TRAVLENDAR+

RASD

Requirement Analysis and Specification Document

Kostandin Caushi 898749

Marcello Bertolini 827436

Raffaele Bongo 900090

Date 29/10/2017

Version 1

Table of Contents

1 Introduction **2**
1.1 Purpose 2
1.2 Scope 2
1.3 Definitions 3
1.4 Acronyms 4
1.5 Abbreviations 4
1.6 Goals 4
1.7 Reference Documents 4
1.8 Document Structure 5

2 Overall Description **6**
2.1 Product Perspective 6
2.2 Product Functions 7
2.2.1 Registration & Login 7
2.2.2 Set Preferences 8
2.2.3 Guarantee at Least 30 Minutes of Lunch Break 9
2.2.4 Add Events to the Calendar 9
2.2.5 Give Advices About the Means of Transport 9
2.2.6 The Local Transport Tickets : Recommendation and Purchasing 10
2.2.7 Trip Management 10
2.3 User Characteristics 11
2.4 Domanin Properties 11

3 Specific Requirements **13**
3.1 Proposed System 13
3.2 External Interface Requirements 13
3.2.1 User Interfaces 13
3.3 Functional Requirements 13

4 Scenarios **17**
4.1 Scenario 1 17
4.2 Scenario 2 17
4.3 Scenario 3 17
4.4 Scenario 4 18
4.5 Scenario 5 18
4.6 Scenario 6 18

1 Introduction

1.1 Purpose

This document represents the Requirement Analysis and Specification Document (RASD). All the goals of the application, the functional and non-functional requirements for achieving them are here reported.

There are also the use cases of the system and various scenarios that represent the examples of a real life system's utilization. This document is addressed to the developers who have to implement the application's requirements and it also could be used as a contractual basis.

1.2 Scope

The system that we will expose in this paper is called Travlendar+. Travlendar+ is a calendar-based application that has the aim of managing the many meetings, events and appointments that a user has to deal with every day.

After the registration and the login, the system will let the user create events in his personal calendar, checking if he is able to reach them on time and supporting his choices about the way of reaching the location. In fact, the user will be able to insert customizable settings which will allow the system to give him back precise advices about the means of transport, including taxis and sharing services means, to use.

Travlendar+ will also give the user the possibility to buy tickets of a town's public means of transport and it will also allow him to manage his travels to reach other cities, creating specific travel events in the calendar section. The system will offer other additional features :

- The possibility to register the season ticket for the public transport. Travlendar+ will notify the user when the expiration date is near.
- The possibility to set the starting time, ending time and the preferred duration of every day lunch. The system will guarantee to preserve at least 30 minutes for this purpose.
- In case of outdoor trips, the user will be able to insert the period he will spend out of town and the system will suggest him the most convenient public transport tickets available keeping in mind the information given.
- The possibility of setting the anticipation time for reaching the various events. The system will warn the user when he needs to leave in order to arrive on time.

1.3 Definitions

- ***System :*** the software and hardware components that characterize Travlendar's environment.
- ***Outdoor :*** a different location from the user's residence town.
- ***Outdoor Transport Service :*** the services dedicated to the transport between different towns.
- ***Local Public Transport Service :*** the services dedicated to the transport within a town.
- ***Event :*** a generic word used for speaking about appointments, meetings, etc. added to the calendar.
- ***Event Path :*** a part of an event solution that the user can perform with a specific mean of transport.
- ***Event Solution :*** a set of event path that allow the user to reach the event location.
- ***Warning :*** the method used by system to warn the user about something.
- ***Best means of transport options :*** the best transport found by the system according to the user's preferences.
- ***Preferences :*** a set of options that modify the behaviour of the system when computing the routes.
- ***Sharing Means :*** bike or car sharing.
- ***User :*** the person who has performed a registration and is logged in the system.
- ***Anticipation Time :*** it specifies how much time before the beginning of the event the user wants to arrive with.
- ***TAG :*** it's a collection of information, created by the user, which can be associated to an event and contains the preferred means of transport and anticipation time to use.
- ***Residence :*** the user's home address.
- ***Accommodation :*** the user's occasional accommodation address located in a town different from the one reported in his profile.
- ***Trip :*** a set of information about the journey that the user has organized.
- ***Travel event :*** an event that contains all the information about the travel between two different towns such as start time, end time and the mean of transport involved.

1.4 Acronyms

- *RASD :* Requirement Analysis and Specification Document
- *API :* Application Programming Interface

1.5 Abbreviations

- [Gn] : Goal n
- [Rn] : Requirement n
- [Dn] : Domain n

1.6 Goals

- [G_1] Allow the user to register and to log into the system.
- [G_2] Allow user to add events in the calendar.
- [G_3] Allow the user to receive the mobility options.
- [G_4] Support the user to avoid getting late on appointment.
- [G_5] Allow the user to have advices about the means of transport that can minimize his carbon footprint.
- [G_6] Support the user to have at least 30 minutes of lunch every day.
- [G_7] Allow the user to buy local public transport ticket.
- [G_8] Give advice about the best public transportation ticket to buy.
- [G_9] Remember the user about the expiry date of his season ticket, if he has inserted one.
- [G_{10}] Allow the user to buy ticket for outdoor travels.
- [G_{11}] Allow the user to use local sharing services.
- [G_{12}] Allow the user to set the preferences in the settings section.
- [G_{13}] Allow the user to handle his trips.

1.7 Reference Documents

- Mandatory Project Assignments.pdf
- Requirement+Engineering+Part+III.pdf
- IEEE standard on requirement engineering.pdf
- Paper by Jackson and Zave on the word and the machine.pdf
- Software Abstractions -Logic Language and Analysis.
  Author: Daniel Jackson

1.8 Document Structure

This paper is composed by 5 chapters :

1. The first chapter is constituted by an introduction of the system, his application domain, his goals and a glossary with the most common expression used in order to give to the reader a basic knowledge of the system and to make him understand better the subsequent parts.
2. The second part consists of an overall description of the system and the main functionalities, which are listed and described. It also describes the relation of the system with the external services and how they interact. The constraints that have to be respected in order to use the system properly are also explained. Moreover, a classification of the system's actors and the domain properties are given to the reader.
3. The third part is composed by the specific requirements identified, both functional and non-functional, some mockups that show the most important features and how the user will be able to interact with the system.
4. The fourth part is composed by 6 scenarios that describe how the system will work and how it will perform his functionalities.
5. The fifth and final part is composed entirely by Alloy code and some snapshot of the world generated with the related tool.

2 Overall Description

2.1 Product Perspective

The system for achieving its goal, it will be related with three kind of external components :

- Root providers (Google maps)
- Local transport services
- Outdoor transport services

To localize the means and the user's position and to retrieve the directions, we will use in our system Google maps API. The system will format the data received in a standardized manner and show them to the user according to his preferences.

Our system will contain all the references to the local and outdoor transport services, which APIs will be known in order to retrieve the requested information about the tickets' prices. Once the user will have chosen the ticket, he will be addressed to the service's website to complete the purchase.

At first the system will advise about the best local public transport tickets only in the towns of Milano, Torino and Rome; for the other towns, the system will give the user the possibility to access to the local transport websites. Regarding the outdoor travels, it will be possible to buy ticket only with a restricted group of transport services such as Trenitalia and Alitalia.

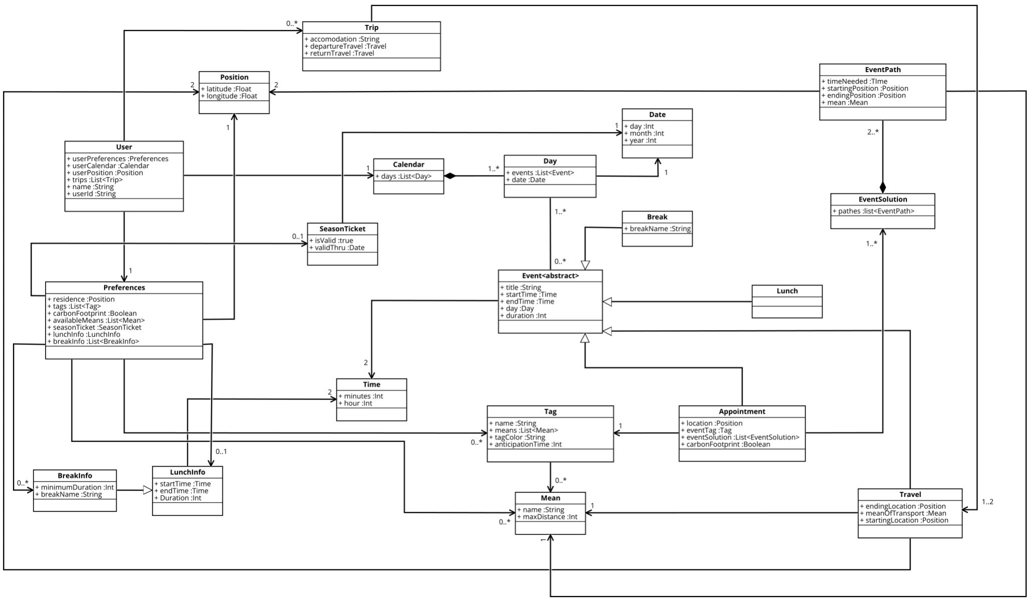
Trip
+ accomodation :String
+ departureTravel :Travel
+ returnTravel :Travel
0..*
Position
+ latitude :Float
+ longitude :Float
2
2
1
User
+ userPreferences :Preferences
+ userCalendar :Calendar
+ userPosition :Position
+ trips :List<Trip>
+ name :String
+ userId :String
Calendar
+ days :List<Day>
1
Day
+ events :List<Event>
+ date :Date
1..*
Date
+ day :Int
+ month :Int
+ year :Int
1
1
EventPath
+ timeNeeded :TIme
+ startingPosition :Position
+ endingPosition :Position
+ mean :Mean
2..*
EventSolution
+ pathes :list<EventPath>
1..*
SeasonTicket
+ isValid :true
+ validThru :Date
0..1
Break
+ breakName :String
1..*
0..*
Event<abstract>
+ title :String
+ startTime :Time
+ endTime :Time
+ day :Day
+ duration :Int
Lunch
Preferences
+ residence :Position
+ tags :List<Tag>
+ carbonFootprint :Boolean
+ availableMeans :List<Mean>
+ seasonTicket :SeasonTicket
+ lunchInfo :LunchInfo
+ breakInfo :List<BreakInfo>
1
Time
+ minutes :Int
+ hour :Int
2
2
Tag
+ name :String
+ means :List<Mean>
+ tagColor :String
+ anticipationTime :Int
0..*
1
Appointment
+ location :Position
+ eventTag :Tag
+ eventSolution :List<EventSolution>
+ carbonFootprint :Boolean
0..1
BreakInfo
+ minimumDuration :Int
+ breakName :String
0..*
LunchInfo
+ startTime :Time
+ endTime :Time
+ Duration :Int
0..*
Mean
+ name :String
+ maxDistance :Int
0..*
1
Travel
+ endingLocation :Position
+ meanOfTransport :Mean
+ startingLocation :Position
1..2

Figure 1: Class Diagram

2.2 Product Functions

In order to provide a clear vision of the system's aims that will be implemented, we are going to show some use cases diagram representing the system's features. A description of the main ones will be provided.

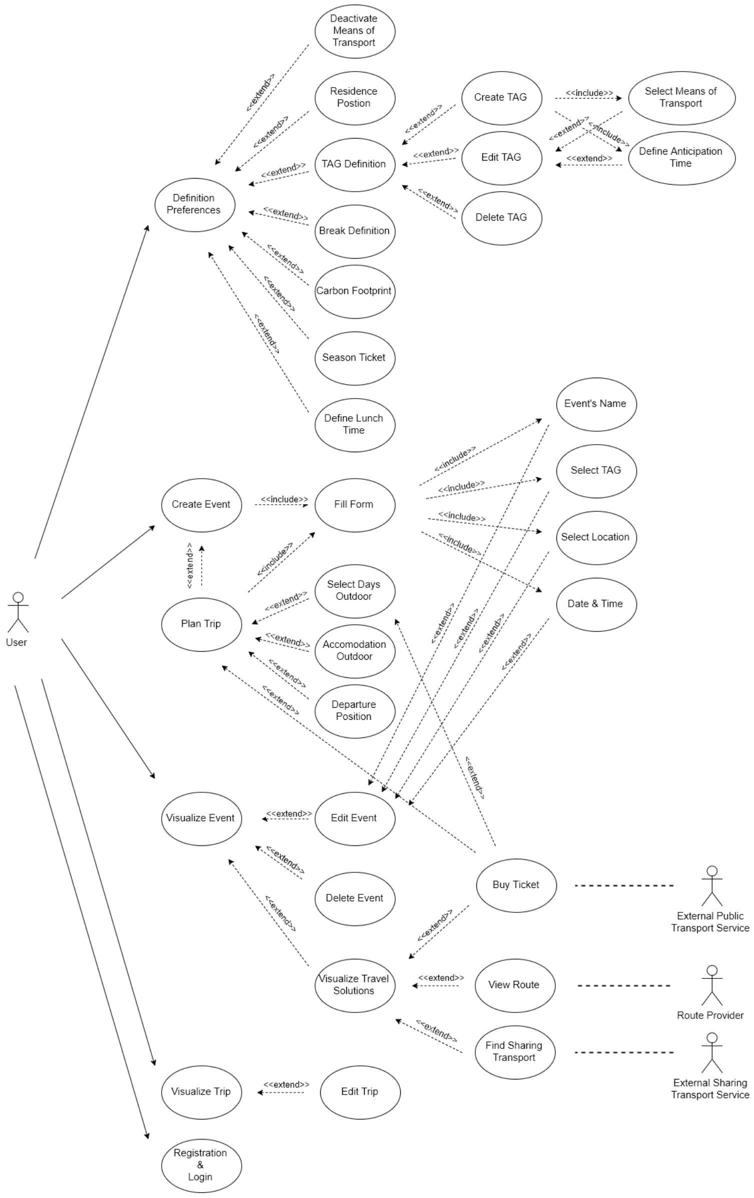
Deactivate Means of Transport
Residence Postion
Create TAG
<<include>>
Select Means of Transport
<<extend>>
Edit TAG
Define Anticipation Time
TAG Definition
Delete TAG
Definition Preferences
Break Definition
Carbon Footprint
Season Ticket
Define Lunch Time
Event's Name
Select TAG
Select Location
Date & Time
Create Event
Fill Form
Plan Trip
Select Days Outdoor
Accomodation Outdoor
Departure Position
User
Visualize Event
Edit Event
Delete Event
Buy Ticket
External Public Transport Service
Visualize Travel Solutions
View Route
Route Provider
Find Sharing Transport
External Sharing Transport Service
Visualize Trip
Edit Trip
Registration & Login

Figure 2: General Use Case

2.2.1 Registration & Login

The system will let the user to sign up, inserting the e-mail and password for creating his personal profile and he will be able to sign-in every time he wants.

2.2.2 Set Preferences

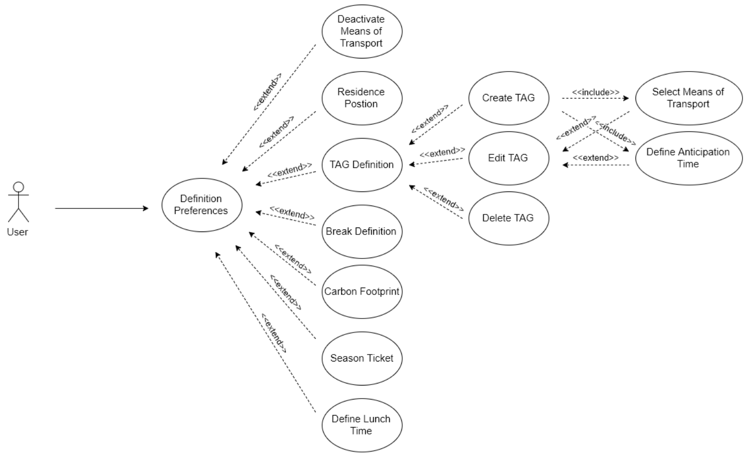
Deactivate Means of Transport
Residence Postion
TAG Definition
Break Definition
Carbon Footprint
Season Ticket
Define Lunch Time
Definition Preferences
User
Create TAG
Edit TAG
Delete TAG
Select Means of Transport
Define Anticipation Time
<<extend>>
<<include>>

Figure 3: Set Preferences Use Case

The user will be able to set many preferences to customize his profile. He will be able to :

- insert his season ticket and receives warnings to remember him the expiration date.
- set his home address that will allow the system to compute the routes for reaching the first event of the day using it as the starting point.
- define the means of transport that he never wants to use.
- minimize his carbon footprint.
- set the constrains about the maximum distance that the user wants to cover with a precise mean of transport.
- create and define TAGs.
- set the default lunch time window and duration. The duration must be equal or greater than 30 minutes.
- define a break time window and its duration as in the lunch preference but with the possibility of setting also a minimum duration time.

A TAG will be used to define the type of the event. It will contain the list of means of transports and the anticipation time for reaching the event associated with.

2.2.3 Guarantee at Least 30 Minutes of Lunch Break

Each lunch event is created automatically considering the time window and duration set in the preference section.

The system will guarantee at least 30 minutes of lunch per day. In fact, every time that the user will try to create an event that overlaps the lunch event, the system will check if it is possible to move the lunch event, keeping the same duration, in the time window defined. If not, the system will check if it's possible to reduce the default lunch duration guaranteeing the 30 minutes' constraint. In this case, the system will ask the user if he wants to confirm the changes proposed, if not, it will not be possible to add the new event.

2.2.4 Add Events to the Calendar

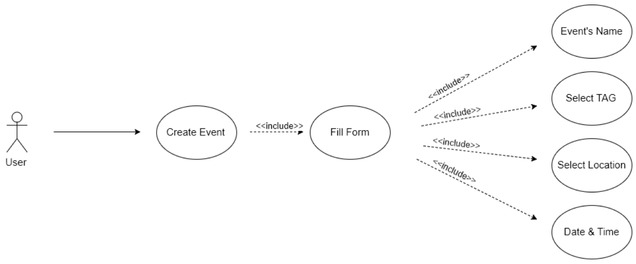
Event's Name
Select TAG
Select Location
Date & Time
User
Create Event
<<include>>
Fill Form
<<include>>
<<include>>
<<include>>
<<include>>

Figure 4: Add Event Use Case

The system will let the user add a new event to the calendar. The user will have to set the title, the date and time, the carbon footprint preference and must associate the event with a TAG. Once done, the system will check if the event is reachable by the previous one and if there's not overlapping. If the two previous conditions are confirmed the new event will appear in the calendar.

2.2.5 Give Advices About the Means of Transport

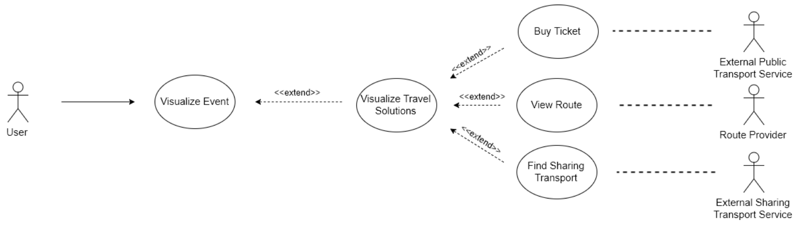
Buy Ticket
External Public Transport Service
<<extend>>
User
Visualize Event
<<extend>>
Visualize Travel Solutions
<<extend>>
View Route
Route Provider
<<extend>>
Find Sharing Transport
External Sharing Transport Service

Figure 5: Advice Mean, Sharing and Ticket Purchase Use Case

The system will show the user the advices about the best means of transport for reaching every event that the he has to attend. The routes will be calculated using as starting point :

- the previous event as the default option.

- the home address/outdoor accommodation in the case the event to reach is the first of the day.
- the user position.

In any case it will be possible to show the directions of every event using as starting point his position just tapping on a specific button in the map interface.

The advices will follow the user's preferences described in the TAG related to the event, keeping into account also the weather conditions and giving him warnings in the presence of strikes. Moreover, it will be given a warning when the user needs to leave in order to reach the event on time.

The system will also localize and show the position of the nearest car and bike sharing services.

2.2.6 The Local Transport Tickets : Recommendation and Purchasing

The system will recommend and allow to buy the best public transport ticket. This feature is available just in the towns reported in the previous section.

Furthermore, during the trip management, after have inserted the accommodation address and the duration, the system will give back the user the tickets available, their prices and the best option that covers the entire period. The user will be able to purchase them after being addressed to the local service website/application.

2.2.7 Trip Management

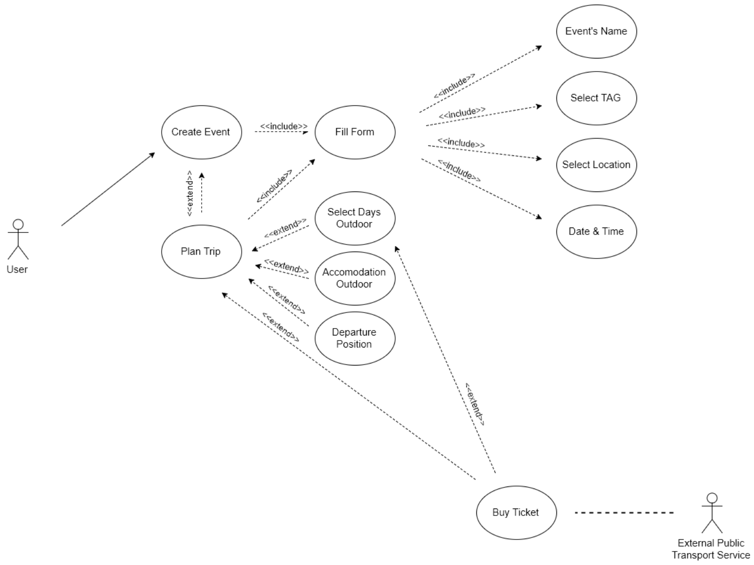
Event's Name
Select TAG
Select Location
Date & Time
Create Event
<<include>>
Fill Form
<<include>>
<<include>>
<<include>>
<<include>>
<<extend>>
<<include>>
User
Plan Trip
Select Days Outdoor
<<extend>>
Accomodation Outdoor
<<extend>>
Departure Position
<<extend>>
<<extend>>
<<extend>>
Buy Ticket
External Public Transport Service

Figure 6: Trip Management and Ticket Purchase Use Case

The system will allow the user to manage trips and, by defining the accommodation and the duration, it will suggest the tickets available for reaching the desired town. It will be possible to arrange a trip directly from the calendar section. Furthermore, when an event is crated and it's located in a town different from the user's position, the trip's arrangement will be prompted.

The travel will be represented as a special event in the calendar's section that covers the entire travel time and contains all the information. The user will be able to edit the trip information any time he wants accessing into the trips section.

2.3 User Characteristics

The system is suitable for any kind of user. In fact, the Travlendar+ goals are appropriate for any person who needs help for taking on the everyday organization.

Actors

- ***User :*** is a person who has done the subscription in the system. Every user, after the login process, is able to use all the system's functionalities.
- ***System Manager Team :*** it's a group of engineers and programmers in which the founders have entrust the maintenance of the system and its upgrade for answering promptly to the users' needs.
- ***Client Service :*** it is a small team that has the aim of receiving and resolving users' complaints and requests.

2.4 Domanin Properties

1. User id must be unique.
2. GPS position is always correct.
3. GPS cannot be switched off.
4. Event creation implies the user's participation to it.
5. Events do not overlap with each other.
6. Every event in the calendar is reachable from the previous one.
7. The duration of the event in world corresponds to the duration of the one in the calendar
8. Only the user can create, edit and delete the event on his calendar.
9. Public transports respect the timetables.
10. The means of transport suggested are actual available.
11. Every user has one calendar.
12. External transport services always answer to the system requests.

13. The user has to create the event before it starts.
14. Subways, Trams and bikes do not emit CO2.
15. Each transport mean belongs to one transport service.
16. User can be just in one position at a time.
17. The weather forecasts are always reliable.
18. External services answer to the system request in no more than 5 seconds.
19. The user who asks to purchase a ticket on the system, will actually buy the ticket in the external service of transport's system
20. The user has to follow at least one of the warnings send from the system.
21. Public transports are composed by: bus, subway, tram, train

3 Specific Requirements

3.1 Proposed System

3.2 External Interface Requirements

3.2.1 User Interfaces

The following mockups represent an idea of the interfaces that will be provided to the user.

3.3 Functional Requirements

[G_1] : Allow the user to register and log into the system

- [R_1] : The System has to check the credentials.
- [R_2] : A registered user must be able to log in the system.

[G_2] : Allow the user to add events in the calendar

- [R_1] : The system mustn't allow events overlapping.
- [R_2] : The system has to guarantee the minimum lunch duration.
- [R_3] : The system has to give the possibility to the user of arriving on time to the new event.
- [R_4] : The system has to be able to add the new event to the calendar.

[G_3] : Allow user to receive the mobility options

- [R_1] : The system has to check that the event has been created.
- [R_2] : The system is able to communicate with the route provider.
- [R_3] : The system sends data about the starting and ending location to the route provider.
- [R_4] : The system receives the path and the available transport from the external service.
- [R_5] : The system has to use the information given in the event's TAG to filter the possible means of transport.
- [R_6] : The system has to check the travel means enabled by the user.
- [R_7] : The system has to obey the constraints set by the user about the travel means.
- [R_8] : The system has to check if the Carbon footprint preference has been enabled.

- [R_9] : The system has to retrieve user position.
- [R_{10}] : The system has to check the weather forecast.
- [R_{11}] : The system has to retrieve the appointment location.
- [R_{12}] : The system has to know how to show the solution found.

[G_4] : Support the user to avoid getting late on appointment

- [R_1] : The system has to retrieve the user GPS position.
- [R_2] : The system has to know how to send warnings to the user.
- [R_3] : The system has to check the destination position.
- [R_4] : The system has to check the possibility of reaching on time the next event with the slowest mean of transport in the list of suggested ones.

[G_5] : Allow the user to have advices about the means of transport that can minimize his carbon footprint

- [R_1] : The system must be able to check the carbon footprint preference.
- [R_2] : The system must know how to filter the means of transport to minimize their carbon footprint.

[G_6] : Support the user to have at least 30 minutes of lunch every day

- [R_1] : The system must be able to check the lunch preferences.
- [R_2] : The system must be able to create every day a lunch based on the user preferences.
- [R_3] : The system must be able to avoid the creation of those events that prevent the presence of a lunch with the minimum duration.

[G_7] : Allow the user to buy local transport ticket

- [R_1] : The system has to check that all the necessary travel info have been inserted by the user.
- [R_2] : The system has to address the user to the web site/application to complete the ticket purchase.

[G_8] : Give advice about the best public transportation ticket to buy

- [R_1] : The system has to know how much time the user will stay in town.

- [R_2] : The system has to be able to get information about tickets from the public transport service.
- [R_3] : The system has to be able to interpret and elaborate the information received.
- [R_4] : The system has to find the most convenient ticket.
- [R_5] : The system has to be able to show the results.

[G_9] : Remember the user about the expiry date of his season ticket, if he has inserted one

- [R_1] : The system has to check the season ticket's expiry date.
- [R_2] : The system has to be able to notify the user that his season ticket is expiring.

[G_{10}] : Allow the user to buy ticket for outdoor travels

- [R_1] : The system must check the data inserted by the user to buy the ticket.
- [R_2] : The system must be able to send the data to the external transport service.
- [R_3] : The system has to be able to interpret and elaborate the information received.

[G_{11}] : Allow user to use local sharing services

- [R_1] : The system has to retrieve the sharing means' location.
- [R_2] : The system has to check the user location.
- [R_3] : The system has to be able to show the sharing mean position to user.
- [R_4] : The system has to be able to redirect the user to the external sharing service system for booking the mean.

[G_{12}] : Allow the user to set the preferences in the settings section

- [R_1] : The system must be able to show the user the possible preferences.
- [R_2] : The system has to register these preferences in its database.
- [R_3] : The system has to have access to these preferences each time is needed.

[G_{13}] : Allow the user to handle his trips

- [R_1] : The system has to be able to create Travel events.
- [R_2] : The system has to be able to show the list trips created.
- [R_3] : The system has to be able to let the user modify the trips when needed.
- [R_4] : The system has to be able to check the trip's data.

4 Scenarios

4.1 Scenario 1

Giuseppe is an employee of a society. One night, before going to sleep, he receives a call from his chief that inform him that he has to attend three meetings in three different zone of Milan. The next day the chief needs to know if Giuseppe can take part at all these meetings or if he has to ask to someone else. Giuseppe has never been in those zones of Milan and, at the moment, he doesn't have his car because it's broken. Thus, he has to find another solution. In that moment an idea came in Giuseppe's mind! He opened the Travlendar+ app on his smartphone and inserted the three events in the calendar using a TAG called *no car*, which was specifically created for this situation. This tag includes all the means of transport supported by the app except the car. During the creation of the last event, Travlendar+ has showed to Giuseppe a warning telling him that the event was unreachable in time from the previous one he set. Knowing this information, Giuseppe called his chief for telling him that he has to ask to someone else just for the third event and that he can attend the first two without any problems because he will take the means suggested by Travlendar+.

4.2 Scenario 2

Michela is a very busy person and she cares about the environment. She has been very happy to download Travlendar+ on her smartphone because it can suggest the means of transport that have the less impact on the environment. Thus, every day Michela inserts her events in Travlendar+ and she associates to each one the CO_2 *free* preference. In this way, the system gives her the information about all the means of transport available for reaching the destination, order by the one who produce the less CO_2 quantity to the one who produce the most. This is very useful for Michela, because now she can both follow her wish of having a good impact on the environment and avoid arriving late to her appointments, finding always a good compromise without any difficulties.

4.3 Scenario 3

Luca lives in Milan and the 20^{th} of October he has a work meeting in Turin. He decides to insert the meeting in his calendar. As soon as he sets all the data defining the event and concludes the creation, a message appears saying that the event is very far and advices Luca to plan a trip. Luca is very happy of having this opportunity and proceeds on. Travlendar+ provides him an interface in which he can chose of buying a ticket for the train or for the airplane or simply adding a travel event with that mean. Luca selects the train, then Travlendar+ asks Luca to fill up a form containing the date and the time in which he would like to leave, in order to provide him all the solutions with the related starting prices. After a quick look, thanks to the ad-hoc filtration of the trains, Luca is able to choose the most suitable ticket. As soon as he decides to purchase it, two things happen:

- a travel event with all the information is automatically created;
- Luca is addressed on the travel service website to complete the purchase.

Luca will be able to check the solutions for arriving to the station on time in every moment tapping on the travel event auto-generated. When Luca returns on the application, he inserts the period of staying and his accommodation in Turin. After this Travlendar+ advices him to buy the ticket that represents the best compromise between price and utility. Anyway, it also shows him the other possible tickets he can buy.

4.4 Scenario 4

Sean lives in Milan and he has to attend a work-lunch in Rome tomorrow. He has already the ticket for reaching Rome, because his company has already bought it. Thus, he wants to add the travel event in Travlendar+. Nothing of more simple! Sean has only to tap for about one second the screen on the calendar interface and two choices will appear on the screen: *create an event* and *plan a trip*. Tapping on plan a trip, Travlendar+ will show to Sean an interface in which he can chose the mean of transport and, after this, he can simply insert all the info of his ticket and create the travel event tapping on *Add*.

4.5 Scenario 5

Gianluca is a financial expert, he has always many meetings during the day and sometimes he forgets to take a break. He needs something which reminds him to reserve some time for having lunch. Travlendar+ is the perfect solution to his needs. Gianluca set in the preferences the span of time in which he wants to have lunch, from 12 to 14, and the preferred duration, 60 minutes. Today Gianluca has to add to his calendar one event from 12:00 to 13:00 and another one from 13:30 to 15:00. Travlendar+, given the Gianluca's setting for the lunch, has already automatically created an event lunch every day between from 12:00 and 13.00. When Gianluca adds the first event, the lunch break slides automatically and covers the period between 13.00 and 14.00 without giving any problems. When the second event was added, some problems came up! In fact, Gianluca will not be able to have a lunch of 60 minute. Despite this, Travlendar+ must reserve at least 30 minutes to lunch per day to each user, so it asks Gianluca if he wants to reduce his lunch time to 30 minutes or postpone the event. Gianluca cannot postpone the event, so he decides of reducing his lunch duration to 30 minutes for tomorrow, leaving the other lunch events unchanged.

4.6 Scenario 6

Gabriele has inserted in his Travlendar+ application his girlfriend tonight's party event but he has completely forgotten about it. Fortunately, Travlendar+ sends him warnings to remind him to go to the event whenever the time for reaching the party's location, without arriving late, with the slowest mean of transport in the list is going to expire. But during almost all these warnings Gabriele was sleeping and he did not read them. He just gets the last one, which advices him to use the car. Unfortunately, Gabriele doesn't have a car in Milan but he has the drive licence and so he can take a car sharing. Travlendar+ is perfect for Gabriele's situation! In fact, when he taps on the car choice, the app opens an interface that asks Gabriele to choose between his own car and a car sharing. Choosing the second one, Gabriele is able to see the route for reaching the nearest car location and he can also be addressed to the car sharing service app to book it and reaching his girlfriend party's location on time. Travlendar+ will also show him the fastest route to reach his destination.