr><td>Event Flow</td><td>The user opens the Search page and inserts the starting point and the destination
The system fails to retrieve from its archive any path near the starting point specified by the user and that lead toward the destination
The system sends to the External Mapping Service a request to compute a path between the starting point and the ending point.
The External Mapping Service computes the path and returns it to the system
The system forwards the computed paths to the user
The user searches among returned paths, selects one of them and starts the trip activity for the selected path</td></tr>
<tr><td>Exit Condition</td><td>A path is displayed to the user</td></tr>
<tr><td>Exception</td><td></td></tr>
</table>

Table 7: Refers to SC7, SC9

[UC8] Trip activity for unregistered user

Name	**Trip activity**
Actors	Unregistered user
Entry Condition	True
Event Flow	1. The user selects the path and starts the trip, sending an information retrieval request to the system 2. The system receives the request, retrieves the path map and sends it to the user 3. The user receives the path map The user might stop and resume the trip whenever he likes 4. The user completes the trip and closes it
Exit Condition	The user completed its trip
Exception	

Table 8: Refers to SC10, SC11

[UC9] Trip activity for registered user

Name	**Trip activity**
Actors	Registered user
Entry Condition	True
Event Flow	1. The user selects the path and starts the activity, sending an information retrieval request to the system 2. The system receives the request, retrieves the path map and sends it to the user 3. The user receives the path map The user might stop and resume the activity whenever he likes 4. The user completes the activity, and sends an "Activity Completion message" to the system 5. The system receives the notification about the activity completion and sends a form to rate the path 6. The user receives the rating form, fills it out and sends it 7. The system receives the rating form and sends an acknowledgement
Exit Condition	The user completed its trip activity and scored to the path
Exception	

Table 9: Refers to SC10, SC11, SC13

[UC10] Automatic trip monitoring

Name	**Automatic trip monitoring**
Actors	Registered user, Weather Service
Entry Condition	True
Event Flow	1. The user selects the route he wants to ride on, selects the option to automatically collect data and starts the activity 2. The user's personal device samples user position every second and stores it 3. Once finished the user stops the activity and the data collected are sent to the system 4. The system calculates some metrics about the user performances (total distance traveled, average speed, maximum speed, minumum speed, maximum altitude excursion, average altitude excursion) 5. The system sends a request to retirve weather conditions during the activty to a third-part Weather Service 6. The Weather Service responds to the system with the requested data 7. The system send to the user the computed informations mentioned in *4.* and *5.*, plus the map showing the path traveled
Exit Condition	The user completed the activity with related metrics
Exception	• Weather conditions can't be retrived from the third party service, therefore only the metrics about user performances are shown to the user • User's device loses GPS signal during the activity, therefore the system is not able to compute all the metrics about user performances and shows only the ones that can be computed with the available data, notifying the user about the partial (or total) data loss

Table 10: Refers to SC12

[UC11] Manual route problem report

Name	**Manual route problem report**
Actors	Registered user
Entry Condition	Route problem hasn't been reported
Event Flow	1. The user after noticing a problem along a route opens the Report Issue page 2. The user searches for the route and selects it, then specifies the issue type, the issue position along the route and adds a description, then submits the report 3. The system receives the issue report, updates the information about that problem and send an acknowledgement
Exit Condition	Path problem has been reported
Exception	

Table 11: Refers to SC15

[UC12] Route problem fixup report

Name	**Route problem fixup report**
Actors	Registered user
Entry Condition	Route fixup hasn't been reported
Event Flow	1. The user opens the app, selects the problem icon on the route and marks it as fixed 2. The system receives the fixup report, updates the information about that problem and sends an acknowledgement
Exit Condition	Route problem has been reported
Exception	

Table 12: Refers to SC16

[UC13] Automatic route problem detection and report

Name	**Automatic route problem detection and report**
Actors	Registered user
Entry Condition	True
Event Flow	1. The user starts an activity with automatic issue detection enabled 2. The BBP app collects data from the user's device sensors and analyzes them in real-time to detect potential issues along the route 3. When the user finishes the activity, the BBP app shows to the user a list of all problems detected and their location, asking the user confirmation for each one of them 4. The user confirmes whether the problems detected are real issues or false positives 5. The BPP app sends to the system the confirmed issues 6. The system sends and acknowledgement, and updates the path status
Exit Condition	Path problem has been detected and reported
Exception	

Table 13: Refers to SC14

[UC14] User's activity history consultation

Name	**User's activity history consultation**
Actors	Registered user
Entry Condition	True
Event Flow	1. The user opens the relative page on the app, optionally applies filters, and sends the request to the system 2. The system runs the query and retrieves the activities matching the request, then sends the result to the user
Exit Condition	The user consults its activity history
Exception	

Table 14: Refers to SC17

[UC15] User activity deletion

Name	**User activity deletion**
Actors	Registered user
Entry Condition	User's activity history contains N activities
Event Flow	1. The user opens the Acivity History and searches for the trip to delete 2. The user selects the trip, selects the option to delete it and confirms the deletion. 3. The system recevies the deletion request, deletes the activity and sends an acknowledgement to the user.
Exit Condition	User's activity history contains $N - 1$ activities
Exception	

Table 15: Refers to SC18

3.2.3 Requirement Mapping

This section maps the Goals identified in Section 1 to the Functional Requirements and Domain Assumptions. This mapping demonstrates that the set of requirements, supported by the assumptions, is sufficient to satisfy the system goals ($R \wedge D \models G$).

G1: A registered user wants to track their personal cycling activities and related performance statistics.	
Requirements	**Domain Assumptions**
[**R1**] The system shall allow any user to create an account. [**R2**] The system shall allow registered user to log in using their credentials. [**R5**] The system shall allow registered user to start the recording of a new trip. [**R6**] The system shall allow registered user to pause and resume the recording of an active trip. [**R7**] The system shall allow registered user to save a trip. [**R8**] During the recording, the system shall track the user's position and its performance statistics. [**R9**] Upon completion of a trip, the system shall automatically retrieve weather data from an external service, if available, and associate it with the saved trip. [**R22**] The system shall allow registered user to view the list of its past trips. [**R23**] The system shall allow registered user to view the details of a specific past trip, including the route on the map, statistics, and weather data (if they exist). [**R24**] The system shall allow registered users to delete a specific trip from their history. [**R25**] The system shall allow the user to search a specific trip in its history. [**R26**] The system shall allow the user to filter the view of its history.	**D1 - Hardware Equipment:** It is assumed that the user's mobile device is equipped with functioning and calibrated hardware, specifically: GPS receiver, accelerometer, and gyroscope. **D5 - GPS Signal Availability:** It is assumed that, for most of the duration of an outdoor trip, satellite coverage is sufficient to ensure useful location accuracy.

Table 16: Requirement Mapping for Goal G1

G2: A registered user wants to contribute to the community inventory by sharing reliable information on the condition of the trails (e.g. quality, obstacles, potholes).	
Requirements	**Domain Assumptions**
[**R2**] The system shall allow registered user to log in using their credentials. [**R10**] The system shall allow registered user to insert manual reports regarding the status of a path. [**R11**] The system shall allow registered user to insert a personal rating of a path. [**R12**] The system shall allow registered user to insert manual reports regarding problems on the path. [**R13**] The system shall allow registered user to enable automatic detection for a trip. [**R14**] When automatic detection is active, the system shall analyze data from the device's sensors to detect potential anomalies. [**R15**] The system shall present the list of automatically detected anomalies to the registered user at the end of the recorded trip for review. [**R16**] The system shall allow the registered user to confirm or discard a detected anomaly.	**D1 - Hardware Equipment:** It is assumed that the user's mobile device is equipped with functioning and calibrated hardware, specifically: GPS receiver, accelerometer, and gyroscope. **D3 - Accuracy of route feedbacks:** It is assumed that the user's feedbacks about routes problems (either manual or automatically detected) are correct and truthful. **D6 - Distinguishable Movement Patterns:** It is assumed that the physical characteristics of cycling are sufficiently distinct from those of other modes of transport or walking in order to allow classification algorithms to operate with an acceptable level of accuracy.

Table 17: Requirement Mapping for Goal G2

G3: Any user (registered or not) wants to find and view the best cycling route between an origin and a destination, based on up-to-date and relevant data.	
Requirements	**Domain Assumptions**
[**R17**] The system shall allow any user to search for cycling paths between starting point and a destination. [**R18**] The system shall compute and visualize one or more valid routes between the specified points on a map. [**R19**] The system shall calculate a Path Score for each route. [**R20**] The system shall display confirmed obstacles on the map with visual markers. [**R21**] The system shall allow the user to filter the search on Path properties.	**D4 - Accuracy of Basemaps:** It is assumed that third-party mapping services provide a correct topological representation of reality, that is if a road exists on the map then it's assumed that physically exists and that it'is drivable safely by bycicles (unless otherwise reported on BBP). **D1 - Hardware Equipment:** It is assumed that the user's mobile device is equipped with functioning and calibrated hardware.

Table 18: Requirement Mapping for Goal G3

G4: The cycling association aims to provide the community with a tool to create, consult, and maintain a reliable and centralized inventory of cycling routes.	
Requirements	**Domain Assumptions**
[**R1**] The system shall allow any user to create an account. [**R3**] The system shall allow registered user to update their personal profile information. [**R4**] The system shall allow registered user to delete their account. [**R16**] The system shall allow the registered user to confirm or discard a detected anomaly.	**D2 - Accuracy of user registration data:** It is assumed that the information entered by users during registration phase is correct and truthful. **D3 - Accuracy of route feedbacks:** It is assumed that the user's feedbacks about routes problems (either manual or automatically detected) are correct and truthful.

Table 19: Requirement Mapping for Goal G4

3.3 Performance requirements

Given the nature of BBP as a mobile application that also operates in active mobility contexts, performance is critical not only for the user experience, but also for the security and reliability of the collected data.

- **Interface Responsiveness:** The system must ensure immediate response times for critical interactions during cycling, with a latency of less than 200 ms, to avoid dangerous distractions for the user.
- **Real-Time Data Processing:** During "Automatic Detection" mode, the local algorithm on the device must process sensor data in real time without causing interface crashes or delays in recording the GPS track.
- **Routing:** The route search functionality must return results, complete with *Path Score*, within 3 seconds for requests in a standard urban environment (10 km radius), ensuring smooth planning.
- **Backend Scalability:** The system must be able to handle simultaneous load peaks (e.g., weekends or cycling events), scaling horizontally to support thousands of simultaneous trip uploads without data loss.
- **Reliability:** The backend service must guarantee 99.9% uptime on a monthly basis, ensuring that users can always sync their trips and access maps.

3.4 Design Constraints

3.4.1 Standards Compliance

The BBP system adheres to rigorous international standards to ensure interoperability, security, and regulatory compliance.

Standard	Description
GDPR (EU 2016/679)	The system manages sensitive geolocation and user profiling data. All processing must comply with the General Data Protection Regulation, guaranteeing the right to be forgotten and data minimization.
WGS 84	Geodetic reference standard for the GPS system. All stored and exchanged coordinates must comply with this standard to ensure compatibility with global maps.
GPX (GPS Exchange)	The system should support the export of trip data in the standard XML format for GPS data, facilitating interoperability with other sports platforms.
ISO/IEC 27001	Standard for information security management, applied to protect the backend infrastructure and user data from unauthorized access.

Table 20: Compliance standards adopted by BBP

3.4.2 Hardware Limitations

The mobile application must operate in a resource-constrained environment, typical of mobile devices during extended outdoor use.

- **Power Consumption:** The automatic detection algorithm (GPS + Sensors) must be optimized to consume no more than 10-15% battery power per hour of use on an average device, ensuring the user does not run out of battery power while traveling.
- **Required Sensors:** Full use of the app is contingent on the physical presence of a calibrated accelerometer and gyroscope. Older or low-end devices without these sensors will only be able to use the app in limited mode (without automatic detection).
- **Intermittent Connectivity:** The design must include an "offline-first" mode for data recording. Upload to the server must occur asynchronously when the connection is stable, handling any timeouts without losing local data.

3.4.3 Any Other Constraint

- **GPS Accuracy:** The accuracy of obstacle detection is limited by the accuracy of the device's civilian GPS. The system must include clustering or manual correction mechanisms to handle inherent hardware inaccuracy.
- **Operating System:** The application must be compatible with Android and iOS versions released in the last 3 years to ensure access to the latest APIs for efficient background sensor management.
- **Local Data Processing:** To ensure responsiveness and minimize mobile data usage, the raw processing of high-frequency sensor data must be performed locally on the user's device. The system is constrained to transmit only the identified "candidate anomalies" to the server, rather than the continuous raw data stream.

3.5 Software system attributes

3.5.1 Reliability

The system must provide robust, scalable storage capable of handling high-volume data ingestion generated by frequent user activity sampling. To ensure data integrity and fault tolerance, replication strategies must be implemented across multiple nodes to prevent data loss. The system should employ an eventually consistent data model across distributed nodes for most data types. However, a differentiated consistency approach could be used: strong consistency must be enforced for sensitive user data to guarantee correctness, while relaxed consistency models are acceptable for non-critical data such as path status updates and user monitoring activities.

3.5.2 Availability

The system must maintain high availability with minimal downtime to ensure continuous service delivery. Core functionalities requiring guaranteed uptime include path planning, activity recording, and real-time user guidance—these services are mission-critical and should target 99.9% availability. To accommodate fluctuating demand patterns, with expected peak traffic periods—such as holidays, weekends, and seasonal recreational periods, urge the need for auto-scaling capabilities to dynamically adjust computational resources and maintain performance under variable load conditions. Since data about path and user have a strict correlation with geographical location,the system should implement a geo-distributed architecture with regional data partitionin, minimiziong latency throught data locality.

3.5.3 Security

The system processes sensitive user credentials, including personal email addresses and passwords, necessitating robust security mechanisms. To ensure data confidentiality, the system must implement end-to-end encryption for stored data, while all client-server communications must be secured via security protocols.

3.5.4 Maintainability

The system architecture must prioritize modularity and loose coupling between components to enable independent updates and minimize maintenance overhead. The development process must follow a modular design approach from inception, ensuring clear interface definitions, dependency management, facilitate isolated testing and enable parallel development workflows.

3.5.5 Portability

The system must achieve cross-platform compatibility across a diverse range of devices and operating systems. To meet this requirement, the technology stack should prioritize platform-independent frameworks and languages that support multiple execution environments without significant code modifications. Therefore, the core system logic must be abstracted from platform-specific dependencies to facilitate seamless deployment across various user devices.

4 Formal anlaysis using Alloy

5 Effort spent

Guglielmi Leonardo

- 11/11/2025 1h (RASD document structure)
- 19/11/2025 1h 30m (Section 1 review)
- 20/11/2025 1h (Scenarios review)
- 21/11/2025 1h (class diagram review)
- 21/11//2025 1h (Section)
- 24/11/2025 2h (Scenario update)
- 26/11/2025 2h (Use Cases)
- 27/11/2025 1h (System Attributes)
- 29/11/2025 1h (Use Cases improvement)
- 03/12/2025 2h (Scenarios update)
- 04/12/2025 2h (Use case update)

Lo Conte Francesco

- 17/11/2025 4h (Completing Section 1 (Introduction))
- 18/11/2025 7h (Section 2 - StateDiagrams, DomainClassDiagram, Scenarios)
- 19/11/2025 1h (Completing Section 2)
- 25/11/2025 5h (User Interfaces and some subsections of section 3)
- 26/11/2025 30 min (Revision)
- 30/11/2025 3h (Various Adjustements)

6 References