

# NetQuest

## PHASE 1

### Work Methodology

We follow Scrum as our methodology. Stand-ups are held once a week on Thursday to review progress, but communication remains active through Discord and WhatsApp throughout the week for any necessary updates or discussions.

### Sprint Events

#### *Sprint Retrospective*

Held at the end of each sprint to reflect on what went well and identify areas for improvement.

#### *Sprint Demo*

After each sprint, a demo is presented to showcase the completed features to stakeholders.

### Sprint Backlog Refinement

We will conduct sprint backlog refinement every week to prepare the tickets for the next sprint. All tickets in the upcoming sprint should be "Ready" to work on before the sprint begins.

### Ticket Workflow

#### *In Progress*

When a developer starts working on a ticket, it moves to the "In Progress" queue. The developer is responsible for implementing both the front-end and back-end, along with writing at least one unit test for the functionality.

#### *In Review*

Once development is complete, the ticket moves to the "In Review" queue. Before starting a new ticket, developers are expected to prioritize reviewing existing tickets.

During a review, the focus should be on:

- Code cleanliness
- Alignment with the requested functionality
- Adequacy of unit tests

If changes are needed, the ticket is reassigned to the original developer and returns to the "In Progress" queue. If no comments are made, the reviewer approves the ticket, triggering the automated pipeline process.

#### *In Deployment*

The code passes through the CI/CD pipeline. If all steps succeed, the ticket moves to the "In Testing" queue.

#### *In Testing*

QA testers verify the functionality by manually executing test cases. They write a comprehensive test report in a Word document, documenting both successes and any failures.

#### *In Acceptance*

If all test cases pass, the ticket is moved to the "In Acceptance" queue. If issues are found, it returns to "In Progress," assigned to the original developer for fixes.

The **Product Owner** reviews tickets in the "In Acceptance" queue to ensure that all project criteria are met before marking the ticket as done.

### Collaboration and Communication

We encourage open communication across the team, especially when blockers arise. If a developer encounters a blocker, they can request assistance through Discord or WhatsApp. We also practice pair programming for complex tasks, which promotes collaboration and knowledge sharing within the team. Informal check-ins ensure that any issues are quickly resolved before they affect progress.

### Testing Strategy

Our testing strategy involves a combination of automated and manual tests to ensure high-quality code. Every feature must include:

#### *Unit Tests*

Developers are required to write unit tests for all new functionality to verify isolated code behavior.

#### *End-to-End Tests*

Manual testing is handled by the QA team, who follow test case scenarios to validate the feature before moving it to the "Acceptance" queue. Testing is crucial to maintaining a reliable codebase as development progresses.

## Continuous Integration / Continuous Deployment (CI/CD)

We use a CI/CD pipeline to automate building, testing, sonar cloud, dependency check and deploying our code. When code is merged into the main branch, it passes through a series of automated tests to ensure quality.

After passing, it is automatically deployed to the **Staging** environment for manual testing by the QA team. Once all tests are successfully completed and the functionality is validated, the feature is ready to be moved to production.

Our CI/CD process ensures:

- Automated Testing
- Staging Environment
- Production Deployment

### *Automated Testing*

All merged codes are tested automatically through units. If any test fails, the pipeline halts, and the issue must be fixed before moving forward.

### *Staging Environment*

Features are deployed to the staging environment for manual testing by QA to simulate the production environment and catch any potential issues.

### *Production Deployment*

After passing all tests and manual validations, the feature is deployed to the production environment, ensuring a smooth and reliable release.

This CI/CD approach helps maintain high code quality, minimize deployment risks, and allow for faster iterations while ensuring that features are fully tested before reaching users.

## Definition of Ready (DoR)

- The user story is clearly defined and understood by the team.
- Acceptance criteria are specified.
- All dependencies are identified and addressed.
- Any necessary designs or prototypes are available.
- The story is small enough to be completed within a sprint (well-groomed and estimated).
- The team has enough information to start and finish the development.
- Stakeholders have approved the user story if necessary.

- Test cases are done

## Definition of Done (DoD)

- Code is written, tested, and peer reviewed.
- Unit tests are written and passed (at least 1 per ticket).
- The code is tested by QA on the staging environment.
- The feature passes all acceptance criteria.
- Everything works without errors, and it is deployed to the production environment.

## Roles

### Product Manager (PM)

#### Responsibilities:

- **Vision & Strategy:** Define the product vision, goals, and overall strategy.
- **Roadmap Planning:** Develop and manage the product roadmap, prioritizing features and releases.
- **Stakeholder Management:** Act as the main point of contact for stakeholders, including partners, investors, and users.
- **Feature Definition:** Work closely with the Business Analyst (BA) and developers to define features and ensure alignment with user needs.
- **Success Metrics:** Set and track KPIs for the product, such as user acquisition, retention, and engagement.

**Key Job:** Ensure that the app delivers value to users and meets business goals.

### Business Analyst (BA)

#### Responsibilities:

- **Requirements Gathering:** Collect and document functional and non-functional requirements for the app, working closely with users, stakeholders, and the PM.
- **User Stories & Acceptance Criteria:** Create detailed user stories with clear acceptance criteria for developers and testers to follow.
- **Process Mapping:** Identify and document workflows, business rules, and user journeys.

- **Gap Analysis:** Analyse any gaps between the current state of the app and the desired future state.
- **Competitive Research:** Analyse similar apps in the market to identify opportunities or potential features.

**Key Job:** Bridge the gap between business objectives and technical development, ensuring clarity in requirements.

## UX/UI Designer

### Responsibilities:

- **User Research:** Conduct user research, interviews, and testing to understand the needs of target personas.
- **Wireframes & Prototypes:** Create wireframes, mock-ups, and interactive prototypes to guide development.
- **Design System:** Develop a consistent design language and components for the app, ensuring a clean and user-friendly interface.
- **User Testing:** Test the app's usability and design with real users, iterating based on feedback.
- **Accessibility & Responsiveness:** Ensure the design is accessible to all users, including those with disabilities, and optimized for mobile devices.

**Key Job:** Design intuitive and visually appealing interfaces that cater to users' needs.

## Developers (Frontend, Backend)

### Responsibilities:

- **Frontend Developers:**
  - Build the user interface based on designs provided by the UX/UI team.
  - Implement features such as the interactive map, filters, and search functionality.
  - Ensure responsiveness and optimization for various devices.
- **Backend Developers:**
  - Build and maintain the server, database, and APIs that handle user accounts, location data, point systems, and AI-driven features.

- Implement security measures to protect user data and handle Wi-Fi spot verification.

**Key Job:** Write code to bring the app to life, ensuring functionality, security, and scalability.

## AI Specialist

### Responsibilities:

- **Recommendation Engine:** Develop algorithms to suggest locations based on user preferences, past behaviour, or popular spots.
- **Data Analysis & Insights:** Analyse user data to optimize suggestions, features, and app performance.
- **Crowd Density Estimation:** Create models that estimate crowd sizes or foot traffic at locations based on available data.

**Key Job:** Design and implement AI-driven features that enhance the user experience, such as personalized suggestions and weather-based filtering.

## Quality Assurance (QA) / Testers

### Responsibilities:

- **Test Planning:** Develop test plans and strategies, defining different types of tests (unit, integration, system, and user acceptance tests).
- **Automated Testing:** Implement automated tests for core features like map functionality, Wi-Fi spot verification, point system, and AI-driven suggestions.
- **Manual Testing:** Conduct manual testing of the app across various devices and environments to identify bugs or usability issues.
- **Performance Testing:** Test the app's performance under different conditions (e.g., high traffic or offline mode).
- **Bug Tracking & Reporting:** Log issues in the bug tracking system and ensure that bugs are resolved before release.

**Key Job:** Ensure the app is free of bugs, performs well, and provides a smooth user experience.

## DevOps Engineer

### Responsibilities:

- **CI/CD Pipelines:** Set up continuous integration and continuous deployment pipelines to automate testing and deployment.

## Marketing Manager

### Responsibilities:

- **User Acquisition Campaigns:** Plan and execute marketing strategies to acquire new users, focusing on digital ads, partnerships, and local campaigns.
- **Partnership Development:** Build relationships with local businesses, cafes, coworking spaces, and municipal governments to enhance the app's visibility and utility.
- **Content Creation:** Develop content (blogs, social media, email campaigns) to promote new features, updates, or user stories.
- **User Retention:** Plan campaigns that encourage existing users to stay active, such as challenges or reward incentives.
- **Data-Driven Marketing:** Analyze app usage data to target campaigns at the right audience, optimizing for growth.

**Key Job:** Drive user acquisition, retention, and growth, ensuring the app reaches the right audience.

## Customer Support

### Responsibilities:

- **User Assistance:** Provide support for users experiencing technical difficulties, or who need help navigating the app.
- **Feedback Loop:** Collect and relay user feedback to the PM and development team for continuous improvement.
- **Community Management:** Manage and engage with the app's user community through social media, forums, or in-app support features.

**Key Job:** Ensure users are supported and their concerns are addressed, creating a positive user experience.

## Collaboration Overview

- **PM and BA** work closely to define requirements and features.
- **Designers** collaborate with **PM** and **UX/UI** to create user-centred designs.
- **Developers** work with **QA** to ensure a smooth development process with frequent testing.
- **DevOps** ensures the app runs smoothly in production.
- **Marketing** and **Customer Support** ensure the app is well-received and users are engaged.

This team structure ensures all aspects of the project, from development to user experience and growth, are well-handled.

## PHASE 2

### Research on the Project Domain: Smart Cities and Public Wi-Fi

Smart cities leverage digital technologies to improve urban life, with a focus on sustainability, efficiency, and enhanced services for citizens. Public Wi-Fi is an integral part of this initiative, promoting connectivity and digital inclusion by providing free or low-cost internet access in public spaces.

#### Key Areas of Research:

- **Urban Connectivity Initiatives:**
  - Many cities around the world have started offering public Wi-Fi as a part of their smart city plans. Examples include New York's **LinkNYC**, Barcelona's city-wide public Wi-Fi network, and Tallinn, Estonia's free Wi-Fi in almost all public spaces.
  - **Key Insight:** These projects aim to bridge the digital divide, making sure everyone has access to internet services regardless of their location or economic status.
- **Benefits of Public Wi-Fi:**
  - Enhances tourism by providing internet access for tourists to navigate cities and share experiences.
  - Increases productivity by allowing remote workers, students, and residents to access Wi-Fi in public spaces like parks and cafes.



- **Key Insight:** Public Wi-Fi also drives local business growth by bringing more people to physical locations, especially cafes and coworking spaces.

## Research on Related Works

### Existing Apps and Services:

- **Wi-Fi Map:** A crowdsourced Wi-Fi locator app where users can find and share free Wi-Fi hotspots globally. It offers a large database of Wi-Fi networks with user-generated passwords and reviews.
  - **Strengths:** Extensive network coverage allows users to add hotspots and includes a built-in map.
  - **Weaknesses:** Some information may be outdated, and the app has ads that may deter some users.
- **Instabridge:** A similar app to Wi-Fi Map, but with an emphasis on automatically connecting users to free Wi-Fi hotspots without needing to enter passwords.
  - **Strengths:** Seamless connection to Wi-Fi networks without needing to input credentials.
  - **Weaknesses:** Limited in some regions, and connectivity issues can arise with user-generated data.
- **Wiman:** A global Wi-Fi network provider that offers Wi-Fi connectivity and maps of free networks.
  - **Strengths:** Offline access to Wi-Fi maps, making it easier for travellers and users with limited data.
  - **Weaknesses:** Limited customization and fewer smart features compared to the competition.
- **Google Maps:** Provides areas with Free Wi-Fi.

### Opportunities for our App:

- **AI Integration for Smart Recommendations:** Unlike existing apps, our app can stand out by offering AI-driven recommendations, adjusting suggestions based on user preferences (e.g., quiet places for working, cafes with good views), among others.

- **Gamification and Points System:** By incentivizing users to add and confirm Wi-Fi spots, our app can create a more reliable and up-to-date database of hotspots. Existing apps don't emphasize user rewards as much as you plan to.
- **Quality of Connection:** Many apps focus on location but don't prioritize the **quality** of the Wi-Fi connection. This can also be a part of the gamification. The application gives extra points when giving qualitative or quantitative information about the connection. For example, for the qualitative information, the user can say the connection did not buffer a Netflix show and the place had 10 people inside, for the quantitative, the user can upload a speed test print screen, from google or speed test by ookla.

## Interviews with Customer

The Gamification feature designed by the team was not very welcomed by the customer. We designed a Gamification system that consisted in obtaining points doing new Wi-Fi spots registrations, reviews, "using" Wi-Fi spots, among others. These points would be used in a shop to buy promotions in Wi-Fi spots and Avatar customizations such as clothes. The idea about promotions was that he understands it, but it is not a priority, in the future it could be considered. However, the whole Avatar system should be removed.

## Non-Functional Requirements (NFRs)

1. **Scalability:** The app should support many users and Wi-Fi spot data across multiple cities without performance degradation.
2. **Performance:** The app should load and display nearby Wi-Fi spots on a map within 3 seconds of opening.
3. **Security:** The app should encrypt all personal user data and use secure connections (HTTPS) for all communications.
4. **Availability:** The app should have 99.9% uptime, always ensuring reliable access to Wi-Fi spot data.
5. **Usability:** The app should be intuitive, with a user-friendly interface, ensuring users can easily find Wi-Fi spots and contribute without needing a tutorial.
6. **Responsiveness:** The app should function smoothly on a variety of devices (smartphones, tablets).

## User Journeys

### User Journey 1: Searching for a Wi-Fi Location

I am at home, and I am tired of studying inside the same monotonous space. I decided to search for a Wi-Fi spot in the application. I searched for some relaxed and chilled environment, and that sells food and beverages in the advanced filtering, but I did not like any of the suggestions so instead I used the AI input box and explained my mood and goal (studying for my next exam). The application gives me suggested locations with decent Wi-Fi needed to open documents and watch videos. I click on my choice of location and my desire navigation application opens with directions to the point.

After arriving at the Wi-Fi spot, I marked my visit and won some points.

Leaving the Wi-Fi spot, I really liked the spot I am at, so I decided to leave a review. I inserted some attributed and classified qualitative and quantitative, such as Quiet: 5 stars, Relaxation: With some sofas and tables, very nice place to stay. I left a comment on why I was there and how I could accomplish my goals, which were studying and watching lecture videos.

After leaving this review, I earned more points!

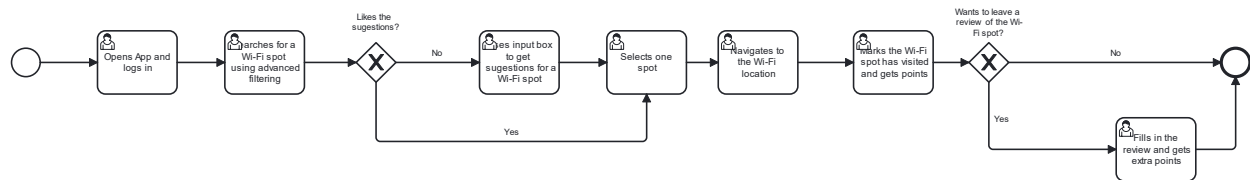


Figure 1 - User Journey 1

### User Journey 2: Registering a New Wi-Fi Location

I just arrived at a new spot, and I realized that had Wi-Fi. I decided to check the app if it was available on the list to leave a review, but I realized that it was not available at the application yet! So, I decided to insert the new location with the information of the address automatically inserted, and only added the environment features, the quality indicators, facilities and weather features.

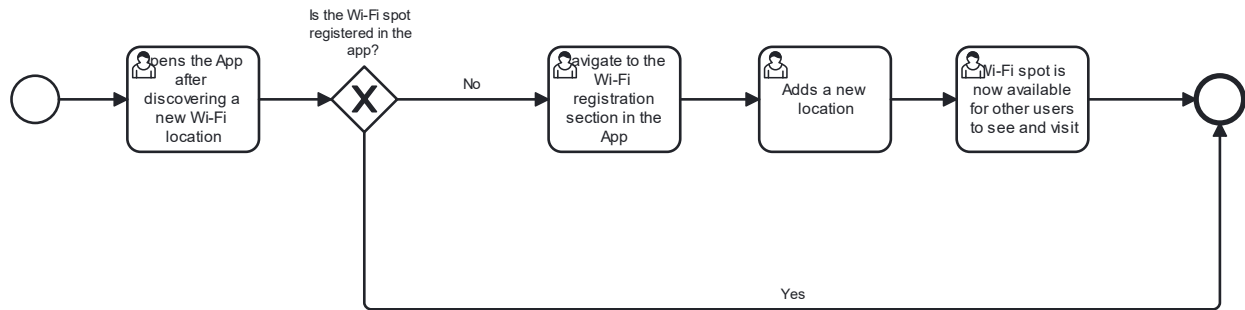


Figure 2 - User Journey 2

### User Journey 3: Registering My Location as a Premium Establishment and Creating Offers

As the owner of a popular coffee chain, I recently subscribed to the app as a premium user to promote my establishments and attract more Wi-Fi seeking customers. When I log in, I'm brought to the Business Dashboard, where I can see various options to manage my Wi-Fi spots and create special offers for the verified ones.

Since my cafes aren't verified in the app yet, I start by registering one of my locations as premium. I navigate to the "Manage Locations" section, find my coffee on the map, select it, and choose to verify it as a premium location. Once verified, my coffee now has a visible "Verified" badge that all users nearby can see, helping it stand out as a trusted Wi-Fi spot.

With the verification done, I want to boost traffic by setting up a promotion. I go to the "Offers" section in my dashboard and create a new discount: "20% off on Large Coffee." I fill in the details, setting the requirement for users to have at least 500 points to redeem it. After adding a brief description, specifying the promotion's validity for one month, and noting that it's redeemable once per visit, I publish the offer.

Now, my promotion is visible in the app's "Store" section, making it easy for nearby users to discover and redeem the discount with their points, bringing more foot traffic to my café.

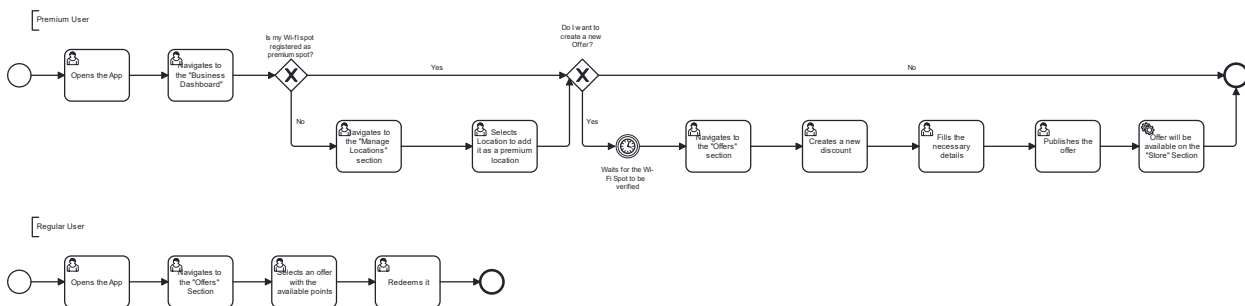


Figure 3 - User Journey 3

# Epics

## User Management

- As a non-registered user, I want to create my account details so that I can use the application.
- As a user, I want to delete my account from the application so that I stop using the application
- As a user, I want to log in to the app using my credentials so that I can access and use the application and its features.
- As a user, I want to log out of my account to ensure my information is secure.
- As a user, I want to view the history of all Wi-Fi locations I have visited so that I can keep track of where I have connected and assess my previous experiences.
- As a user, I want to view my history of reviews so that I can keep track of all the Wi-Fi spots I have reviewed.
- As a user, I want to manage my account details (edit) so that I can maintain accurate and up-to-date information.

## Wi-Fi Spot Registration

- As a user, I want to see a map that displays my current location
- As a user, I want to select a location on the map and register a new Wi-Fi spot to share with others.

## Wi-Fi Spot Visit and Reviews

- As a user, I want the app to record my visit to a Wi-Fi spot so I can track my history and earn points.
- As a user, I want to leave a review for the Wi-Fi spot I visited to help others know the quality of the connection and amenities.
- As a user, I want to view the history of reviews for a specific Wi-Fi location when I click on its details on the map, so that I can assess the quality and experiences shared by other users before deciding to visit or connect.
- As a user, I want to select a Wi-Fi location on the map and have the option to be guided there.

## Points System

- As a user, I want to view my current points balance so that I know how many points I have earned.

- As a user, I want to see the history of all my point transactions to track how I've earned and spent points.

## Points Shop

- As a user, I want to spend points to receive discounts on partner services or goods.

## Documentation and Guidelines

- As a user, I want to view the GDPR policies to understand how my data is used on account creation.
- As a user, I want to access the app FAQ to understand how to use the app and its features.
- As a user, I want to know how I can earn points, so I can plan my activities accordingly.

## Wi-Fi Spot Search

- As a user, I want to filter Wi-Fi spots by criteria like speed, amenities, and crowd level to find the best location for my needs.
- As a user, I want the application to recommend the best Wi-Fi spots based on my request, so that I can find the most suitable location for my needs.
- As a user, I want to use a search engine to find Wi-Fi spots by name or location.

## AI-Powered Suggestions

- As a user, I want the app to suggest the best times to visit certain Wi-Fi locations based on patterns of previous usage, so I can avoid busy periods and enjoy a more efficient connection.

## Dashboard

- As a user, I want to see a list of the Wi-Fi locations I have created so I can easily manage them
- As a user, I want to edit the details of a location so that I can keep the information up to date.
- As a premium user, I want to add offers to my premium Wi-Fi locations so that users can redeem their points for offers
- As a premium user, I want to view statistics for my verified Wi-Fi spots so I can understand their performance and engagement with other app users

## Manage Verified Location

- As a premium user, I want to assign the role of worker to users that work at my verified location so they can manage and redeem offers on behalf of clients.
- As a worker, I want to redeem a client offer so that I can provide clients with the benefits associated with the offer.

Every User Story mentioned is detailed in ours [Product Backlog](#).

## Gamification

### **Register a new Wi-Fi location:**

- When a user identifies and adds a new free Wi-Fi location, they earn points.

### **Receive a positive review for a Wi-Fi location shared:**

- If another user uses and gives a positive review to a Wi-Fi spot that the user shared, the original user earns points.

### **Connect to a new Wi-Fi:**

- Users earn points for connecting to a new Wi-Fi spot they haven't connected to before.

### **Leave a review for a Wi-Fi location:**

- Users gain points when they review the quality of a Wi-Fi spot.

### **A partner claims a location with a premium account:**

- When a premium partner claims a Wi-Fi location originally added by a regular user, that user earns many points.

### **Promotions:**

- Users can use points to access offers on **premium accounts**. These premium accounts are for businesses (e.g., McDonald's) that pay to market their establishment within the app, increasing customer foot traffic by offering their Wi-Fi locations.

# PHASE 3

## Product Vision

### Why?

Are you tired of working or studying from home? Our application offers the best locations in your city where you can enjoy free Wi-Fi. Whether you want a change of scenery, need a quiet outdoor space, or just want to work in a café, this app will help you find the perfect spot.

### What?

The app allows users to search for and mark zones where free Wi-Fi is available. As you use the app, you can confirm the availability of Wi-Fi at specific locations, and the user who first marked the spot earns points. It's a community-driven initiative to map and share the best Wi-Fi spots across the city.

### How?

- **User-driven Mapping:** Users contribute by adding locations with free Wi-Fi, helping others find convenient spots.
- **Confirmation and Rewards:** Users confirm Wi-Fi availability when they visit these spots, and the original contributor earns points for each confirmation.
- **Point System:** Points can be used for rewards, such as redeeming mobile data or other offers at partner locations.
- **Interactive Map:** The app provides a map that displays all free Wi-Fi spots in real-time, helping users navigate to the best location.
- **AI Suggestions:** With AI-powered suggestions:
  - The app suggests the best times to visit certain Wi-Fi locations based on patterns of previous usage, so user can avoid busy periods and enjoy a more efficient connection.
- **Search Filters:** Users can search for locations based on specific preferences, such as outdoor areas with a view, quiet places to study, or trendy spots to work.



## Personas

### Persona 1: The Regular User (Remote Worker / Student / Digital Nomad / Casual User)

**Name:** Jamie Nguyen

**Age:** 29

**Occupation:** Freelance Graphic Designer / Digital Nomad

#### Motivations

- To find convenient, high-quality Wi-Fi locations that support remote work, studies, or leisure without disruptions.
- To discover new spots based on personal preferences, such as quieter environments with reliable Wi-Fi.
- To take advantage of point-based rewards and discounts as a part of the app's social and financial perks.

#### Goals

- Locate Wi-Fi spots that are rated highly by other users for connectivity and environment.
- Earn points by engaging with the app, which can be redeemed for discounts or benefits at partner establishments.
- Access AI-driven suggestions to simplify the search for ideal locations based on preferences and past activity.

#### Pain Points

- Inconsistent Wi-Fi quality, especially in busy or unfamiliar locations.
- Lack of information about crowd levels or the atmosphere of locations.
- Difficulty balancing connectivity with a comfortable, distraction-free environment.
- Lack of human interaction and mobility, such as walking and not staying in the same place all day.

#### How the App Helps

- **Customizable Search:** Jamie can input specific preferences (e.g., “quiet location with fast Wi-Fi”) and receive a filtered list of options that match.
- **AI Suggestions:** AI analyses Jamie’s activity and preferences, recommending times and spots that align with work-friendly settings.

- **Points and Rewards:** Jamie can earn points for finding and reviewing locations, which can then be redeemed at premium partner spots, creating value for consistent usage.
- **Takes you outside.** By giving suggested locations with high quality Wi-Fi based on his goal, Jamie now can leave home, do his work, have human interaction outside and walk/drive to get to know undiscovered places of the city.

## Persona 2: The Premium Establishment (Verified Partner Location)

**Name:** Alex Patel

**Age:** 45

**Occupation:** Owner of Coffee Shop Chain

### Motivations

- To increase foot traffic from remote workers and digital nomads seeking reliable Wi-Fi.
- To distinguish the coffee shops as verified and preferred locations on the app for enhanced visibility.
- To foster loyalty and recurring visits through exclusive discounts available via the app.

### Goals

- Enhance the coffee shop's brand presence on the app as a verified, premium Wi-Fi location.
- Offer unique discounts or promotions (e.g., 20% off on select items) that can be redeemed through user points to attract loyal customers.
- Utilize the app's user analytics and traffic data to optimize store hours, staffing, and customer service.

### Pain Points

- Managing peak times when Wi-Fi usage is high and ensuring a positive experience despite crowding.
- Competing with other coffee shops and coworking spaces in busy areas.
- Balancing promotional efforts with the goal of building long-term customer loyalty.

### How the App Helps

- **Verified Premium Status:** Alex's coffee shops appear prominently on the app, attracting users seeking trusted, high-quality Wi-Fi.
- **Point-Driven Discounts:** The app enables Alex to create redeemable discounts (e.g., 20% off a purchase with enough points), boosting loyalty and encouraging repeat visits.
- **Foot Traffic Insights:** Alex can view user patterns and preferences to adjust operational hours or promotions, creating an optimized experience for visitors.

## General Workflow

1. **User Opens App:** Sees a map with free Wi-Fi locations nearby.
2. **User Filters Results:** Chooses filters like indoor/outdoor, rating, connection quality, or AI suggestions.
3. **User Contributes:** Adds a new Wi-Fi location or confirms an existing one, earning points for contributions.
4. **User Redeems Points:** Users can track points earned and redeem them for rewards like mobile data.

## Domain Model

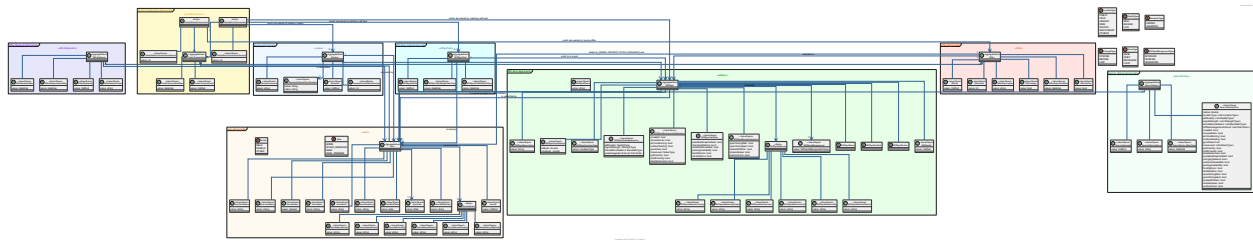


Figure 4 - Domain Model

The domain model is based on six Aggregates:

- **User** – It contains all the user information and their roles.
- **Wifi Spot** – All the information needed for characterization of a spot. It is important to have this information for advanced filtering
- **Review** – This aggregate is directly connected to the Wifi Spots, it contains all the reviews done by users
- **Offer** – Offer aggregate, that contains the offers available for the users
- **Offer Redeem** – Aggregate that joins the redeem of the offer on the store by the user

- Points Transaction – This aggregate contains the transactions when earning points by doing some action such like creating, visiting or reviewing a spot and when wasting the points on offers.
- Search History – This aggregate is useful to save data from the user and their behaviour so the suggestions and other components can be more accurate than only having the visits.

## Architecture Solution

The decision of the architectural structure will be a modular monolithic. This happens because our project will start from scratch and the natural flow of an application is to start with a monolithic due to their initial simplicity.

For better maintainability and with the intention of, in the future, migrating to microservices, it will be already defined the boundary contexts, as a modular monolithic.

## Boundary Contexts

Our boundary contexts include the following aggregates:

### **User:**

- User

### **Review:**

- Review

### **Store:**

- Offer
- Offer Redeem

### **Wifi Spot:**

- Wifi Spot
- Wifi Spot Visit

### **Points:**

- Points Transaction

### **Search History:**

- Search History

## Component Diagram Level 2

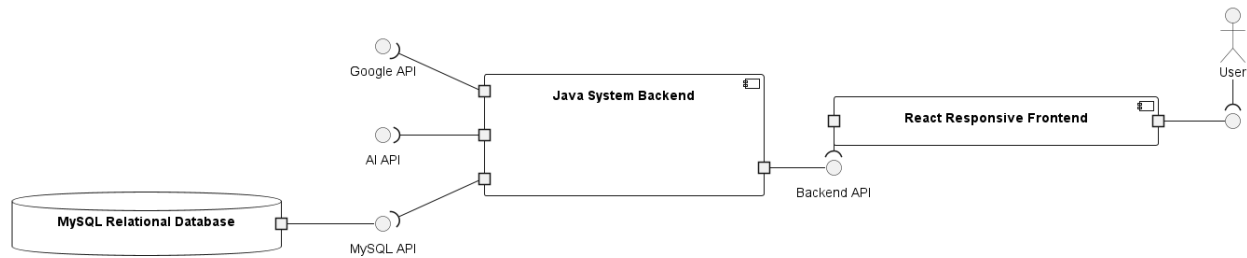


Figure 5 - Component Diagram Level 2

In this diagram, there is only one Component for the backend. In the future, this can be distributed in components based on the boundary contexts explained.

There will only be one frontend, being responsive, it can be used by Mobile Phone users, thus simplifying the first steps of development. There can be a Mobile Application in the future.

The frontend will consume the backend API to gather the data necessary. This data will be persisted in only one database.

It was thought and analysed to use more than one database, one for each boundary context, but based on the same reason as explained before, there will not be a component for each, there will only be one database.

If the system and the database is already configured and designed with the boundary contexts in mind, it will be an easy and smooth process to migrate from a common database to a boundary context individual database.

Based on technologies, it will be used React to develop the frontend based on the experience of the team and because it has a powerful native responsive system.

For the backend technology, Java will be used since we have hands 'on experience with it. It is a widely known language with support for Object Oriented Programming and Domain Driven Design. It also works in most devices thanks to the Java Virtual Machine system it uses.

The database technology used will be MySQL, it is an open-source database, and it has a large community and documentation available. The use of the relational database is based on the knowledge of the team, but also because the domain is intuitive for a relational model first, based on a monolithic. In the future, it is possible that, depending on the boundary context and its business purpose, a non-relational database for that component can be justified.

## Assumptions and Dependencies

There are certain assumptions to have when using the application:

- Users have access to internet connection
- Users have location services enabled on their devices, allowing the app to accurately determine their current location.
- Users are likely to have at least one navigation app (Google Maps or Waze) installed on their devices for the redirection feature to work.
- Users are familiar with basic smartphone functionalities, including how to interact with maps and navigation apps.
- Users are motivated to find and use Wi-Fi locations for personal or professional needs, such as studying, working, or socializing.
- Users are aware of their surroundings and use common sense while navigating to new locations, particularly in unfamiliar areas.
- Users may have varying levels of accessibility needs, and the app should consider features to accommodate those needs.

For the external dependencies, whenever the AI API is down, the AI Features of our app stop working, however the application can still be used. The same goes for the Google API, which is the GPS Service.

## Risk Management Plan

### 1. Internet Connection Availability

- **Risk:** Users may experience poor or no internet connection, impacting app functionality.
- **Mitigation:** Implement offline functionality for certain app features, like cached locations. Provide reminders to reconnect for optimal experience.

### 2. Location Services Dependency

- **Risk:** Users may have location services disabled, limiting app accuracy in suggesting nearby Wi-Fi.
- **Mitigation:** Display a prompt for users to enable location services. Offer manual search options for users to input their location manually.

### 3. Navigation App Requirement

- **Risk:** Users without a navigation app won't be able to use redirection features.

- **Mitigation:** Include in-app directions as a basic alternative. Recommend compatible navigation apps upon setup.

#### 4. **User Familiarity with Smartphone Functions**

- **Risk:** Some users may struggle with map interaction or navigation features.
- **Mitigation:** Incorporate user-friendly tutorials and in-app tips to guide new users through app functionalities.

#### 5. **User Motivation for Wi-Fi Use**

- **Risk:** If users are not actively motivated to find Wi-Fi locations, app engagement may suffer.
- **Mitigation:** Introduce rewards such as offers so users can redeem buying with points gained by interacting with the application.

#### 6. **User Safety Awareness**

- **Risk:** Users may not pay attention to their surroundings while navigating to new locations.
- **Mitigation:** Include safety reminders and encourage users to be cautious, especially in unfamiliar areas.

#### 7. **Accessibility Needs**

- **Risk:** Users with accessibility needs may face challenges using the app.
- **Mitigation:** Implement accessible design principles, including text scaling, high-contrast modes, and compatibility with screen readers.

#### 8. **AI API Dependency**

- **Risk:** When the AI API is down, AI-powered features will be unavailable.
- **Mitigation:** Develop a fallback for core functionalities so the app remains usable without AI features. Notify users when AI features are temporarily unavailable.

#### 9. **Google API Dependency for GPS**

- **Risk:** If the Google GPS service is unavailable, location-based features may not work accurately.

- **Mitigation:** Implement messaging to inform users of limited functionality during GPS outages and consider alternative ways for users to input or update their location.

Regular risk reviews and monitoring these mitigation strategies will help reduce the impact of these potential issues.

## Deployment Diagram decisions

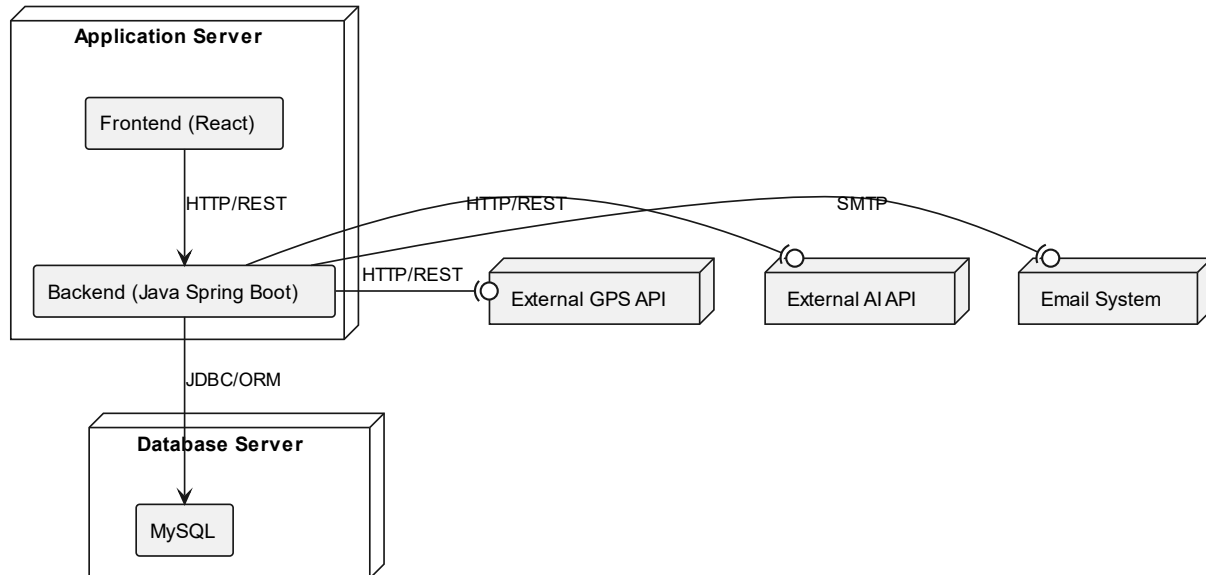


Figure 6 - Deployment Diagram

## Database Server

We decided to have only one database server because according to the dimension of the application and the amount of data it will have, it is not justified. There will not be a lack of performance on its connections and even in the future if there is any need to migrate the application to microservices, it will be possible anyway. MySQL is a free and open-source technology suited for academic applications, as mentioned before.

## Application Server

Since the application is considered a start-up, we decided to use something easier to utilize like render which is a platform specifically designed for simplifying the deployment and hosting of web applications and static websites, while having a free tier. We considered using cloud services like AWS or Azure, but the fact that it's not free (REST API calls and deployment service) and its main advantages aren't necessary for such a small project as this one made us ignore it for the moment.



Also, the decision to make the front-end and back-end application into a single server was due to it being easier to develop and since we are a small team, its management is easier.

REST APIs [Info](#)

Select the units, number, and frequency for REST API requests based on expected volume. The calculations below exclude free tier discounts.

REST API request units

The unit will apply to the number of requests you will specify below.

thousands

Requests

10000

Unit

per month

Cache memory size (GB)

Optional. Choose cache memory size in GB. Dedicated cache memory is billed at an hourly rate.

None

▼ Show calculations

10,000 requests x 1,000 unit multiplier = 10,000,000 total REST API requests

Tiered price for: 10,000,000 requests

10,000,000 requests x 0.0000035 USD = 35.00 USD

Total tier cost = 35.00 USD (REST API requests)

**Tiered price total for REST API requests: 35 USD**

0 USD per hour x 730 hours in a month = 0.00 USD for cache memory

**Dedicated cache memory total price: 0.00 USD**

**REST API cost (monthly): 35.00 USD**

Figure 7 - Estimation Cost for REST API in AWS

## MOSCOW

### Must

- As a non-registered user, I want to create my account details so that I can use the application.
- As a user, I want to delete my account from the application so that I stop using the application
- As a user, I want to view the GDPR policies to understand how my data is used on account creation.
- As a user, I want to log in to the app using my credentials so that I can access and use the application and its features.
- As a user, I want to log out of my account to ensure my information is secure.
- As a user, I want to see a map that displays my current location
- As a user, I want to select a location on the map and register a new Wi-Fi spot to share with others.
- As a user, I want to select a Wi-Fi location on the map and have the option to be guided there

- As a user, I want the app to record my visit to a Wi-Fi spot so I can track my history and earn points.
- As a user, I want to filter Wi-Fi spots by criteria like speed, amenities, and crowd level to find the best location for my needs.
- As a user, I want to leave a review for the Wi-Fi spot I visited to help others know the quality of the connection and amenities.

## Should

- As a user, I want the application to recommend the best Wi-Fi spots based on my request, so that I can find the most suitable location for my needs.
- As a user, I want to view the history of reviews for a specific Wi-Fi location when I click on its details on the map, so that I can assess the quality and experiences shared by other users before deciding to visit or connect.
- As a user, I want to see a list of the Wi-Fi locations I have created so I can easily manage them
- As a user, I want to edit the details of a location so that I can keep the information up to date.
- As a user, I want to access the app FAQ to understand how to use the app and its features.

## Could

- As a user, I want to know how I can earn points, so I can plan my activities accordingly.
- As a user, I want to view my current points balance so that I know how many points I have earned.
- As a user, I want to view the history of all Wi-Fi locations I have visited so that I can keep track of where I have connected and assess my previous experiences.
- As a user, I want to view my history of reviews so that I can keep track of all the Wi-Fi spots I have reviewed.
- As a user, I want to manage my account details (edit) so that I can maintain accurate and up-to-date information.

## Wont

- As a premium user, I want to add offers to my premium Wi-Fi locations so that users can redeem their points for offers
- As a user, I want to spend points to receive discounts on partner services or goods.

- As a user, I want to see the history of all my point transactions to track how I've earned and spent points.
- As a premium user, I want to view statistics for my verified Wi-Fi spots so I can understand their performance and engagement with other app users
- As a user, I want to use a search engine to find Wi-Fi spots by name or location.
- As a premium user, I want to assign the role of worker to users that work at my verified location so they can manage and redeem offers on behalf of clients.
- As a worker, I want to redeem a client offer so that I can provide clients with the benefits associated with the offer.
- As a user, I want the app to suggest the best times to visit certain Wi-Fi locations based on patterns of previous usage, so I can avoid busy periods and enjoy a more efficient connection.

## Cost Analysis

### 1. Hardware Costs

Hardware is essential for the development and hosting of the application. Even though the project uses open-source software, the hosting infrastructure and developer equipment come with costs.

#### *Database and Application Servers*

- **Database Server:**
  - The project uses MySQL, which is free, but hosting this database requires a virtual private server (VPS) or cloud server. For a small-scale project, an entry-level VPS would suffice.
  - Estimated cost: **€50 - €100/month** for a basic server (e.g., a 2GB RAM, 50GB SSD VPS from providers like DigitalOcean or Linode).
- **Application Server:**
  - The app will initially be hosted on **Render**, which has a free tier for small projects. However, if the project scales, paid cloud services or VPS solutions might be required to ensure performance and reliability.
  - Estimated cost for scaling: **€100 - €500/month**, depending on traffic and usage levels.
  - Considering there will be a staging and production environment, the costs will be duplicated. However, the staging environment can be deployed to a cheaper server.

## ***Developer Equipment***

- **Laptops and Accessories:**

- Each developer needs a reliable setup to develop, test, and debug the application. This typically includes high-performance laptops, monitors, and necessary peripherals.
- Estimated cost for each developer setup (laptop, monitor, accessories): **€2,000 - €3,000.**
- The team includes at least 1 front-end, 1 back-end developer, a UX/UI designer and a Quality Assurance Tester. But it can increase to 10.
- Total hardware costs for four developers: **€8,000 - €12,000 to €20,000 - €30,000 (every 4 years).**

## **2. Software Costs**

The project uses several open-source tools, reducing the need for licensing fees. However, additional software tools for project management and collaboration may be required.

### ***Licenses and Tools***

- **MySQL, React, Java:**

- These technologies are open-source and free to use: **€0.**

- **Project Management and Collaboration Tools:**

- Tools such as **Jira** for project management and **GitHub Premium** for repository hosting and team collaboration can have costs, especially when managing larger teams.
- Estimated cost: **€10 - €15/user/month**. For a team of 10 people, this equates to **€100 - €150/month**.
- Developers can use free available tools to develop such as Microsoft Visual Studio and Eclipse, however there is the availability of the use JetBrains IDEs, **€50/user/month** based on official estimate.

## **3. Salary Costs**

Salaries make up the bulk of the project's expenses. The project requires a multidisciplinary team, each contributing to different aspects of development, management, and support. Salaries are based on industry standards in Europe.

## ***Team Composition and Average Salaries (Annual Costs)***

- **Front-end Developer:**
  - Responsible for building the user interface, ensuring responsiveness, and handling the design implementation.
  - Estimated salary: **€40,000 - €50,000/year.**
- **Back-end Developer:**
  - Manages the server, database, API, and core app functionality.
  - Estimated salary: **€40,000 - €50,000/year.**
- **Product Manager (PM):**
  - Oversees the project strategy, stakeholder management, and roadmap planning.
  - Estimated salary: **€55,000 - €65,000/year.**
- **Business Analyst (BA):**
  - Gathers requirements, documents workflows, and ensures the team understands the business objectives.
  - Estimated salary: **€45,000 - €60,000/year.**
- **UX/UI Designer:**
  - Designs user-friendly interfaces and ensures the app's visual appeal and accessibility.
  - Estimated salary: **€30,000 - €40,000/year.**
- **DevOps Engineer:**
  - Responsible for continuous integration (CI), continuous deployment (CD), cloud infrastructure, and app reliability.
  - Estimated salary: **€40,000 - €50,000/year.**
- **Data Scientist/AI Specialist:**
  - Develops algorithms for Wi-Fi recommendations and AI-driven suggestions based on user preferences and weather data.
  - Estimated salary: **€40,000 - €55,000/year.**
- **QA/Testers:**
  - Ensures that the app works correctly by conducting manual and automated testing.
  - Estimated salary: **€30,000 - €40,000/year.**
- **Marketing Manager:**
  - Plans and executes user acquisition campaigns and builds partnerships with local businesses.
  - Estimated salary: **€30,000 - €40,000/year.**

- **Customer Support:**
  - Manages user queries, technical support, and community engagement.
  - Estimated salary: **€30,000 - €40,000/year**.

**Total Annual Salary Costs for a 9-member team:**

- Low estimate: **€380,000/year**.
- High estimate: **€490,000/year**.

## **4. Cloud Infrastructure Costs**

As the app grows, cloud infrastructure costs will increase, especially if the project transitions from a free tier to paid cloud services for scalability and performance.

***Render Hosting and Cloud Services:***

- **Render** offers a free tier for small projects. However, scaling the application may require moving to a more robust platform like **AWS** or **Azure**.
- **AWS/Azure/Render (paid tier):** Monthly cost could range from **€500 - €1,500/month** depending on the app's traffic and resource usage.

## **5. Marketing Costs**

Effective user acquisition and partnership development are critical to the app's growth. These costs include digital marketing campaigns, user outreach, and building relationships with local businesses or public Wi-Fi providers.

***Digital Campaigns and User Acquisition:***

- Estimated cost for small-scale online campaigns (Google Ads, Facebook Ads): **€1,000 - €3,000/month**.

***Partnership Development:***

- Partnering with cafes, coworking spaces, and local businesses for visibility and user engagement may incur costs for marketing material or partnership agreements.
- Annual budget for partnership development: **€5,000 - €10,000/year**.

## 6. Miscellaneous Costs

### *Security, Legal, and Compliance:*

- The app will need to comply with data privacy laws such as **GDPR** in Europe, which may require legal consultation and the implementation of proper data protection measures.
- Estimated annual cost for compliance and security measures: **€5,000 - €10,000/year**.

### Summary of Estimated Costs

Category	Low Estimate (€)	High Estimate (€)
Hardware (monthly)	€150/month	€600/month
Software (monthly)	€600/month	€650/month
Salaries (annual)	€380,000/year	€490,000/year
Cloud Infrastructure (monthly)	€500/month	€1,500/month
Marketing (monthly)	€1,000/month	€3,000/month
Miscellaneous (annual)	€5,000/year	€10,000/year
Laptops and peripherals (every 4 year)	€8,000/4 year	€30,000/4 year
Total (annual)	€414,000/year	€676,500/year