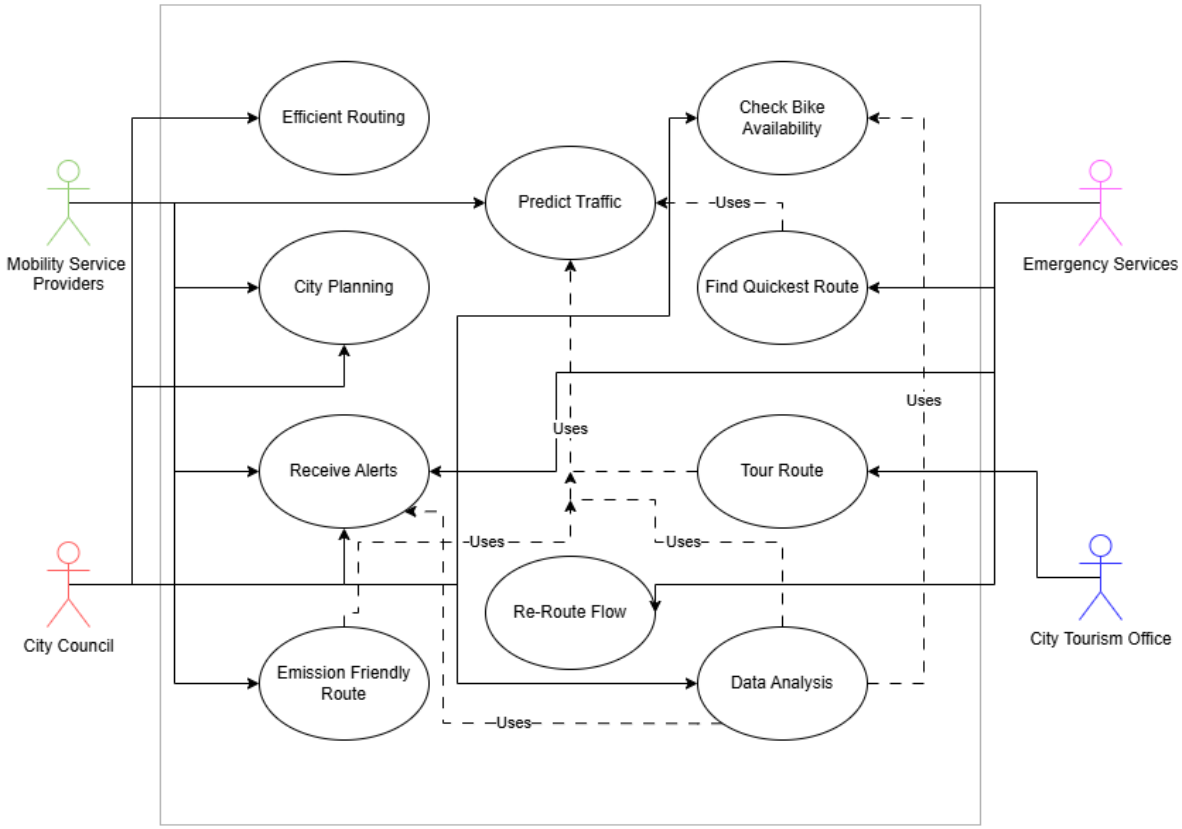
CS7CS3 Advanced Software Engineering Group Project

Requirements/Use Cases

Project Name: Sustainable City Management

Group: *9*
Aoife Walshe
Oisin Power
Parker Kavanagh
Darragh Hardiman Smyth
Charlie O'Malley

1. Use Case Diagram



2. Use Cases

2.1 Use Case Name: City Planning

1. Description

1.1 Goals & Responsibilities:

The main goal is to act on data insights to improve the urban mobility while maintaining sustainability in mind.
Some examples include:

- Ensuring bikes are evenly distributed around the city to avoid shortages or over-concentration of resources.
- Provide further public transport areas with higher volume of commuters travelling by car.
- Restrict car traffic or introduce pedestrian-only zones in highly polluted areas.
- Highlight areas that lack services, such as bus routes, bikes, etc.

The focus of each of these changes should revolve around the people who live in effected areas, for instance, how a new bus route may alter the commuters travel time to popular zones etc.

1.2 Actors:

This use case will be primarily used by the city council, as well as the mobility service providers as these will work in tandem to address new public transport route implementations, as well as any misalignments in the balance of resources throughout the city.

1.3 Triggers and Inputs:

The user will need to select the City Planning feature upon opening the application. Once this mode has been enabled, the user may enter the desired area to analyse, as well as selecting the specific data group being targeted, for example, emissions, bikes distribution, bus routes, etc. The user also has the option to input the timeframe the data covers, for instance, the past week, or past month. The application will then display all relevant data and suggestion for the selected data set.

2. Flow of Events

Basic Flow			
User		System	
1	User Selects City Planning mode		
		2	System displays City Planning mode
3	User picks data indicators, geographic area, and timeframe		
4	User clicks Analyse button		
		5	System retrieves data from Data Analysis use case using inputted information
		6	System processes data into uniform format
		7	System analyses correlations for input indicator
		8	System identifies and makes recommendations
		9	System displays recommendations for improved routes
10	User Exports analysis report		

Alternative Flow 1			
User		System	
Branch from step 5 of Basic Flow			
		5.1	System detects that a data source is not available
		5.2	System switches to displaying historical information
		5.3	Overlays historical information with current data from working data sources
		5.4	System display message “Using historical data due to current data unavailability”
5.5	User identifies hole in transport infrastructure		

3. Special Requirements

3.1 Platform

- Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

System will rely on a considerable number of Data sources

- Dublin linked open data API
- Real time bus location
- Dublin bikes API DCC
- Pedestrian count Api
- Bus route map
- Air Quality API
- Weather API (for correlation of transport data and rainy days)
- Maps API

4. Preconditions

- At least one data source is available
- API’s can be reached
- Historical data must be stored for fallback protection

5. Postconditions

- Data sources overlayed on map
- Current data individually selectable
- Historical data viewable via time selection menu
- Report of recommended changes ready to export

2.2 Use Case Name: Tour Route

1. Description:

1.1 Goals & Responsibilities:

This case intends to not only balance travel time but also sustainability by providing the tour operator with an optimal route that avoids major road congestion areas, primarily provided by the data analysis case, as well as information on attraction related data, such as opening times, pricing etc.
In the case of a walking route, the interface intends to provide the best route that covers a wide range tourism features while balancing travel time.

1.2 Actors:

The city tourism office and private tour companies are the key actors being able to track tourism attractions popularity, opening times, pricing etc. While being able to create appealing routes that cover said attractions that are in most demand in a costly and timely manner.

1.3 Triggers and Inputs:

The triggers would primarily involve selecting the tourism feature upon opening the application, this will then ensure that the program will act accordingly, providing the best route that involves all selected tourist attractions.
The inputs include a start and end point, as well as desired attractions or locations the user wishes to include in the route.

2.Flow of Event

Basic Flow			
User		System	
1	User selects tourist feature		
		2	The system modifies the display and provides a listing of the most popular tourist attractions in the moment, along with relevant information
3	User selects various attractions they wish to include in the route		
		4	The system prompts the user for a start and end location, as well as a method of transport required for use (walking or bus)
5	User inputs the start and end location and chooses the bus method		
		6	The system displays the optimal bus route and indicates the best times to commence the route.

Alternative Flow 1			
User		System	
Branch from step 5 of Basic Flow			
5.1	User inputs the start and end location and chooses the walking method		
		5.2	The system displays the optimal walking route which includes which attraction to visit in what order

Alternative Flow 2			
User		System	
Branch from step 2 of Basic Flow			
		2.1	System detects that a data source is not available
		2.2	System switches to displaying historical information
		2.3	Overlays historical information with current data from working data sources
		2.4	System display message “Using historical data due to current data unavailability”

3.Special Requirements

The system’s functions rely on the APIs providing relevant data regarding tourism attractions in Dublin. In addition, the system requires the availability of real time data regarding the status of road obstructions, traffic etc, provided by the data analysis case.

3.1. Platform

- Accessibility on major modern browsers (Firefox, Chrome, Safari)

3.2 External Dependencies

- Map Data (google maps or equivalent).
- Tourist attraction data (data.gov.ie).

4. Preconditions

- The System must have access to all relevant data to provide the user with an accurate route.
- User benefits from having an internet connection to obtain current information.

5. Postconditions

- The system will display routes that include trending tourist attractions, using buses or walking with time and sustainability kept in mind.

2.3 Use Case Name: Receive Alerts

1. Description:

1.1 Goals & Responsibilities:

This use case provides real-time notifications to service providers and planners regarding traffic incidents, anomalies, and user defined threshold breaches. Such alerts should include information regarding traffic congestion, accidents, road closures or power outages affecting traffic lights. The system gathers data from multiple sources and issues timely notifications to city managers to enhance situational awareness and support decision-making.

1.2 Actors:

The main actors for this use case are the City Council, Mobility Service Providers and the Emergency Services as they would need to know where accidents occur so that they can help mitigate or avoid the accident.

1.3 Triggers and Inputs:

The system can be triggered by several events. One trigger occurs when a user-defined threshold, such as traffic speed or volume, is exceeded, using inputs including traffic data, location, and duration. Another trigger is the detection of a traffic accident, which relies on inputs such as the location of the incident, its severity, and the volume of traffic affected. A third trigger arises from public transport delays or disruptions, with relevant inputs including the affected location, routes, and duration of the disruption.

2.Flow of Events

Basic Flow			
User		System	
1	User defines threshold		
		2	System continually monitors for threshold breaches
		3	System sends push notification if threshold is exceeded
4	User has access to information and data relevant to the notified incident		

Alternative Flow 1			
User		System	
Branch from step 1			
		1.1	Data sources not available, system notifies user that feature is not available

3.Special Requirements

3.1. Platform

- Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

- Map data (google maps or equivalent)
- Real time data regarding congestion, weather, etc.
- Availability of relevant API (google map etc).

4. Preconditions

- The System must have access to all relevant data to provide the user with an accurate route.
- An active internet connection is required for this feature, otherwise it would not be possible to give alerts on real time data.

5. Postconditions

- The system will provide the user with real time alert when a defined threshold has been exceeded.

2.4 Use case Name: Re-Route Flow

1. Description:

1.1 Goals and responsibilities:

- To provide information on current established transport routes such as utilization, emissions, capacity and suggest possible alternate or more efficient routes.
- To identify high traffic transport routes that could be disrupted, have their capacity reduced or blocked entirely and help design an alternate route plan to accommodate the traffic in the event of a blockage or capacity reduction.
- To help planners react quickly to unforeseen events that cause traffic disruption and provide alternate routes as well as prioritising Emergency Service access.
- The system should consider any disruptions in the surrounding routes if they are available from the data sources, these should only be used if the data is up to date
- The system should provide an explanation or reasoning for the best suggested route
- The time to process the suggestion should not be longer than 5 seconds.
- The system should provide more than one suggestion where possible.

1.2 Actors:

Traffic planners can use the system to identify patterns in traffic and create rerouting plans, get suggestions on more efficient routes based on what currently exists and to make inform decisions when planning around big events such as GAA matches, parades, festivals etc.
Guards can use this case to reroute traffic after a road accident for example to allow safer and faster access to the necessary Emergency Services.

1.3 Triggers and Inputs:

The system can be triggered by user actions or external events related to route management. One trigger occurs when a user requests to view overburdened or underutilized transport routes, with inputs including the desired area, selection of a specific route, and any request for rerouting suggestions. Another trigger arises when a user activates the temporary reroute feature in response to a large event, using inputs such as affected areas and transport routes, event duration, and specific service requirements (e.g., maintaining ambulance or Garda access). A final trigger occurs when the temporary reroute feature is used due to a storm or disaster, relying on inputs such as the affected area, severity of the event, and likelihood of disruption.

2.Flow of Events

Basic Flow			
User		System	
1	User selects rerouting function		
		2	System shows the user an interactive map with highlighted routes and utilization information
3	User selects route that is under or over utilized		
		4	System highlights that specific route as well as surrounding possible alternate routes
5	User requests suggestion for best alternate route based on desired characteristics		
		6	System highlights and provides reasons for the best possible alternate route or routes and presents them to the user

Alternate flow 1			
User		System	
Branch from step 3 of Basic Flow			
3.1	User selects the type of route that will be affected (bus, car, Luas etc.)		
		3.2	System highlights the specific type of route
3.3	User selects route that will be affected		
		3.4	System highlights that specific route as well as surrounding possible alternate routes
3.5	User selects severity of disruption (full blockage, capacity reduction, slowing traffic)		
3.6	User selects duration of disruption		
		3.7	System highlights and provides reasons for the best possible alternate route or routes and presents them to the user

Alternative Flow 2			
User		System	
Branch from step 2 of Basic Flow			
		2.1	System detects that a data source is not available
		2.2	System switches to displaying historical information
		2.3	Overlays historical information with current data from working data sources
		2.4	System display message “Using historical data due to current data unavailability”

3. Special Requirements:

3.1 Platform:

- The system will likely run on a webserver so access to the internet as well as a modern web browser would be necessary to utilize it.

3.2 External dependencies:

- Map data (google maps or equivalent)
- Requires availability of current or recent traffic data from the desired area.
- Accurate and up to date map with detailed information such as bus stops, cycle lanes, Luas tracks, pedestrianised roads etc.

4. Preconditions:

- Upcoming event with known location and duration
- The data for that area must be available for the system to give reliable suggestions

5. Postconditions:

- The system will show alternate routes and suggest the best one based on what causes the least disruption to other routes.

2.5 Use Case Name: Emission Friendly Route

1. Description

1.1 Goals and Responsibilities:

This will provide alternative routing with emissions considered. To reduce overall emissions and improve air quality to help the environment.
This will be done by:

- Taking a slower route with reduced emissions compared to the fastest route.
- Taking a shorter route through lower speed areas.
- Taking a longer one that will avoid idling in traffic.

Using these tactics this routing will reduce emissions and create a better environment with improved air quality and reduced emissions.

1.2 Actors:

It will be used by city planners and mobility service providers by highlighting which bus routes could be optimised to reduce emissions. It would also highlight the emission heavy routes that commuters use. They could use this information to add bus routes or alternative traffic management solutions to reduce emissions.

1.3 Triggers and Inputs:

The trigger occurs when the user selects the emission-friendly routing feature, prompting the system to generate routes that prioritize reduced emissions and fuel efficiency. The inputs include the selected area, which defines the region for route generation, and real-time traffic flow data, which helps determine the most efficient and environmentally friendly route.

2. Flow of Events

Basic Flow			
User		System	
1	User Selects Emission Friendly Routing – Planner		
		2	System switches planning mode for emission routing
		3	Calculates emission heavy zones and more emission friendly routes
		4	Highlights areas of high emissions which could be optimised by using more emission friendly routes.

Alternative Flow 1			
User		System	
Branch from step 3 of Basic Flow			
		3.1	Data sources unavailable
		3.2	Uses most recent historical data for calculations
		3.3	System display message “Using historical data due to current data unavailability”

3. Special Requirements

3.1. Platform

Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

System will rely on a number of Data sources

- Dublin linked open data API
- Weather API
- Air Quality API
- Maps API

4. Preconditions

- At least one data source is available
- API’s can be reached
- Historical data must be stored for fallback protection

5. Postconditions

- Emission friendly route overlayed on map.
- Emission heavy routes displayed.
- Alternative routes that would reduce emissions displayed.

2.6 Use Case Name: Find Quickest Route

1. Description

1.1 Goals and Responsibilities:

This feature will provide the quickest route between a start and end point. It will generate a series of routes that look to be the fastest at that time by considering distance, average speed, and traffic. On top of that, this use case must keep user data private and use the most up to date information when applicable.

1.2 Actors:

This will be used mainly by Emergency Responders who intend to find the quickest route between two destinations. In addition to these it will also be used by city planners by highlighting which routes are used the most thus receiving more wear. This information can later be used to implement new bus routes or alternative traffic management solutions to reduce congestion, travel time and use.

1.3 Triggers and Inputs:

This case will be used when the users opt to find the quickest route between two points by selecting the feature button. The user will be prompted to enter the start and end point.

2. Flow of Events

Basic Flow			
User		System	
1	User selects the routing feature		
		2	System loads and displays the maps page, prompting the user to input a start and end point
3	User inputs start and end point		
		4	System receives information and parameters
		5	System calculates the most time efficient route using distance, average speed, and traffic related data.
		6	System sends route to user
7	User receives data and displays the route		

Alternative Flow 1			
User		System	
		2.1	The system shows an alert to the user saying that the real-time data sources are not available and will use historical data
3.1	User inputs start and end point		
		4.1	System receives information and parameters
		5.1	System calculates most time efficient route with the historical data

3. Special Requirements

3.1 Platform

- Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

System will rely on a significant number of Data sources

- Dublin linked open data API
- Real time bus location
- Bus route map
- Weather API (for correlation of transport data and rainy days)

4. Preconditions

- At least one data source is available
- API’s can be reached
- Historical data must be stored for fallback protection

5. Postconditions

The user should have received and followed the quickest path from their start point to the destination. Said route will be curated using relevant data and in the case of new obstructions, it will be updated in real time.

2.7 Use Case Name: Efficient Routing

1. Description

1.1 Goals & Responsibilities:

To highlight inefficiencies in routing and help simulate new routes to find an optimum routing system.

1.2 Actors:

This will mainly be used by city council to plan how they use their limited resources in an efficient manner. It will help them spot inefficiencies and make it easier for them to better use their resources. This system can be used by other actors (e.g. Logistics companies, waste collectors etc.) to help plan routing for their own resources to make better use of them.

1.3 Triggers and Inputs:

This will be executed when using the software when the user navigates to the efficient routing page on the website. This will then display a UI prompting the user to input its current routes and times.

2. Flow of Events

Basic Flow			
User		System	
1	User navigates to the efficient routing page		
		2	System loads efficient routing page and displays to user and prompts user to input routes
3	User inputs routes		
4	User inputs estimated times for routing and data is sent off to the system		
		5	System receives information and parameters
		6	System simulates routing with expected traffic and highlights low speed zones and other forms of inefficiencies.
		7	System sends data back to user
8	User receives data and displays map form		
9	User selects if they want suggestions from system is, so it goes into alternative flow 2 if not it terminates.		

Alternative Flow 2			
User		System	
Branch from step 8 of basic flow			
8.1	User selects that they would like advice and sends this to the system		
		8.2	System receives request for advice
		8.3	System calculates alternative routes to that may increase the efficiency of the system
		8.4	System sends data back to user
8.5	User displays data received from system		

Alternative Flow 3			
User		System	
Branch from step 6			
		6.1	Traffic data not available
		6.2	System provides route calculations without traffic data

3. Special Requirements

3.1 Platform

Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

System will rely on a considerable number of Data sources

- Dublin linked open data API
- Maps API
- Bus route map
- Weather API (for correlation of transport data and rainy days)

4. Preconditions

- At least one data source is available
- API’s can be reached
- Historical data must be stored for fallback protection

5. Postconditions

The user should have received a map displaying inefficiencies to help in improving the efficiency of their system. If requested the user should have also received some advice or alternative routing suggestions.

2.8 Use Case Name: Check Bike Availability

1. Description

1.1 Goals & Responsibilities:

See the amount of Dublin Bikes available across the city at a given time and suggest how to redistribute them efficiently

.

1.2 Actors:

This will be mainly used by Environmental Bodies within the City Council. The City Council would be interested in this use case so that they can monitor the bikes to make sure they are returned and that there are enough of them throughout the city. As well as that, they would be eager to see if there is trend between the number of bikes used and a decrease in emissions.

1.3 Triggers and Inputs:

When users select the bike option, they can choose the area to get all bike information within that area.

2. Flow of Events

Basic Flow			
User		System	
1	User navigates to the Bike page		
		2	System loads maps page and displays to user and prompts user to enter location
3	User input's location		
		4	System receives information
		5	System finds bike racks and calculates bikes available
		6	System sends locations and bike availabilities back
7	User receives data and displays locations and figures on a map		

Alternative Flow 1			
User		System	
		2.1	The system shows an alert to the user saying that the data sources are not available and will use historical data
3.1	User input's location		
		4.1	System receives information
		5.1	System finds bike racks and availability based on the most recent historical data

3. Special Requirements

3.1. Platform

Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

- Dublin bike API
- Dublin city API

4. Preconditions

- At least one data source is available
- API’s can be reached
- Historical data must be stored for fallback protection

5. Postconditions

The user should have received the location of bike ranks within the city along with their availability at that time.

2.9 Use Case Name: Predict Traffic

1. Description

1.1 Goals & Responsibilities:

The aim of this use case is to predict heavy amount of traffic in certain areas at different times of the day. It could use historical knowledge for this and help other use cases such as ‘Find Quickest Route’ to aid them with their calculations.

1.2 Actors:

Mobility Service providers and Traffic Planners will be interested in using this case.

1.3 Triggers and Inputs:

This will be executed inherently by multiple use cases such as ‘Find Quickest Route’, ‘Tour Route’, ‘City Planning’ and ‘Data Analysis’. The inputs include desired start and end points, time of day and date.

2. Flow of Events

Basic Flow			
User		System	
1	User input’s location and end destination		
		2	The system will find quickest route
		3	The system will predict traffic levels along route
4	User receives quickest route with traffic considered		

Alternative Flow 1			
User		System	
Branch from step 2			
		2.1	System will find another quickest route if traffic is too heavy

Alternative Flow 2			
User		System	
Branch from step 3			
		3.1	Traffic data not available
		3.2	System will calculate quickest route without taking traffic into account
		3.3	System will notify user that data sources are not available

3. Special Requirements

3.1. Platform

Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

System will rely on a considerable number of Data sources such as:

- Dublin linked open data API
- Real time bus location
- Bus route map
- Weather API (for correlation of transport data and rainy days)
- Maps API

4. Preconditions

- At least one data source is available
- API's can be reached
- Historical data must be stored for fallback protection

5. Postconditions

This use case should be able to predict if there is a 'low', 'medium' or 'high' level traffic for specified areas at that time of day and pass on that information.

2.10 Use Case Name: Data Analysis

1. Description

1.1 Goals:

To combine, process and visualise data from several sources to identify long-term trends and correlations.

1.2 Actors:

Environmental Bodies and the City Council can use the analysis to monitor trends (emissions, traffic etc.) and make informed decisions based on these trends.

1.3 Triggers:

The user navigates to the data analysis page and selects which type of data they would like displayed and analysed. This will have options such as geographic area and time.

2. Flow of Events

Basic Flow			
User		System	
		1	The system displays data indicators at present
2	User picks data indicators, geographic area and the time for analysis		
3	User clicks Analyse button		
		4	System retrieves data from multiples sources in JSON, XML, etc
		5	System processes data into uniform format
		6	System analyses correlations between indicators
		7	System identifies and highlights underserved areas
		8	System displays recommendations for improved routes
9	User Exports analysis report		

Alternative Flow 1			
User		System	
Branch from step 5 of Basic Flow			
		5.1	System detects that a data source is not available
		5.2	System switches to displaying historical information
		6	Overlays historical information with current data from working data sources
		7	System analyses correlations between indicators
		8	System identifies and highlights underserved areas
		9	System displays recommendations for improved routes
10	User exports analysis report		

3. Special Requirements

3.1. Platform

Accessibility on Major modern browsers (Firefox, Chrome, Safari).

3.2 External Dependencies

System will rely on a considerable number of Data sources

- Dublin linked open data API
- Real time bus location
- Dublin bikes API DCC
- Pedestrian count Api
- Bus route map
- Air Quality API
- Weather API (for correlation of transport data and rainy days)
- Maps API

4. Preconditions

- At least one data source is available
- API’s can be reached
- Historical data must be stored for fallback protection

5. Postconditions

A Visual analysis of the data based on what the user selected
The user can export the analysis in a standard format.