POLITECNICO
MILANO 1863

MSc Computer Science
and Engineering

Software Engineering 2
academic year 2017-2018

TRAVLENDAR+

Requirements Analysis and Specification Document

894135
Franklin Onwu
franklinchinedu.onwu@mail.polimi.it

899318
Ivan Sanzeni
ivan.sanzeni@mail.polimi.it

884021
Matteo Vantadori
matteo.vantadori@mail.polimi.it

Related professor:
Prof. Matteo Giovanni Rossi

Release Date: October 29th 2017
Version 1.0

Contents

1 | Introduction

1.1 Goals

Section [G.0] treats all the goals related to registration to the application:

G.0.1 The unregistered user can sign up to the Travlendar+ services.

G.0.2 The registered user can sign in to Travlendar+.

G.0.3 The registered user can sign out to Travlendar+.

G.0.4 The registered user can delete his/her account from the Travlendar+ services.

Section [G.1] treats all the goals related to the creation and personalization of an event:

G.1.1 The Travlender can schedule a new event, adding name, time slot, location, type and (eventually) a description.

G.1.2 The Travlender can modify the name of the event.

G.1.3 The Travlender can modify the description of the event.

G.1.4 The Travlender can modify the starting time of the event.

G.1.5 The Travlender can modify the ending time of the event.

G.1.6 The Travlender can modify his/her event from a work event to a personal event or viceversa.

G.1.7 The Travlender can insert the description at a later time or modify it.

G.1.8 The Travlender can choose how many minutes or hours early arrive to the destination.

G.1.9 The Travlender can delete an existing event.

Section [G.2] treats all the goals related to the customization of the Travlender preferences:

G.2.1 The Travlender can add his/her bus, train, metro or tram season tickets, if he/she already has one.

G.2.2 The Travlender can prefer the quickest way to travel by default.

G.2.3 The Travlender can prefer the cheapest way to travel by default.

G.2.4 The Travlender can prefer the most ecological way to travel by default.

G.2.5 The Travlender can prefer only ways to travel accessible by a person with disabilities by default.

G.2.6 The Travlender can allow the application to select a certain time of vehicle only in a chosen time slot.

G.2.7 The Travlender can set a maximum distance to cover by walk.

G.2.8 The Travlender can set a maximum amount of money to spend in public or shared transportations.

Section [G.3] treats all the goals related to all the purchases required by public and shared services:

G.3.1 The Travlender can buy in-app a ticket for travel by metro.

G.3.2 The Travlender can buy in-app a ticket for travel by bus.

G.3.4 The Travlender can buy in-app a ticket for travel by trolleybus.

G.3.5 The Travlender can buy in-app a ticket for travel by tram.

G.3.6 The Travlender can buy in-app a ticket for travel by train.

G.3.7 The Travlender can book in-app a taxi.

G.3.8 The Travlender can take a bike from a bike sharing service.

G.3.9 The Travlender can take a car from a car sharing service.

Section [G.4] treats all the goals related to the travel and all the changes that the Travlender can make in real time:

G.4.1 The Travlender must arrive in time to the event, as far as possible.

G.4.2 The Travlender can decide the path to follow to reach the upcoming event.

G.4.3 The Travlender can decide to follow a different road from that suggested by the application.

G.4.4 The Travlender should follow a path sheltered from adverse weather conditions.

Section [G.5] treats all the goals related to the category of flexible events:

G.5.1 The Travlender can select a minimum amount of time to dedicate to the flexible event.

1.2 Scope

Travlendar+ is a calendar-based application designed to schelude any kind of event, supporting the user to reach the location of the events all across Milan, combining different sort of means, in relation to the user preferences. The application handles four typologies of event:

1. Activity: which provides calendar, reminder and street direction supports and cannot be overlapped with other activities.
2. TO DEFINE: which provides calendar, reminder and street direction supports and can be overlapped with activities.
3. Flexible event: which provides calendar, reminder and street direction supports and can be overlapped with activities as long as exists a minimum amount of time fixed by the user.
4. Transfer: which provides calendar and reminder supports and cannot be overlapped with other activities. It is used for events that take place outside Milan.

Travlendar+ also provides in-app purchases for public transport tickets (metro, bus, trolleybus, tram, train) in Milan and bookings for sharing or public services (taxi, car-sharing, bike-sharing). Travlendar+ takes account of different user preferences, like the opportunity to travel owned vehicles, the possibility to choose different algorithms to set the course (quickest, cheapest or most ecological) and offers the opportunity to reach the location choosing a path sheltered from adverse weather conditions.

1.3 Definitions, acronyms, abbreviations

Definitions

Travlendar+ = the name of the application.

Travlender = a registered and logged user of Travlendar+.

Event =

Activity = kind of event that provides calendar, reminder and street direction supports and cannot be overlapped with other activities.

TO DEFINE = kind of event that provides calendar, reminder and street direction supports and can be overlapped with activities.

Flexible event = kind of event that provides calendar, reminder and street direction supports and can be overlapped with activities as long as exists a minimum amount of time fixed by the user.

Transfer = kind of event that provides calendar and reminder supports and cannot be overlapped with other activities. It is used for events that take place outside Milan.

Quick =

Cheap =

Eco =

Personal =

Work =

Not wet =

Acronyms

RASD =

GPS =

API =

TMS = Transporting Managing System

Abbreviations

G.n.m =

D.n.m =

R.n.m =

1.4 Revision history

1.5 Reference documents

1.6 Document structure

2 | Overall description

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 Assumptions, dependencies and constraints

2.4.1 Assumptions

Application assumptions

A.1.1 In the minimization of carbon footprint all the public vehicles are considered like having zero-emission since the user's presence would not influence the travel mean's emission.

A.1.2 The car's trip cost is assumed to be the same for all cars, and only depends on the distance.

Domain assumptions

D.2.1 The payment credentials are verified by a reliable external service.

D.2.2 The traffic informations, coming from external services are reliable.

D.2.3 The gps positions are always accurate.

D.2.4 It is possible to keeps track of the position of your personal means through external service.

D.2.5 The cost for all car trips is given by the estimate of the kilometers per liter and euros per liter.

D.2.6 All the trams, buses, trolleybuses and metro are available for people with disabilities.

D.2.7 There are no unforseen events that can cause delay to the user (es. accidents).

3 | Specific requirements

3.1 External interface requirements

3.1.1 User interface

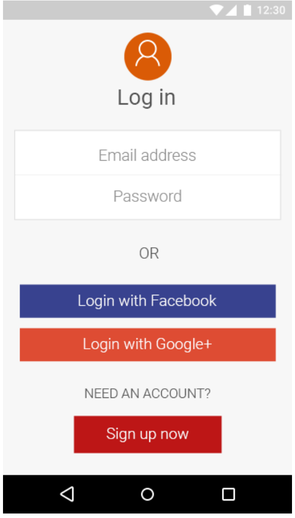


Figure 3.1: Login

The unregistered or unlogged user can log in to his/her account (if he/she has already got one) or create a new one. The application allows to log in with his/her own Facebook or Google+ account.

The first time the *Travlender* starts the application a pop-up appears, in which Travlendar+ asks permission to use the user's location. The user can tap on *allow* to give it, or *cancel* to refuse. In the second case, many application functions will be not accessible.

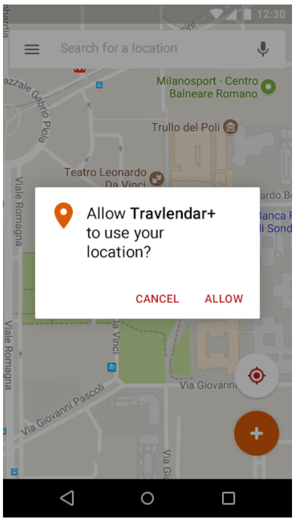

Allow Travlendar+ to use your location?

Figure 3.2: Alert

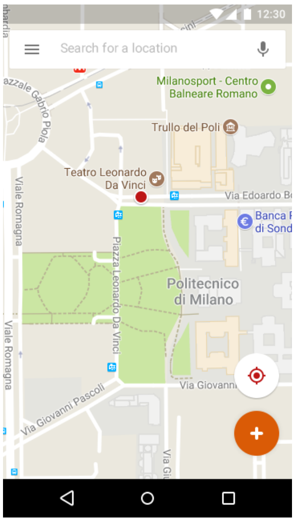
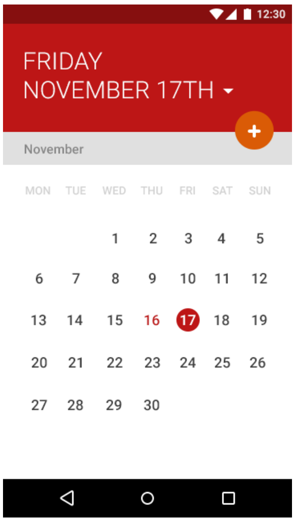


Figure 3.3: Map

If the *Travlender* decides to use the map view, he can see two buttons on the right. The first one centers the map on his/her position, the second one allows him/her to schedule a new event.

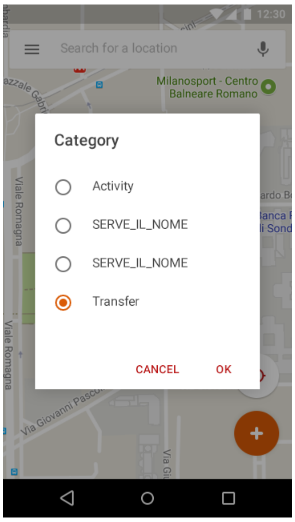
If the *Travlender* decides to use the calendar view instead, he can see only one button on the right that allows him/her to schedule a new event. The Travlender can go to the previous or next month swiping on the left or on the right, respectively.


FRIDAY
NOVEMBER 17TH

November

MON	TUE	WED	THU	FRI	SAT	SUN
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Figure 3.4: Calendar


Category

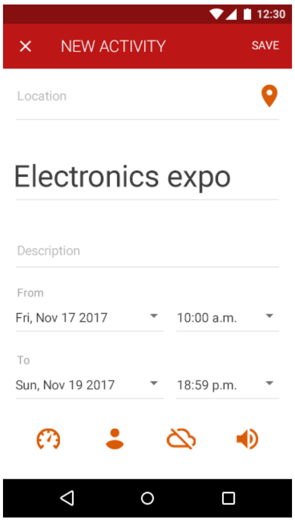
○ Activity
○ SERVE_IL_NOME
○ SERVE_IL_NOME
● Transfer

CANCEL OK

Figure 3.5: Choice of event category

AAA

Now a new screen appears, in which the user can insert the location of the event, its name, a description, the starting and ending date and time and can modify the default settings, tapping on the fourth buttons on the bottom. The first one permits to change the travel preference from *quick* to *cheap*, or *eco*. The second one permits to change the event type from *work* to *personal*. The last but one ables (or disables) the *not wet* travel. The last one ables (or disables) *notifications*.



12:30

NEW ACTIVITY SAVE

Location

Electronics expo

Description

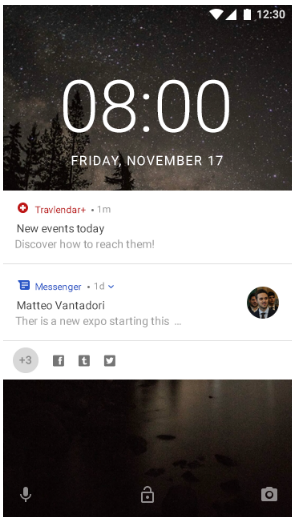
From

Fri, Nov 17 2017 10:00 a.m.

To

Sun, Nov 19 2017 18:59 p.m.

Figure 3.6: Creation of a new event



12:30

08:00

FRIDAY, NOVEMBER 17

Travlendar+ • 1m

New events today

Discover how to reach them!

Messenger • 1d

Matteo Vantadori

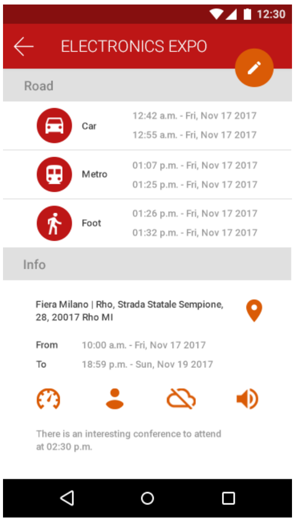
Ther is a new expo starting this ...

+3

Figure 3.7: Lock screen

AAA

AAA



12:30

ELECTRONICS EXPO

Road

Car 12:42 a.m. - Fri, Nov 17 2017
12:55 a.m. - Fri, Nov 17 2017

Metro 01:07 p.m. - Fri, Nov 17 2017
01:25 p.m. - Fri, Nov 17 2017

Foot 01:26 p.m. - Fri, Nov 17 2017
01:32 p.m. - Fri, Nov 17 2017

Info

Fiera Milano | Rho, Strada Statale Sempione, 28, 20017 Rho MI

From 10:00 a.m. - Fri, Nov 17 2017
To 18:59 p.m. - Sun, Nov 19 2017

There is an interesting conference to attend at 02:30 p.m.

Figure 3.8: Summary of the event

3.1.2 Hardware interfaces

3.1.3 Software interfaces

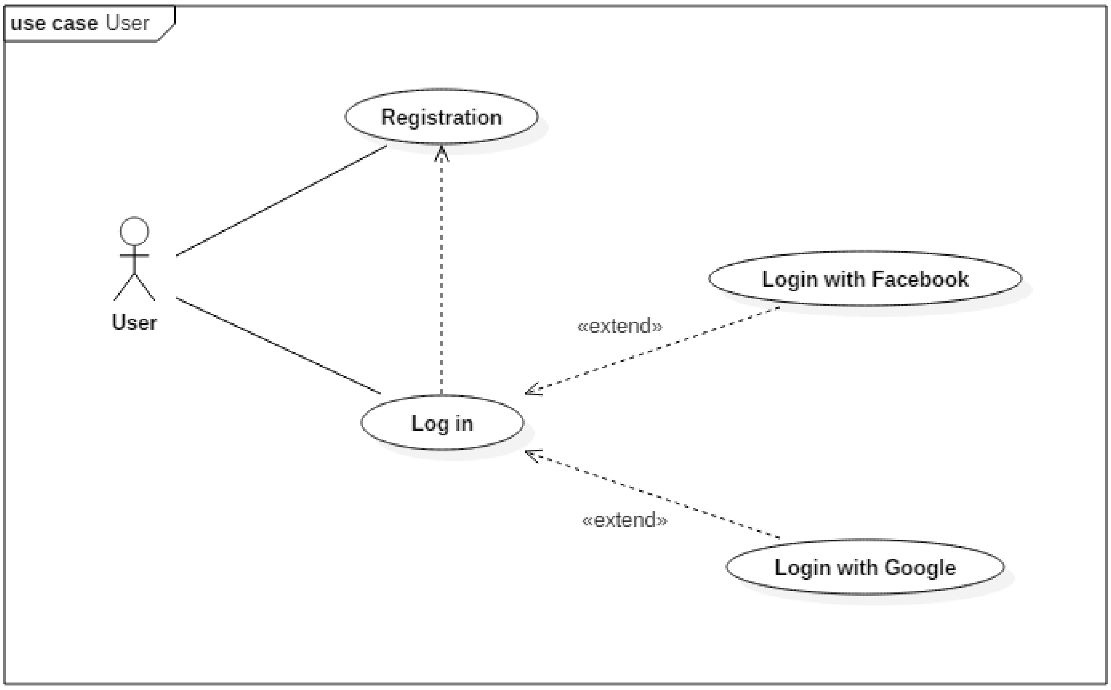
3.1.4 Communication interfaces

3.2 Functional requirements

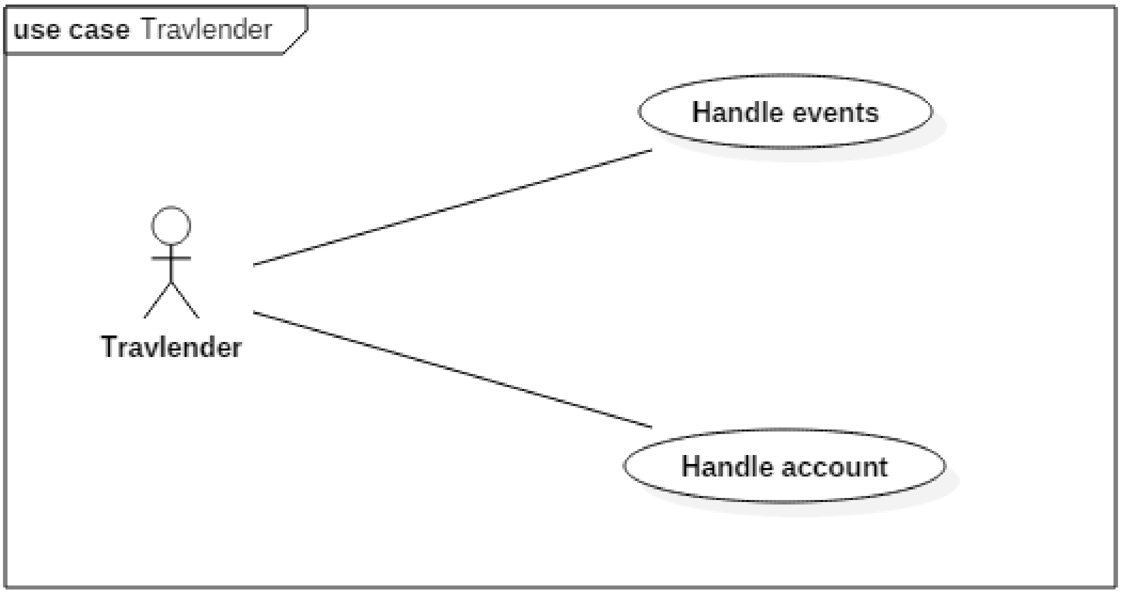
INSERIRE DEFINITION OF USE CASE DIAGRAMS!!!

3.2.1 Use case diagrams

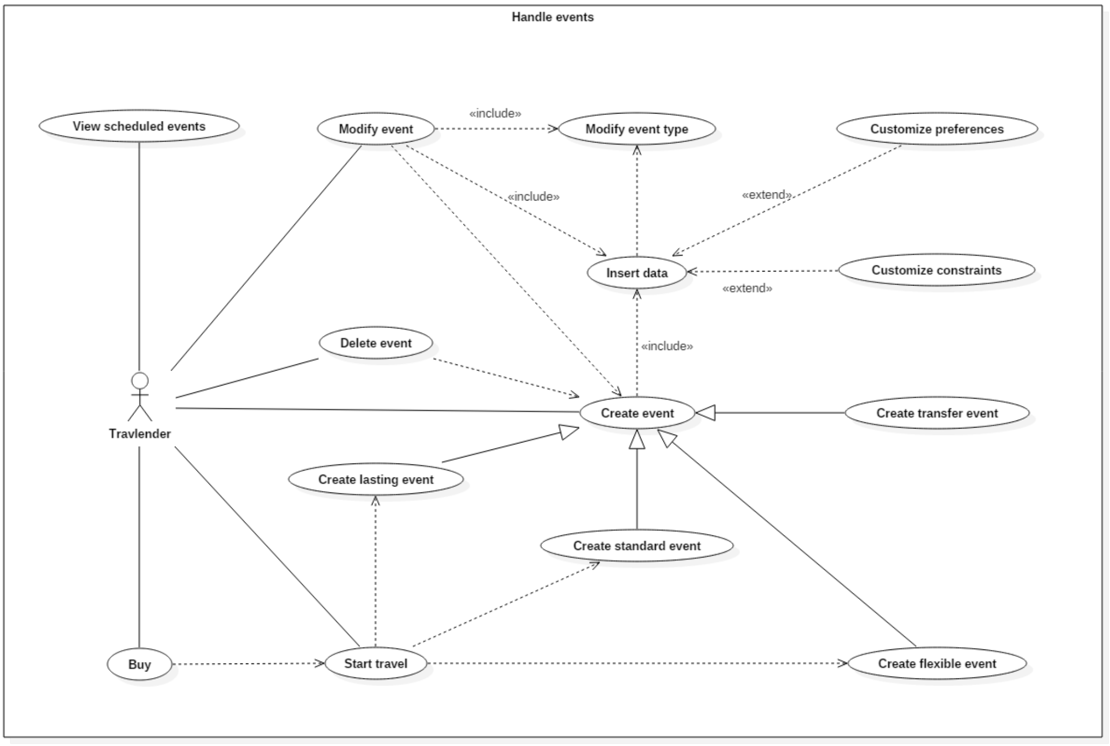
User



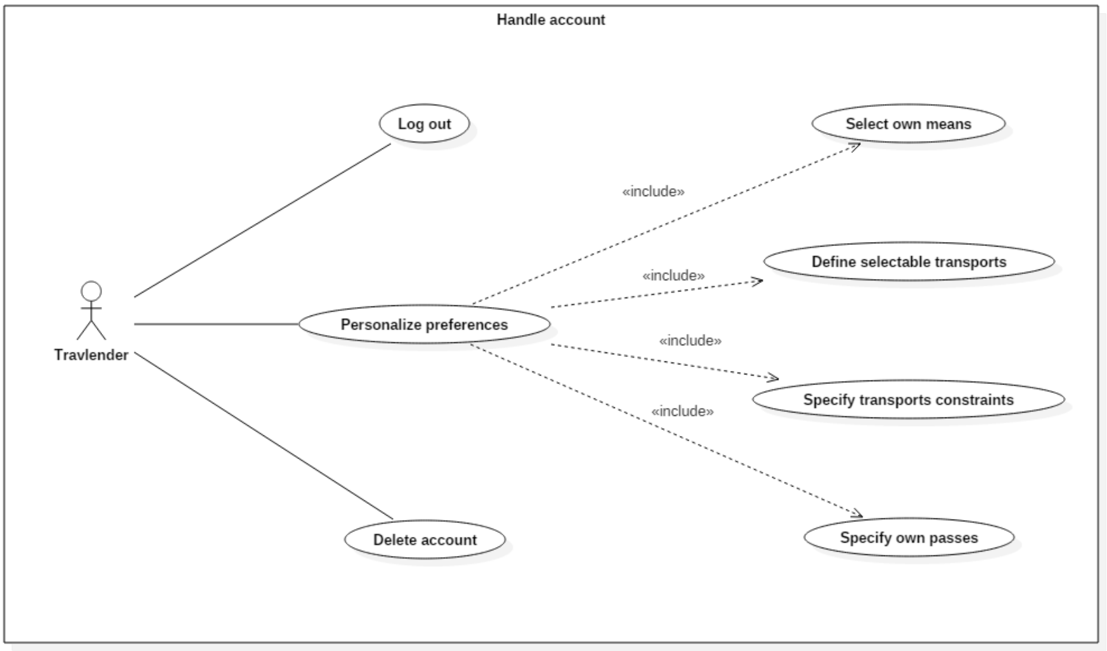
Travlender



Handle events



Handle account



Name	User registration
Actors	User
Goals	
Input conditions	The user must have downloaded the application and opened it on the login screen
Events flow	1. The user taps Become a Travlender 2. The user fullfills the registration form and taps Send 3. The system verifies the data 4. The system shows a confirm message on the screen
Output conditions	The user is registered
Exceptions	The user makes a mistake in the registration form: 4.a. The system shows an error message on the screen Then the flow restarts from point 2

Name	Login
Actors	User
Goals	
Input conditions	The user must be registered and have opened the appliation on the login screen
Events flow	1. The user fills out Email address and Password fields and taps Enter 2. The system verifies the account. 3. The system shows a confirm message on the screen
Output conditions	The user is logged
Exceptions	The user inserts an invalid email address or password: 3.a. The system shows an error message on the screen Then the flow restars from point 1

Name	Login with Facebook
Actors	User
Goals	
Input conditions	The user must be registered and have opened the appliation on the login screen
Events flow	1. The user taps Login with Facebook 2. The system verifies the account 3. The system shows a confirm message on the screen
Output conditions	The user is logged
Exceptions	The user hasn't an associated Facebook account: 3.a. The system shows an error message on the screen Then the flow restars from point 1

Name	Login with Google+
Actors	User
Goals	
Input conditions	The user must be registered and have opened the appliation on the login screen
Events flow	1. The user taps Login with Facebook 2. The system verifies the account 3. The system shows a confirm message on the screen
Output conditions	The user is logged
Exceptions	The user hasn't an associated Google+ account: 3.a. The system shows an error message on the screen Then the flow restars from point 1

Name	View scheduled events
Actors	Travlender
Goals	
Input conditions	The user must be logged
Events flow	1. The Travlender swipes the screen on the left and a menu appears 2. The Travlender taps Scheduled events
Output conditions	The Travlender can view the scheduled events
Exceptions	There are no exception cases

Name	Create standard event
Actors	Travlender
Goals	
Input conditions	The user must be logged
Events flow	1. The Travlender taps the + button 2. The Travlender taps the Standard event category 3. The Travlender inserts the name 4. The Travlender inserts starting time and ending time 5. The Travlender inserts a description (optional) 6. The Travlender modifies the constraints (optional) 7. The Travlender adds a location 8. The Travlender taps Save
Output conditions	A new standard event is created
Exceptions	The Travlendar inserts a time slot overlapped with another standard event: 5.a. The system shows an error message on the screen Then the flow restars from point 4
	The Travlender inserts a non-existent location: 8.b. The system shows an error message on the screen Then the flow restars from point 7

Name	Create lasting event
Actors	Travlender
Goals	
Input conditions	The user must be logged
Events flow	1. The Travlender taps the + button 2. The Travlender taps the Lasting event category 3. The Travlender inserts the name 4. The Travlender inserts starting time and ending time 5. The Travlender inserts a description (optional) 6. The Travlender modifies the constraints (optional) 7. The Travlender adds a location 8. The Travlender taps Save
Output conditions	A new lasting event is created
Exceptions	The Travlender inserts a non-existent location: 8.a The system shows an error message on the screen Then the flow restars from point 7

Name	Create flexible event
Actors	Travlender
Goals	
Input conditions	The user must be logged
Events flow	1. The Travlender taps the + button 2. The Travlender taps the Lasting event category 3. The Travlender inserts the name 4. The Travlender inserts starting time and ending time 5. The Travlender inserts a minimum time slot 6. The Travlender inserts a description (optional) 7. The Travlender modifies the constraints (optional) 8. The Travlender adds a location 9. The Travlender taps Save
Output conditions	A new flexible event is created
Exceptions	The Travlender inserts a time slot bigger than the event duration: 6.a The system shows an error message on the screen Then the flow restars from point 5
	The Travlender inserts a non-existent location: 9.b The system shows an error message on the screen Then the flow restars from point 8

Name	Create transfer event
Actors	Travlender
Goals	
Input conditions	The user must be logged
Events flow	1. The Travlender taps the + button 2. The Travlender taps the Transfer event category 3. The Travlender inserts the name 4. The Travlender inserts starting time and ending time 5. The Travlender inserts a description (optional) 6. The Travlender adds a location 7. The Travlender taps Save
Output conditions	A new transfer event is created
Exceptions	The Travlender inserts a non-existent location: 7.a The system shows an error message on the screen Then the flow restars from point 6

Name	Delete event
Actors	Travlender
Goals	
Input conditions	The user must be logged and must exist at least an event
Events flow	1. The Travlender swipes the screen on the left and a menu appears 2. The Travlender taps Scheduled events 3. The Travlender taps the event he wants to delete 4. The Travlender taps Delete
Output conditions	A new transfer event is created
Exceptions	There are no exception cases

Name	Modify event
Actors	Travlender
Goals	
Input conditions	The user must be logged and must exist at least an event
Events flow	1. The Travlender swipes the screen on the left and a menu appears 2. The Travlender taps Scheduled events 3. The Travlender taps the event he wants to modify 4. The Travlender modifies the desired fields 5. The Travlender taps Save
Output conditions	A new transfer event is created
Exceptions	The Travlender inserts a non-valid field: 5.a The system shows an error message on the screen Then the flow restars from point 4

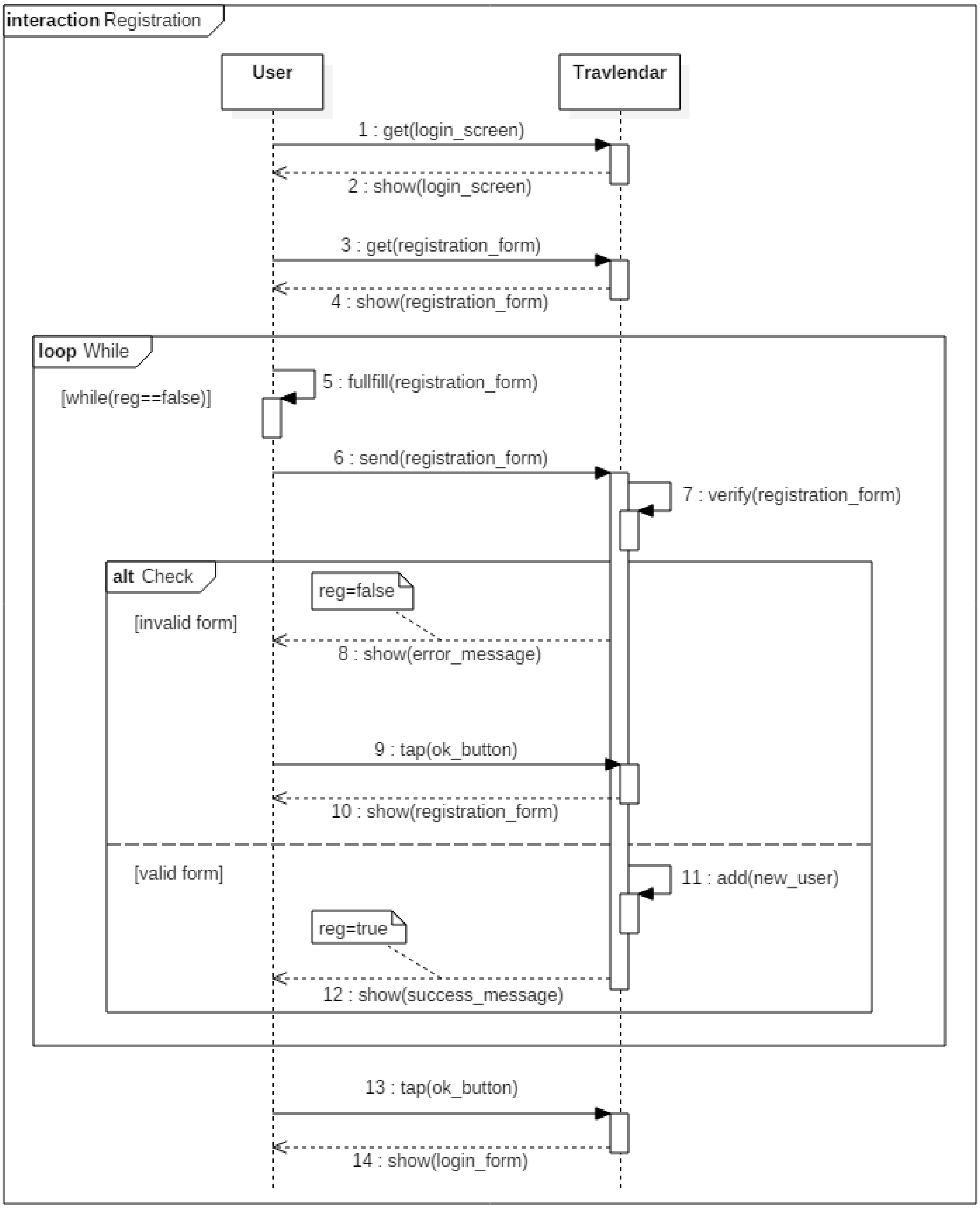
Name	Start travel
Actors	Travlender
Goals	
Input conditions	The user must be logged and must exist at least an event
Events flow	1. The system warns the Travlender that an event is about to starts 2. The Travlender taps Go
Output conditions	The event starts
Exceptions	The Travlender taps Go when he can no longer reach the event before the start, but he can still reach it before the end: 2.a. The system warns the Travlender he/she is late Then the flow restarts from point 2
	The Travlender taps Go when he can no longer reach the event before the end: 2.b The system warns the Travlender he/she can no longer reache the event before the end Then the flow ends

Name	Buy
Actors	Travlender
Goals	
Input conditions	The user must be logged and the event must be started
Events flow	1. The Travlender swipes the screen on the left and a menu appears 2. The Travlender taps Scheduled events 3. The Travlender taps the ongoing event 4. The Travlender taps the public mean he/she wants to buy a ticket or the mean he/she wants to book 5. The Travlender taps Buy/Book
Output conditions	The Travlender buys the ticket or books the mean
Exceptions	There are no exception cases

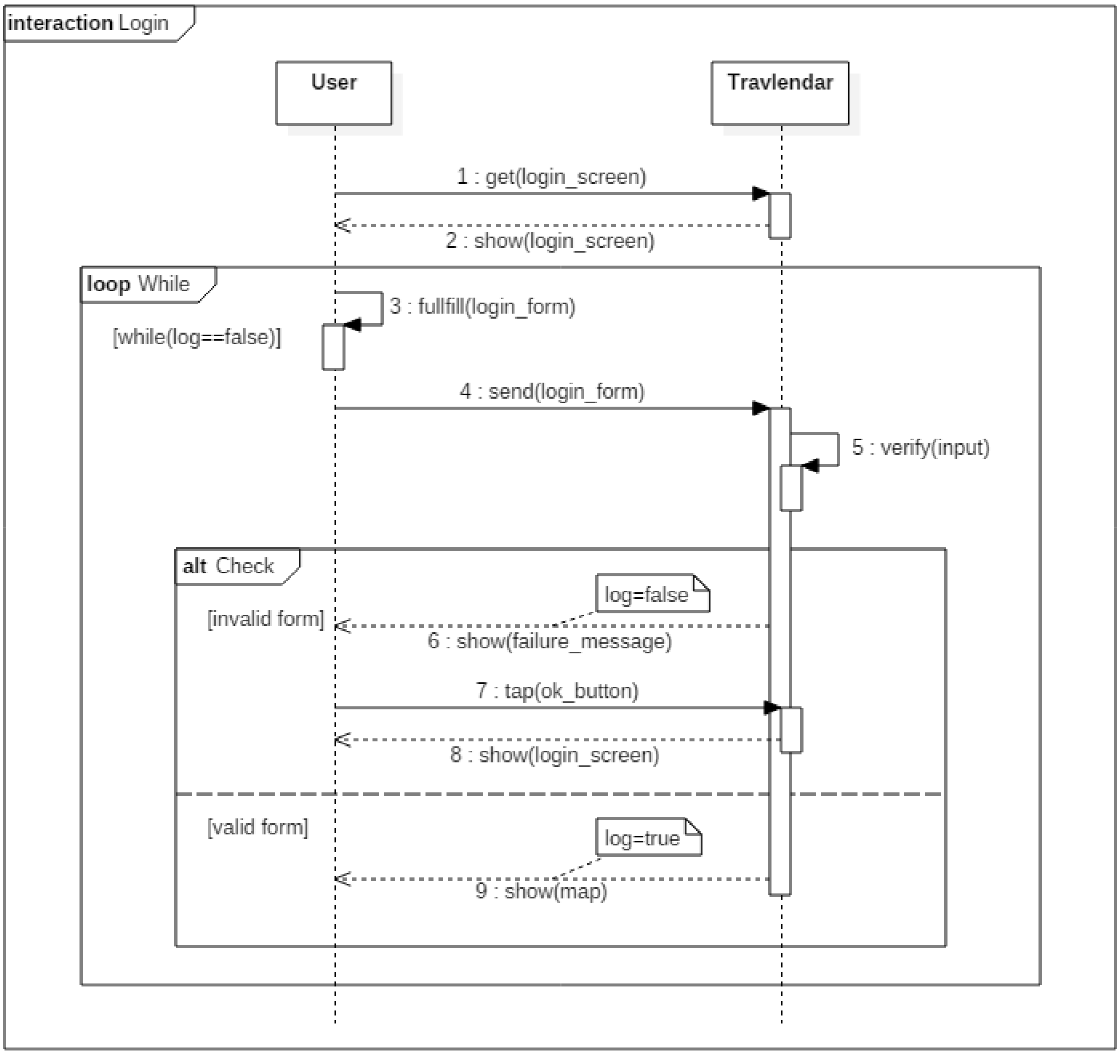
3.2.2 Class diagram

3.2.3 Sequence diagrams

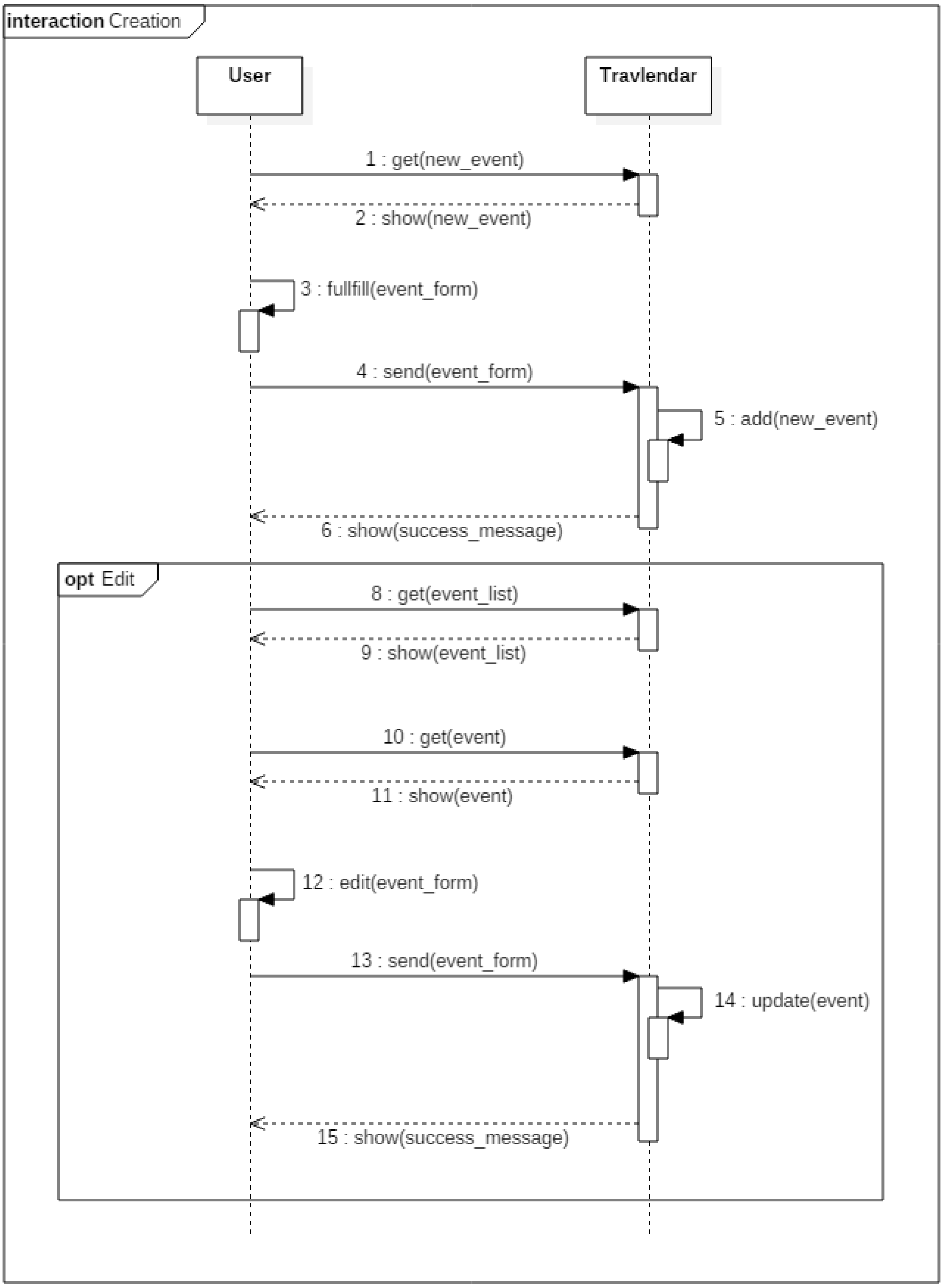
User registration



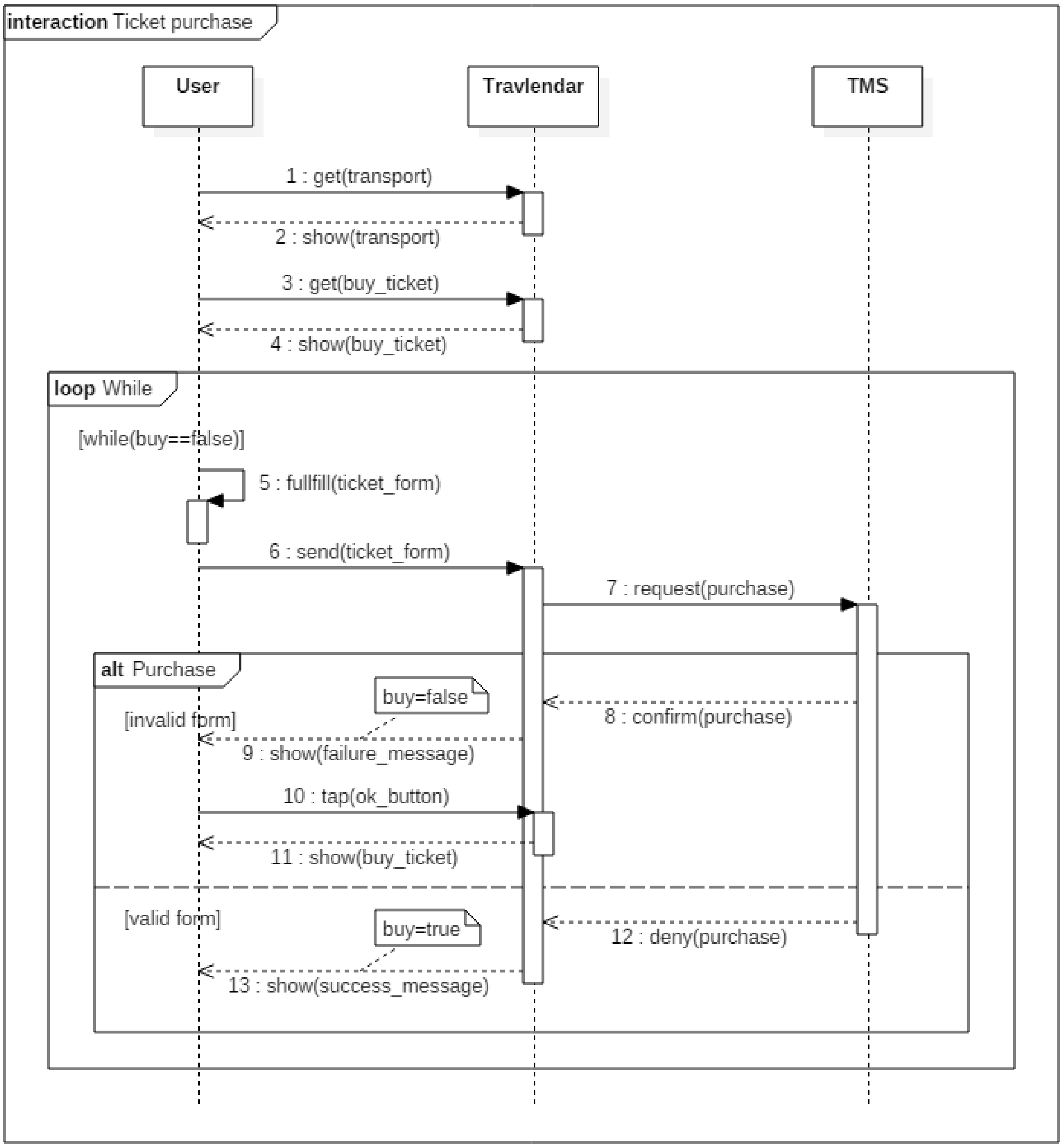
User login


interaction Login
User
Travlendar
1 : get(login_screen)
2 : show(login_screen)
loop While
[while(log==false)]
3 : fullfill(login_form)
4 : send(login_form)
5 : verify(input)
alt Check
[invalid form]
log=false
6 : show(failure_message)
7 : tap(ok_button)
8 : show(login_screen)
[valid form]
log=true
9 : show(map)

Event creation and edit



Ticket purchase



Section [R.0] treats all the requimerements related to registration to the application

R.0.1 The system must provide a registration form to the user.

R.0.2 The system must verify whether the user entered all the required data.

R.0.3 The system must check whether the user results already registered.

Section [R.1] treats all the requimerements related to the application login:

R.1.1 The system must check the input data and controls if they correspond to an existing account.

R.1.2 The system must guarantee the login of the user if and only if the inserted credentials results verified.

Section [R.2] treats all the requimerements related to the creation or modification of events:

R.2.1 The system must verify that the user is available for the duration of the event.

R.2.2 The system must verify that the free time preceeding the event is long enough for the travel to the new location.

R.2.3 The system must control wheter it is possible for the user leaving from the previous event's location to arrive on time.

R.2.4 The system must control wheter it is possible for the user leaving the event to be able to arrive on time at the following event's location.

R.2.5 The system must control wheter given the current position the user can be able to arrive at the location on time for the appointment.

R.2.6 The system most inform the user when it is not possible to schedule his event in the allotted time.

R.2.7 The system must suggest possible solutions to arrange conflicting events: either postponing the beginning time of the later event or anticipating the ending time of the earlier one.

Section [R.4] treats all the requimerements related to the public means reservation:

R.4.1 The system must be able to locate the nearest bike to bike sharing system in the city.

R.4.2 The system must be able to locate the nearest car sharing system in the city.

R.4.3 The system must be able to locate the public trasportation areas in the city.

Section [R.5] treats all the requimerements related to outside phenomena:

R.5.1 The system must notify the user in occurance of strikes.

R.5.2 The system must notify the user in occurance of days of bad weather.

Section [R.6] treats all the requimerements related to the customization of the preferences:

3.3 Performance requirements

3.4 Design constrains

3.4.1 Standards compliance

3.4.2 Hardware limitations

3.4.3 Any other constraint

3.5 Software system attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

4 | Formal analysis using alloy

5 | Effort spent

6 | References