Contents

Table of Contents

depth 2

Introduction

Travlendar+ is an appointment management system capable of efficently scheduling a user's calendar. Thanks to this application it is possible to calculate in advance the estimated time to reach a meeting point based on the transportation choices made my the user.

The application not only is used for interregional travels but also for personal or work appointments. Once the user is registered he or she will be able to set their preferences regarding transportation choices to reach their destination. The application will suggest different itineraries to the user based on different transportation methods, based on metrics such as transportation time and cost.

Travelendar+ is a mobile and web application capable of improving its user's everyday life with simplicity, efficency and safety.

Purpose

The main functions offered by Travlendar+ are substantially three:

- **Calendar** App manager functions
- **Maps Geolocalization** App functions
- Customizable **Personal User** page and preferences functions.

To make more readable the document, we divided the main Goals of Travlendar+ for these category

The Goals are:

Calendar Functions

1) Show a personal calendar of 'Events' submitted.
2) Let the User submit 'Events' on the calendar to plan his/her future appointments.
3) Notify the User when the 'Event' is about to start and she/he needs to leave to be in time to the 'Event Meeting Location'.
4) Let the User to create 'Flexible Event' for everyday short and low priority appointments.

Maps Geolocalization Functions

1) Geolocate the gps coordinates of the "Starting Location" and the "Meeting Location" on a virtual map.
2) Calculate a list of possible shortest routes from a "Starting Location" and the "Meeting Location" with the 'Vehicles' available.
3) Calculate the extimation time of arrival at the destination for each route.
4) Let the User choose a route from the list of "Best Route" provided by the algorithm
5) Let the User modify the "Best Route" adding 'Costraint' for 'Intermidiate Locations', preferred 'Vehicle', max distance with a specified 'Vehicle' or max time on a specified 'Vehicle'.
6) Consider on the possible 'Vehicle' avaible all the public transports of the city, railway stations, aeroports, train stations, car and bike sharing systems, bike, car and by foot.
7) notify with a Warning message if the 'best Route' chosen by the User is not good and he/she may arrive on late at the 'Meeting' because of its Extimation Time too long.

8) Suggest a "Best Route" to the User with a 'Vehicle' which is appropriate for the day time of the appointment, the geographical location, the type of the meeting and the weather.

User Profile Manager Functions

1) Let the User to sign in to the Service filling an online form.
2) Let the User to login to their personal User page and update their informations.
3) Let the User fill their 'Vehicle' preferences or dislike for best result on Algorithm 'Best Route' calculus.
4) Let the User buy online tickets for the majority of the public transports involved on the 'Best Route' chosen.
5) (**optional**) Let the User modify settings for the Algorithm 'Best Route'such activating 'Green Mode' or enabling options like "No traffic lighters", "No Schools at 16:00", "Show Autovelox".
6) (**optional**) Let the User submit on his/her page the availability of public transports subcription, driver licence, coupons for special transports for best result on Algorithm 'Best Route' calculus.

Scope

Travelandar+ has a simple scope, helping the Users to planning efficiently their appoinments. We can analyze all the shareded phenomena by the App and the Users.

Shared Phenomena:

1) Registration on the Travlendar+ Service
2) Submitting the 'Event'
3) Submitting the preferences of the User
4) Alarm function
5) Calculus of the 'Best Route'

Not Shared Phenomena:

1) Geographical positions of the 'Event's Locations.
2) Presence of roads, streets, railway stations, public transports in the city.
3) Preferences of the User.

Definitions

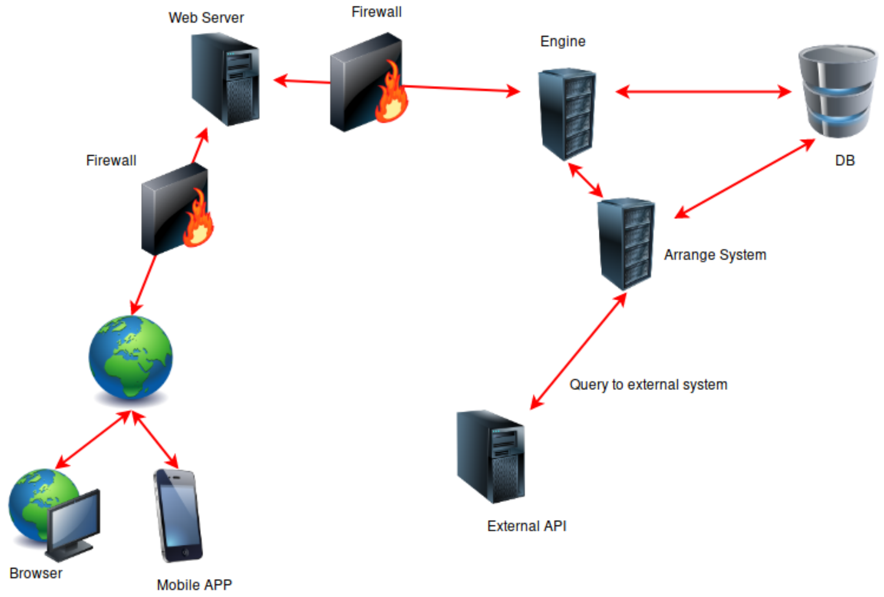
- *Alarm function*: a way in which the phone can notify the user that something important is happening. It is often a sound or a vibration according to the phone used.
- *API:* Application programming interface; it is a common way to communicate with another system.
- *Best Route*: The best route calculated by the algorithm to reach a given event without any delay. The algorithm also considers the user's preferences.
- *Costraint*: Something that controls what you do by keeping you within particular limits.

- *Event*: The users can create some events and submit them to the calendar. The application checks if there are some overlaps and calculates the best route to reach the events.
- *Green Mode*: A user's preference which means that the user prefers to keep his Ecological Footprint as lower as possible. (for instance the user prefer to use a bike instead of a bus)
- *Itermidiate Location*: Any locations between the start point and where the event take place.
- *Meeting Location*: The place or position that the user specifies during the submission of a new event.
- *Route*: The roads you follow to get from one place to another place. The routes are calculated after an event submission.
- *Starting Location*: The position where the algorithm starts to calculate the routes to reach the event.
- *Vehicle*: Something such as a car, bike or bus that takes people from one place to another, also "by foot" is considered a vehicle.
- *Warning*: If there are some problems with an event (like overlapping) the application produces a notify that is attached to the event. The user can select the warning and see more details.

Architectural Design

High-level components and their interaction

The architecture of the system is a 3 Logic Tiers.



When the User submit an event via Web or App the Engine proceed to update the DB and calcolate the route to propose to the user. When the sistem need external information the Arrange System provides through a query to external API (for istance: the ATM schedule or Google Maps route)

The Database store all user informations about his profile preference and his shedule (Events and Best route chosen by the user)

Component View

Deployement View

Runtime View

Component Interfaces

Architectural Styles

Overall Architecture

The software is divided in multiple tiers. On the server side there will be the following components:

1) Frontend endpoint server
2) Web scraping and API daemon to obtain transportation and weather data
3) RethinkDB document-oriented database
4) Routing server

The client will be thin in both the web and android version.

Frontend Endpoint Server

The frontend endpoint server's purpose is to provide REST endpoint in JSON, it is the main and only interface between the client and the server. The API calls will be done through HTTP and will use a token infrastructure for authentication. All API calls will be processed through a secure connection.

The following endpoints will be present in the first (v1) release:

- **/v1/get_token** – POST - user provides credentials and obtains a token to use for further transactions
- **/v1/get_appointments** – GET - receive a JSON describing the appointments specified in the date range
- **/v1/get_route_options** – POST - provides the routing alternatives given a source and destination point
- **/v1/register_appointment** – POST - registers an appointment, if id is specified the appointment is modified
- **/v1/delete_appointment** – POST - deletes the appointment

Other Design Decisions

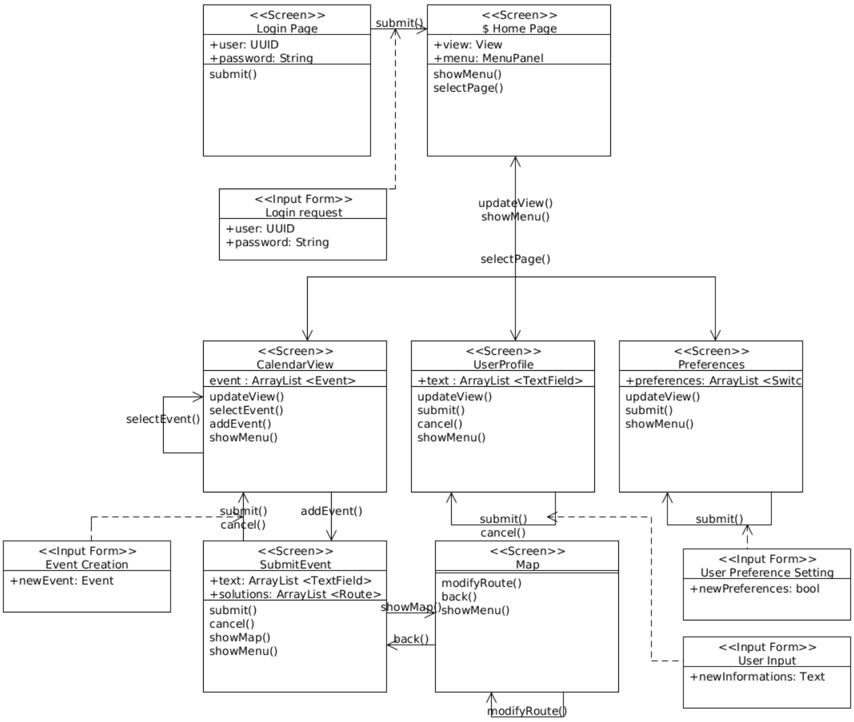
Algorithm Design

User Interface Design

User Interface MockUp

see Section on RASD document.

UX Diagram



Requirements Traceability

Implementation, Integration an Testing

Effort Spent

Prosdocimi Marco

15/10/2017 2h

18/10/2017 2h

21/10/2017 7h

22/10/2017 4h

25/10/2017 3h

26/10/2017 1h

28/10/2017 3h

29/10/2017 2h

Ruggiano Enrico

15/10/2017 2h

16/10/2017 2h

18/10/2017 1h

20/10/2017 4h

21/10/2017 2h

22/10/2017 3h

24/10/2017 2h

25/10/2017 3h

28/10/2017 2h

29/10/2017 5h

Giacomo Vercesi

14/10/2017 5h

15/10/2017 3h

17/10/2017 1h

20/10/2017 3h

22/10/2017 2h

27/10/2017 3h

28/10/2017 5h

29/10/2017 7h

References

Picture

All the picture used for the UI are desiged by yanalya / Freepik.

Those were free downloded from http://www.freepik.com with a Free Licence.

For more informations just read the "Licence free.txt" file in the repository or visit •
http://www.freepik.com/terms_of_use

Icon Credits

- Position, Setting Icon:

  Icon made by CC 3.0 BY from www.flaticon.com

- Calendar, Upload, Menu, Submit Event, Cancel, Accept, Exit, Like, Dislike, Partial like, Hour, Destination Icons:

  Icon made by Flaticon Basic License BY from www.flaticon.com

- Vehicle Icons:

  Icon made by Flaticon Basic License BY from www.flaticon.com

Standard IEEE

standard ISO/IEC/IEEE 29148