EcoRoute - v0.2.0

Andrea Valentini

Last update: 10th April 2024

Contents

1 The project and project goals 3

2 Requirement analysis 5

2.1 Relevant human and non-human actors 5

2.2 Use cases 6

2.3 Domain assumptions 7

2.4 Requirements 7

2.4.1 Functional requirements 7

2.4.2 Non-functional requirements 7

3 Design 8

3.1 General description of the architecture 8

3.2 Sequence diagrams 8

3.3 Critical points and design decisions 8

1 The project and project goals

The following is a description of the project problem and the goals to be achieved to complete the assignment. We have divided this section into three groups:

- The **preface** (or scenario) helps understand the environment to develop a sound software system.
- The **problem posed** section includes lists to emphasize the critical points.
- The **goals to achieve** by the assignment.

Note: We analyzed the **citizens' stakeholders** in this project.

Preface

Two urgent global concerns are environmental sustainability and climate change; because of air pollution and greenhouse gas emissions, transportation - especially urban commuting - contributes to worsening those issues.

Even today, urban areas are characterized by a heavy reliance on personal vehicles, which are seen as the most comfortable and efficient way of commuting, despite several studies showing that better alternatives exist in most cases.

Improving public transportation systems' efficiency can make them more appealing to daily commuters and is, therefore, a promising way to lessen environmental impact and, at the same time, to increase the overall quality of citizens' life (article).

Problem posed

The project "Eco-City Commute" (ECC) aims to create a comprehensive software system that makes public transportation within an urban area as easy and efficient as possible, promoting its adoption.

ECC receives data from sensors, deployed on public transport means, that provide:

- Information about their respective occupancy rates.
- Real-time information about public transit timetables.
- Information about bike and ride sharing, from specific services (think at ATM in Milano, BikeMi, Mobike, BlaBlaCar, ...).

Based on these pieces of information, ECC offers services to two types of stakeholders:

- **Citizens**: ECC offers a *mobile app* that allows citizens to input:
  - The origin (within the urban area);
  - The destination (within the urban area);
  - Eventually constraint, for example: they do not want to use a bike; they must arrive at destination within a certain timeframe.

The application takes the input and displays (output):

  - Environmentally friendly routes possibly combining different transportation means.

- **Urban area managers**: ECC offers to managers a dashboard through which they can visualize reports concerning the daily usage of the various available transportation means, their occupation rates and delays (if any).

Goals to achieve

We analyzed the **citizens' stakeholders**. So, the main goal was to develop a software system that offers citizens a mobile app to make public transportation within an urban area as easy and efficient as possible. Therefore, the document seeks to meet two objectives:

- Analyze the requirement aspects.
- Make a well-architecture design.

2 Requirement analysis

2.1 Relevant human and non-human actors

The only *human actor* is the citizen:

- **Citizen**. A user who uses the mobile application and enters the origin and destination of his journey. As the problem says, it can also add some constraints to the research.

Instead, there are several *non-human actors* that allow the user to search for some specific services or even a public transport timetable:

- **Sensor**. An electronic device installed on public transport vehicles that provides information about their occupancy.
- **PTT Server**. A Public Transit Timetable Server identifies the public transport company's server. The application makes some queries to this server to find out which public transport line is available at the time specified by the citizen actor.
- **BRS Server**. A Bike-Ride-Sharing Server identifies the server that the mobile application can query to get the information requested by the citizen actor. The BRS Server actor is as abstract as possible, because we want to be able to add as many services as we want.

  For example, a BRS server could be the BlaBlaCar site that the application queries to know if there is a car available for rent.

The Citizen is the primary actor because he is the stakeholder and the main user of the mobile application.

Instead, the non-human actors are all supporting actors because they provide a service to the application (e.g. the sensor offers the occupancy rate of public transport).

2.2 Use cases

The following use case diagram represents the EcoRoute application with a high level visualization. In the diagram it is possible to see the primary actor (citizen) that interacts (initiate) with the application and the supporting actors (Sensor, PTT Server, BRS Server) that support (participate) the application with the collection of the data from different services.

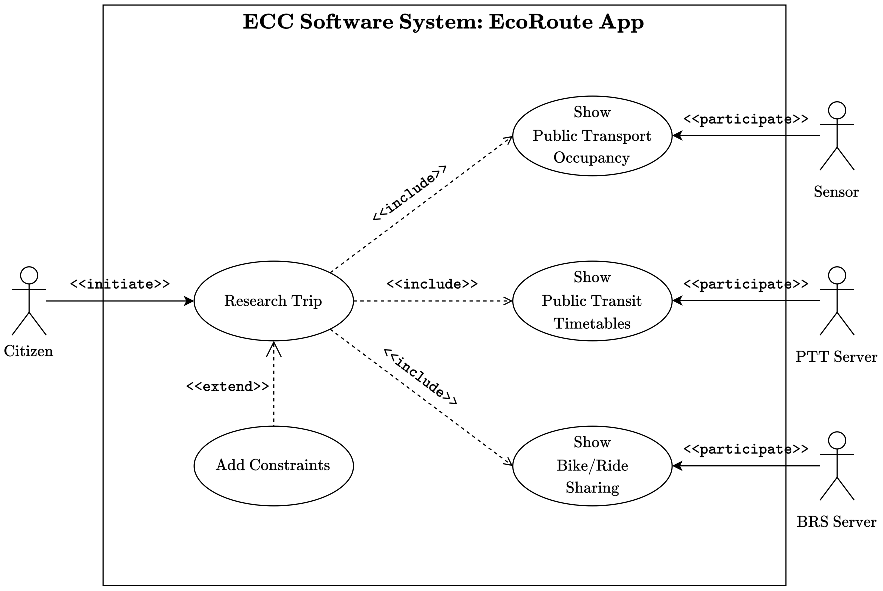

Figure 1: Use case diagram for the EcoRoute App.

To view/download the Use Case Diagram in high quality, click or scan the QR code below:

The use cases identified are 5:

- **Research Trip**
- **Add Constraints**
- **Show Public Transport Occupancy**
- **Show Public Transit Timetables**
- **Show Bike/Ride Sharing**

2.3 Domain assumptions

2.4 Requirements

2.4.1 Functional requirements

2.4.2 Non-functional requirements

3 Design

3.1 General description of the architecture

3.2 Sequence diagrams

3.3 Critical points and design decisions