Use cases

1. Visitor registration

|  |  |
| --- | --- |
| Actors | Visitor |
| Goals | [G1] |
| Input conditions | 1. The Visitor has an Internet access. 2. The Visitor is on the home page. |
| Events flow | 1. The Visitor clicks on the “Sign in” button on the home page to start the registration process. 2. The Visitor fills all mandatory fields. 3. The Visitor clicks on the “Confirm” button. 4. The system saves the data. 5. The system sends a SMS to the new User with the password. |
| Output conditions | The Visitor ends the registration process successfully and become a new User. From now on he/she can log in to the application providing his/her credentials. |
| Exceptions | 1. The Visitor is already registered. 2. The Visitor inputs incorrect data in one or more mandatory fields. 3. The Visitor takes a username that has already been associated with another User. 4. The Visitor chooses an email that has already been in the system.   All exceptions are handled with notifying the issue to the Visitor and taking back to the point 2 of Events Flow. |

2)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G1] |
| Input conditions | The User is on the home page. |
| Events flow | 1. The User inputs his/her credentials into the “Username” and “Password” fields 2. The User clicks on the “Log in” button to get access. 3. The system redirects the User to his/her personal area. |
| Output conditions | The User gets access to his/her personal area successfully. |
| Exceptions | 1. The User inputs invalid Username. 2. The User inputs invalid Password.   All exceptions are handled with notifying the issue to the User and taking back to the point 2 of Events Flow.   1. The User inputs invalid Password 5 times.   The exception is handled with notifying the issue the User and offering to change his/her password. |

\item (G02): The system allows User to set his/her route inside a city or a region

3)

|  |  |
| --- | --- |
| Actors | Client |
| Goals | [G2] |
| Input conditions | The client has already logged in. |
| Events flow | 1. The Client sets up the initial point using GPS localization. 2. The Client sets up the arrival point using GPS localization or specified address. 3. The Client checks if his/her current position and the arrival point has been correctly defined. 4. The Client clicks on the “Confirm” button. 5. The system builds the route. 6. The system offers the Client the ways in order of increasing their length. |
| Output conditions | The Client gets the set of routes to reach his/her destination. |
| Exceptions | 1. GPS is out of work. 2. The Client inputs the coordinates of initial and arrival point in different regions. 3. The Client’s current position cannot be defined correctly. 4. It’s impossible to build the route inside the city/region.     All exceptions are handled with notifying the issue to the Client and taking back to the point 1 of Events Flow. |

\item (G03): The system allows a User to choose a kind of transport among pre-defined travel means according to his/her preferences

4)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G3] |
| Input conditions | The User has the set of paths to reach his/her destination. |
| Events flow | 1. The User puts flags near his/her preferable travel means. 2. The User clicks on the “Confirm” button to save his/her choice. 3. The system shows the User the list of available routes according to   his/her preferences |
| Output conditions | The User gets the set of the preferable routes |
| Exceptions | 1. There is no User’s preferable transport.   This exception is handled redirecting the User to the page of Technical support.   1. User has not chosen any preferable transport.   All exceptions are handled with notifying the issue to the Client and taking back to the point 1 of Events Flow. |

\item (G04): Every user can decide the range of time to reserve for breaks.

\item (G13): The system must avoid overlaps in user's scheduled travels.

5)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G4] [G13] |
| Input conditions | The User has already logged in. |
| Events flow | 1. The system shows the User schedule of his/her meetings. 2. The User chooses a period of time between meetings. 3. The User clicks on the “Create a break” button. 4. The User sets up his/her break. 5. The User sets up the durability of his/her break. 6. The User clicks on “Confirm” button. |
| Output conditions | The User gets the schedule of his/her meetings. |
| Exceptions | 1. The User chooses invalid range of time given for a meeting. 2. The User does not choose the range of time for the break.   All exceptions are handled with notifying the issue to the Client and taking back to the point 2 of Events Flow.   1. The User does not input any data for the break.   The exception is handled with notifying the issue to the Client and offering to cancel the operation or take back to the point 4 of Events Flow.   1. The durability of the User’s break is too long and cross with the period of other meetings. 2. The User does not set up the durability of his/her break.   The exception 4 and 5 are handled with notifying the issue to the User and taking back to the point 5 of Events Flow. |

\item (G10): The system must provide a way to permit to a single user to buy a ticket for public transports.

6)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G10] |
| Input conditions | The User chooses the route with using public transport except taxi |
| Events flow | 1. The system offers the User to buy a ticket for his/her trip. 2. The User clicks on the “Confirm” button. 3. The system redirects the User to ATM site (“buy ticket” page) |
| Output conditions | The User buys the ticket on ATM site. |
| Exceptions | 1. The User does not want to buy a ticket   The exception is handled with notifying about the issue and cancelling the operation.   1. ATM site does not work.   The exception is handled with notifying and redirecting to the Technical Support |

(G15): The system must inform the user about upcoming meetings.

7)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G15] |
| Input conditions | The User has a meeting in his/her schedule. |
| Events flow | 1. The system computes the path time. 2. The system sets up the timer to alert the User about upcoming meeting in the required time. 3. The timer works. 4. The user switches off the timer. |
| Output conditions | The User knows about the upcoming meeting. |
| Exceptions | 1. The User does not hear the alarm.   The exception is handled with re-alerting the User in 1 minute. |

Activating the route

8)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G] |
| Input conditions | The User chooses a route. |
| Events flow | 1. The User clicks on “Let’s go!” button. 2. The system activates the route. 3. The system activates the tracing of the User’s location. |
| Output conditions | The User receives actual information about his/her motion through the route. |
| Exceptions | 1. The User does not want to go and clicks on “Cancel” button.   The exception is handled with notifying about the issue and taking back to the home page. |

\item (G08): The system must provide information about problems/strikes for all the travel means included in the software.

\item (G17): The system must provide an alternative path in case of problems along the selected one.

9)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G08] |
| Input conditions | The User has the set of possible paths |
| Events flow | 1. The system receives information about the strike from ATM and Trenord sites. 2. The system blocks the participating public transport. 3. The User gets a notification about the strike. 4. The system shows the User available travel means. 5. The system asks the User about the continuing of the operation. 6. The User clicks on “Confirm” button. 7. The User chooses preferable travel means. 8. The User clicks on “OK” button. |
| Output conditions | The User has a set of preferable routes. |
| Exceptions | 1. The User chooses blocked travel means.   The exception is handled with notifying about the issue to the User and taking back to the point 4 of Events Flow.   1. The User does not want to continue the operation.   The exception is handled with notifying about the issue to the User and taking back to the home page. |

About dangerous zone

10)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G08] |
| Input conditions | The User has the set of possible paths |
| Events flow | 1. The system compares the paths with the pre-defined set of risk zones. 2. If the piece of the path corresponds to one of the risk zone, the system sends the User a notification. 3. User clicks on “I understand!” button. 4. The system shows the User available travel means in order of increasing a risk. 5. The system asks the User about the continuing of the operation. 6. The User clicks on “Confirm” button. 7. The User chooses a preferable travel mean. 8. The User clicks on “OK” button. |
| Output conditions | The User has a preferable route. |
| Exceptions | 1. There are no risk zones in the path.   The exception is handled with redirecting the User to Use Case 7   1. The User does not want to go through the risk zone.   The exception is handled with notifying about the issue to the User and taking back to the Use case 10 |

Alternative route

11)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G] |
| Input conditions | The User has a dangerous path.  Exception 2 of the Use case 8 |
| Events flow | 1. The system computes a path without the risk zone. 2. The system shows the User a list of secure paths. 3. The User chooses a preferable path. 4. The User clicks on “Confirm” button. |
| Output conditions | The User has a preferable path |
| Exceptions | 1. The User does not find a preferable path.   The exception is handled with notifying the User about cancellation of the operation and taking back to the home page.   1. There are no paths without the risk zone.   The exception is handled with notifying the User about the issue and taking back to the Use Case 7 |

\item (G11): The system must provide the nearest location of a bike provided by a pre-defined bike sharing service provider.

12)

|  |  |
| --- | --- |
| Actors | User  Technical support |
| Goals | [G11] |
| Input conditions | The User has a set of preferable routes. |
| Events flow | 1. The User chooses a bike-sharing route. 2. The system redirects the User to the map. 3. The system shows the nearest locations of bike-sharing. 4. The User chooses a preferable location. 5. The system shows the User information about a bike unlocking. 6. The User clicks on the “I understand!” button. 7. The system redirects the User to Mobike site. |
| Output conditions | The User rents the bike on Mobike site. |
| Exceptions | 1. The User does not find a preferable location.   The exception is handled with notifying the User about cancellation of the operation and taking back to the Use case 4.   1. Mobike site does not work.   The exception is handled with notifying about the issue to the Technical Support and the User and redirecting the User to the Technical support page.   1. The User does not like the way how to unlock the bike.   The exception is handled with the User’s clicking on the button “I don’t want” and taking back to the Use case 4. |

\item (G16): The system must allow the user to change the part of the path during his/her trip.

13)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G16] |
| Input conditions | The route is activated. |
| Events flow | 1. The User clicks on “Change the route” button. 2. The system asks the User to confirm the pause. 3. The User clicks on “Confirm” button. 4. The system stops the route. 5. The system asks the User what he/she wants to change.    * 1. If the User wants to change the meeting location, he/she inputs the necessary address in the field.      2. The User clicks on “Confirm” button.      3. The system computes the path.      4. The User puts a flag near a preferable travel mean in the list of available preferable travel means.    1. 1. If the User wants to change a travel mean, he/she puts a flag near a travel mean in the list of available preferable travel means.       1. The User clicks on “Confirm” button. 6. The User clicks on “Let’s go” button. 7. The system activates a route. |
| Output conditions | The User activates the changed route. |
| Exceptions | 1. The User wants to take a break and he/she clicks on “Create a break” button.   The exception is handled with redirecting to the Use Case 4.   1. The User inputs the wrong address.   The exception is handled with notifying about the issue to the User and taking back to point 5.1.1.   1. The User wants to stop the route.   The exception is handled with clicking on “Cancel” button, notifying about the issue to the User and taking back to the home page. |

Car-sharing

14)

|  |  |
| --- | --- |
| Actors | User  Technical support |
| Goals | [G] |
| Input conditions | The User has a set of preferable routes. |
| Events flow | 1. The User chooses a car-sharing route. 2. The system redirects the User to the map. 3. The system shows the nearest locations of car-sharing. 4. The User chooses a preferable location. 5. The system shows the User information about a car unlocking. 6. The User clicks on the “I understand!” button. 7. The system redirects the User to Car2go site. |
| Output conditions | The User rents the car on Car2go site. |
| Exceptions | 1. The User does not find a preferable location.   The exception is handled with notifying the User about cancellation of the operation and taking back to the Use case 4.   1. Car2go site does not work.   The exception is handled with notifying about the issue to the Technical Support and the User and redirecting the User to the Technical support page.   1. The User does not like the way how to unlock the car.   The exception is handled with the User’s clicking on the button “I don’t want” and taking back to the Use case 4. |

Using the means which are not included to the application

15)

|  |  |
| --- | --- |
| Actors | User  Technical support |
| Goals | [G11] |
| Input conditions | The User has a set of paths. |
| Events flow | 1. The User clicks on “Choose travel mean” button. 2. The User clicks on “More” button. 3. The User chooses a preferable transport in the list of additional shared travel means. 4. The system shows a corresponding link of the necessary site. 5. The User clicks on the link. |
| Output conditions | The User is redirected to the necessary sharing site. |
| Exceptions | 1. There is no the necessary travel mean.   The exception is handled with the User clicking on the “I want more!” button and redirecting him/her to the Technical support page.   1. The link does not work.   The exception is handled with notifying about the issue to the Technical Support and the User and redirecting the User to the Technical support page. |

own vehicle

16)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G] |
| Input conditions | The User chooses a “Car” option in the list of available travel means |
| Events flow | 1. The system asks the User about having a private car. 2. The User clicks on “Confirm” button. 3. The system asks to remember his/her decision. 4. The User clicks on “Allow” button. |
| Output conditions | The User has a route by his/her own car. |
| Exceptions | 1. The User does not have a private car.   The exception is handled with offering the User “taxi” or “car-sharing” option.   1. The User does not want the system to remember his/her choice.   The exception is handled with clicking on “Don’t allow” button. |

unreachable destination

17)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G] |
| Input conditions | The User creates a meeting. |
| Events flow | 1. The system computes the shortest required time to reach the destination. 2. The system compares the shortest required time and the time left for the meeting. 3. If the time the User has is less than the shortest required time, the system notifies the User about the lack of time. |
| Output conditions | The User has a warning about unreachable time. |
| Exceptions |  |

fix problems with redirecting to external company

18)

|  |  |
| --- | --- |
| Actors | Technical support |
| Goals | [G] |
| Input conditions | 1. The technical support has logged in. 2. The Technical support receives the notification about the issue with redirecting to a site. |
| Events flow | 1. The Technical support sends a query to the database with additional links to similar sites without API providing. 2. The Technical support selects a link of a corresponding site. 3. The Technical support replaces broken link with a page with apologizes and the link. |
| Output conditions | The system has a working link. |
| Exceptions |  |

carbon footprint

19)

|  |  |
| --- | --- |
| Actors | User |
| Goals | [G] |
| Input conditions | The User has a set of preferable routes. |
| Events flow | 1. The system selects the route by foot, by bike or by using bike-sharing service. 2. The system defines these routes as the most ecological route. 3. The system offers the User the most ecological route. 4. The User clicks on “I choose this route!” |
| Output conditions | The User has a preferable route. |
| Exceptions | 1. The User does not want to use the most ecological route.   The exclusion is handled with User clicking on “I don’t want” button and showing the list of the other possible routes. |

weather conditions

20)

|  |  |
| --- | --- |
| Actors | User  Weather forecast site (meteo.it) |
| Goals | [G] |
| Input conditions | The User has a set of preferable routes. |
| Events flow | 1. The system addresses a query of the nearest weather forecast to Weather forecast site providing us API. 2. The system gets the information about the nearest weather forecast 3. If it rains or snows during the User’s trip, the system blocks unsuitable travel means (bike-sharing, bike, walking). 4. The system warns the User about the bad weather conditions. 5. The User clicks on “OK” button. 6. The User selects a preferable travel mean in the list of available travel means. |
| Output conditions | The User has a preferable route. |
| Exceptions | 1. The User does not want to go to the meeting when the weather is bad.   The exclusion is handled with User clicking on “Cancel trip”, confirming his/her decision and taking back to the home page. |