

TripLap: Planning Application for Travellers

Amadeu Pereira

up201605646@fe.up.pt

João Ferreira

up201605330@fe.up.pt

Nuno Lopes

up201605337@fe.up.pt

May 27, 2020

Abstract

The current article describes the project for the discipline of Annotation Languages and Data Processing.

The project focuses on the development of a software application, specifically for use on smaller devices, such as smartphones and tablets. The app target group are travellers who wish to manage their travel whilst on the move. The user, with a destination in mind, is presented with a suggestion of places, in the selected city, from which they can choose the ones that better suite their preferences.

This manuscript also describes related state-of-the-art, system requirements, the chosen architecture, the selected data sources, a calendar for the app development and an application overview.

The document concludes with a brief discussion and evaluation of the final result and an analysis of whether the proposed objectives were achieved or not.

the system requirements of the application were defined.

The next step was the creation of the section 4. This one was used to explain the planned architecture for the system, based on the requirements previously delineated.

Afterwards, an analysis of the type of data needed for the project was performed, as well as a search for providers of this data. This is explained in more detail in the section 5.

A description of the plan to ensure an efficient and optimal time management can be found in section 7. Meanwhile, in section 6 there is a brief overview of the visual designs of the application.

There is also a section 8 with a discussion of the challenges faced during the development of this project, a section 9 where the final application is evaluated and, finally, in section 10 the conclusions related to the project.

1.1 Motivation

The main motivation for this project was derived from the personal need the authors of this project have experienced in the past. Understanding that the application can benefit several others who have similar experiences, the authors decided that this would be the perfect opportunity to develop a software application that goes along with our needs that we strongly believe are the same as other travellers.

1.2 Objectives

For this project, the authors pretended to develop a user-friendly and time-efficient mobile application that allows them to carefully plan

their trips. The main features of this application can be summarised in the following points:

- Allow the users to select a city and a time frame in which their trips are planned to occur;
- Supply to the users relevant information about the weather and places to visit during the trip;
- Give detailed information about points of interest, restaurants, shops and hotels of the previously select cities;
- Give the users the possibility to filter the points of interest, restaurants and hotels;
- Show the best route through the use of a map for all the places chosen in each day of the trip;
- Advise about possible problems the user may face with regards to weather conditions and places availability;

2 State-of-the-art

Several other applications are already available for similar purposes as the ones described in this document. In the following paragraphs the authors present some of these together with a critical analysis of the features and functionalities they believe that are important in this type of solutions.

I Love Aveiro [2] is a prototype for an application that aims at augmenting the experience of a traveller to the fullest when visiting Aveiro, Portugal. Everything, from the colours to the font of the letter used was deliberated, in order to transmit to the users the idea of a youthful and modern city. The prototype for this app contained a list with the main points of interest in Aveiro and some default routes that a user could take throughout the city. It also allows them to create their own tracks, to filter the places in the city and add some of these to their favourites. Finally, it also possesses an achievement section which introduces some kind of gamification to the prototype.

YOGO [3] is a trip planning application idea. As the description suggests, it provides the user with useful information about destinations from all over the world and helps managing their time and budget during the journey. Besides this, since the target group is the young generation from 18 to 38 years old, the prototype promotes a strong community bond. It also contains lists with points of interests, allows the users to manage and create trip plans, search for places and flights and lastly maintain a chat with the people who are planning to travel with you.

TripScout [4] is a travel guide application that brings to travellers the recommendations from publishers and local bloggers. It gives to the users the opportunity to find the best places, which are ranked and rated by these influencers, while still being able to add plans to their trip.

Taking into account the previous paragraphs, the group believes that all of these prototypes and applications have interesting features that could be relevant for TripLap. This app helps planning city visits of several days, allowing to assign, for each place, multiple time windows. The application also combines the selection of places to visit with the best route between them.

3 System Requirements

The system developed had as initial requirements that it should be able to handle the gathering of data from multiple sources and in different formats according to their source, which in our case can be restaurants, weather, shops, hotels and points of interest APIs. The system also had a set of user stories, which will be discussed in the next subsection.

3.1 User Stories

There were several user stories that served as a guide for us to fulfill all the application objectives and these can be found in table 2.

4 Solution Architecture

For the development of the server it was chosen *Node.js* with the *Express* framework, providing features for managing requests and routes. These tools allow the creation of a RESTful API which clusters data from six other APIs (OpenWeatherMap API, Zomato API, Amadeus API, Foursquare API, MapQuest API and SerpWow API) retrieving only meaningful data to the client, in JSON format. More information about this sources of data in Section 5. To prevent multiple equal requests to the external APIs the group decided to use a cache system.

On the client side, a framework named *React Native* was used to develop the mobile application. We are also taking advantage of asynchronous local storage system to store the planned trips in the user's mobile phone.

To help the development process of the client we used *Expo Cli*, which allows for a multi platform development of React Native code.

All the requests to APIs are done using *Axios*, which is a promise based HTTP client *Node.js*.

A summarised schema of the architecture can be found in figure 1.

5 Data sources

An essential part of this work was the access to data as location, weather, interests, among others. To ensure this access, several data sources are used as described in the following paragraphs. All the used APIs use JSON as the format for their responses.

5.1 Mobile Position Sensors

The positions sensors from the mobile phone are used to locate the user in the map so that the application can better advise the user with places to visit near them.

5.2 Weather API

The application has access to a public weather platform, named OpenWeatherMap¹, which in-

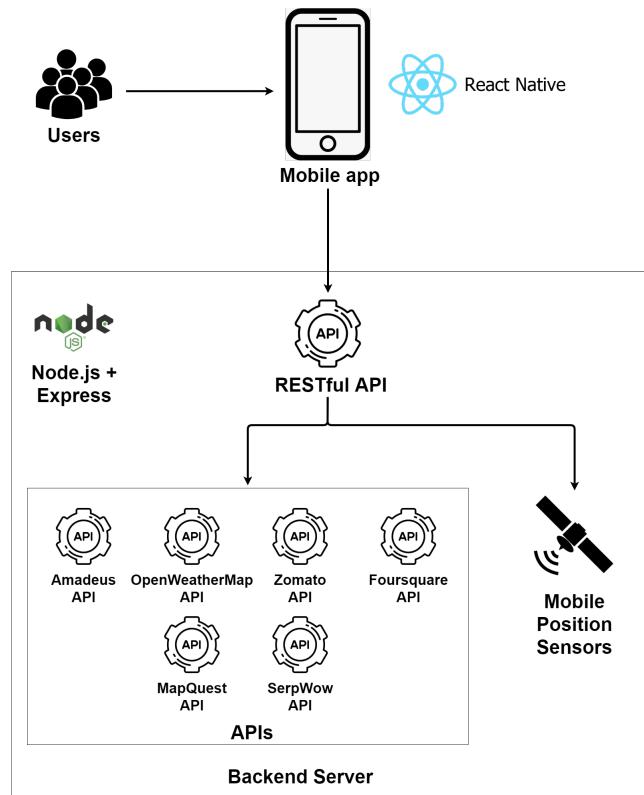


Figure 1: System architecture

forms the user of the weather forecast during the trips duration.

5.3 Restaurants API

The Zomato API² is used to fetch data from nearby restaurants such as the name, location, rating, average cost, type of cuisine, address, phone number and the schedule.

5.4 POI API

The mobile application uses the Foursquare API³ to search points of interest in the selected destinations or the current location of the users. These POIs can either be related to art, events, nightlife or outdoors.

5.5 Hotels API

To look for hotels, the application uses the Amadeus API⁴. This data source gives, not only

²<https://developers.zomato.com/api>

³<https://developer.foursquare.com/>

⁴<https://developers.amadeus.com/>

¹<https://openweathermap.org/api>

the name of the hotel, but also a description, the address, the phone number, the email, the amenities, the rating and average price.

5.6 Shops API

In order to present the shops in a city, the Foursquare API³ is also used. In this case, it provides the application with data about all kind of shops, from supermarkets to drugstores and from shopping malls to bakeries.

5.7 Google Maps

Due to the low limit of free usage that the Google Maps API has and since the React Native Community provides and maintains a free package for Google Maps⁵ to be used on mobile applications, the authors opted to use this package instead of the API to show the best route of a trip.

5.8 Location API

The Amadeus API, referred previously, requires the application to know the latitude and longitude of the city inserted by the user. In order to get this values, the authors made use of the MapQuest API⁶, which allows us to get the geographic coordinates of a city.

5.9 Images API

Geared towards getting a more visual experience to the user, we tried to, whenever possible, provide images alongside the information about places. This being said, the application uses the SerpWow API⁷.

6 Application Overview

In this section the views of the application shown in Annexes 11.2 are explained in detail.

Image 2 displays a simple screen where the user can input either their current location or the city they plan to travel to and the date frame in which their planned trip will occur (Image 3). It also contains a button that takes the user to their previously planned trips that can be seen in image 4.

Image 5 presents the main application screen. This one allows the user to see the most essential information about the planned trip, such as the city, date frame and weather conditions. It also allows the user to access Points of Interest, Restaurants, Shops, Hotels and, finally, access to the full plan of the trip.

Image 16 shows the full plan of a trip, with the places to visit organised by the days. It is possible to be forwarded to a map with the route for a specific day (image 17) by tapping the map button in that day.

Image 6 lists Points of Interest. By checking and unchecking each of those points, the users can add (image 9) or remove them from their trip plan, respectively. Besides that, this page contains a button that opens the filters (image 7).

Image 8 lists Restaurants. By checking and unchecking each of those restaurants, the users can add (image 9) or remove them from their trip plan, respectively. Besides that, this page contains a button that opens the filters (image 10) and by clicking in a restaurant the user can get more information about it (image 11).

Image 15 shows how the list of Shops is displayed. By checking and unchecking each of those shops, the users can add (image 9) or remove them from their trip plan, respectively.

Image 12 presents how the list of Hotels is displayed. By checking and unchecking each of these hotels, the users can add (image 9) or remove them from their trip plan, respectively. Besides that, this page contains a button that opens the filters (image 13) and by clicking in a hotel the user can get more information about it (image 14).

⁵<https://github.com/react-native-community/react-native-maps>

⁶<https://developer.mapquest.com/documentation/>

⁷<https://serpwow.com/docs/search-api/overview>

7 Development plan

In order to develop this project, several checkpoints were defined based on the course deadlines. This information can be found in the Table 1.

Task	Deadline
Project Proposal	11.03.2020
Project setup, integration	25.03.2020
Architecture and Prototype (Presentation and Discussion)	16.04.2020
Architecture and Prototype (Deliverable)	22.04.2020
RESTful API Development	24.04.2020
Mobile App Development	24.05.2020
Final handout	27.05.2020

Table 1: Project checkpoints deadlines

The plan was thoroughly followed, without any change, until the end of the development by the authors of the application, having all the predefined tasks being accomplished before the set deadlines.

8 Discussion

During the development of the mobile application the group faced some moments of deliberation, in which we endeavoured to choose the best course of action.

The first approach taken was regarding the question on whether or not a database should be used, since some kind of data storage is essential in the context of the application. After some discussion, this alternative was perceived as being time-consuming and overkill for the context of this course. To solve this problem the authors opted to make use of the device's local storage. This approach corresponds to the application needs and also was easy to implement.

Some challenges emerged when choosing which APIs would better suits the application.

These problems mainly consisted of three areas: paid subscription, low free request limit and low area coverage. All the major APIs available would either be a paid service or, when

being "free", they would have a low quota request before an upgrade is required. Also, most APIs have the recurring problem of only being focused on big cities, lacking data, sometimes nonexistent in many places.

The chosen weather API only gives a weather forecast up to five days, which affects the planning of trips that extend further than that.

Another unexpected problem that appeared, but was quickly solved, was the fact that Amadeus API was down for a few days.

Although the APIs chosen are less restrictive than the others available on the market, they all have some kind of limitations regarding the number of requests done. To solve this issue, a mock server simulating the API was used in the development stage and a cache system was implemented in production, storing previously made requests.

During the development phase, it was faced an unforeseen issue related to the fact that one the application's goals was to build it cross-platform. This is related to the fact that different operating systems have distinct ways of implementing the same feature, which is time-consuming due to the unexpected research.

In addition to the stated above, as progress was being made and the data sent by the APIs was being integrated into the app, the group realized that some more data was required. It was the case of the images needed for hotels and cities and also the geographic coordinates demanded by the Amadeus API. To obtain this data, SerpWow and MapQuest API were incorporated in the app.

Finally, the application complexity was underrated, being more complex than what initially thought for the established duration of the development. This led to the non-fulfilment of a few application's objectives described in the next section.

9 Evaluation of the Result

In general, the group believes that the app was concluded with success in spite of all the setbacks described in the previous discussion.

Comparing the objectives defined during the idealization of the project and the current state of TripLap, the group settled that only two functionalities were not implemented. These functionalities are the warnings notified to the user regarding, for example, not appropriate weather condition for the plans being made and the ability to rearrange the trip plan. The reason for these functionalities not being integrated due to the complexity of the planned application.

In relation to the applications similar to ours we believe that the simplicity, but at same time, the quite big amount of features that were implemented can be seen as qualities that highlight TripLap from the others. At the same time, applications like YOGO and *TripScout* are certainly more experienced than our group of developers and may have some more insight in the industry of travelling.

Little details like infinite scrolling, which is possible by using pagination, and a cache system recording previously done requests to the server, decreasing the same request's response time, are what makes the user experience unique improving the application flow.

were new to these mobile application's technologies.

References

- [1] World Travel and Tourism Council. Travel and tourism continues strong growth above global gdp. <https://www.wttc.org/about/media-centre/press-releases/press-releases/2019/travel-tourism-continues-strong-growth-above-global-gdp>. Accessed in march, 2020.
- [2] Tiago Bastos. I Love Aveiro. <https://www.behance.net/gallery/83473811/Mobile-App-I-Love-Aveiro>. Accessed in march, 2020.
- [3] Julie Truong. YOGO - Trip Planner App. <https://www.behance.net/gallery/78632553/YOGO-Trip-Planner-App>. Accessed in march, 2020.
- [4] TripScout, Inc. TripScout Travel App & Trip Planner. <https://tripscout.co/>. Accessed in april, 2020.

10 Conclusions

This article presented the project theme and the respective objectives of the application. All the sources from which the group retrieved data were described in its appropriate section, the architecture defined and the development calendar followed. Moreover, detailed user stories and all the application's views were provided, accompanied it a short description describing the functionalities implemented in such screens.

As a final conclusion for this document, the team concluded that using external APIs added more value to the final product. However they can eventually cause problems in the normal behaviour of the application, for instance if they have issues or are unavailable for a long time considering that the mobile application extremely relies on these APIs. Despite this, the group believes that a successful application was achieved considering that all the developers

11 Annexes

11.1 User Stories

User Stories	Description
US01 - Search city	As a user, I want to search for a city so that I can select my trip destination.
US02 - Select the trip date	As a user, I want to select dates so that I can select my trip start date and end date.
US03 - Home-page	As a user, I want to see the homepage with access to feature like points of interest, restaurants, shops and hotels so that I can search in each of these features and see my trip plan.
US04 - Planned trips	As a user, I want to see my previously planned trips so that I can access the trip plans whenever I need.
US05 - Remove planned trips	As a user, I want to be able to remove a planned trip so that I only keep the trips I will need.
US06 - Trip plan	As a user, I want to see a plan for each trip so that I see what is scheduled for each day of each trip.
US07 - Trip warning	As a user, I want to be notified with warnings about weather conditions and occurrences that may affect my plan so that I may rearrange and improve the trip plan.
US08 - Trip map	As a user, I want to see a route for each day of the trip so that I can visit all places in the best possible order.
US09 - List POI	As a user, I want to see a list of points of interest for a specific city so that I can select which ones I want to visit.
US10 - List restaurants	As a user, I want to see a list of restaurants for a specific city so that I can select which ones I am interested in.
US11 - List shops	As a user, I want to see a list of shops for a specific city so that I can select which ones I am interested in.
US12 - List hotels	As a user, I want to see a list of hotels for a specific city so that I can select which ones I am interested in.
US13 - Filter restaurants and hotels	As a user, I want to be able to filter the list of restaurants and hotels by rating and price range so that I can easily find the ones that fit me best.
US14 - Filter POIs	As a user, I want to be able to filter the list of points of interest by Arts and Entertainment, Outdoors and Recreation, Nightlife Spot and Events so that I can easily find the categories that fit me best.
US15 - Detailed information	As a user, I want to see more detailed information about each point of interest, restaurant, shop and hotel so that I can make the wisest choice.
US16 - Add place	As a user, I want to add a place to visit on a specific day of my trip so that I add it to my plan.
US17 - Remove place	As a user, I want to remove a place from my plan so that I can only keep in the trip the places I truly want to visit.
US18 - Rearrange trip	As a user, I want to be able to rearrange the plan of my trip so that I can move each of the plans between days.
US19 - Route mode	As a user, I want to be able to select between walking or driving when seeing a route for my plan so that I can choose the most appropriate route for me.

US20 - Weather prediction	As a user, I want to see a weather prediction for the selected city during the selected days so that I can choose the most appropriate places to visit.
---------------------------	---

Table 2: User Stories

11.2 Application Views

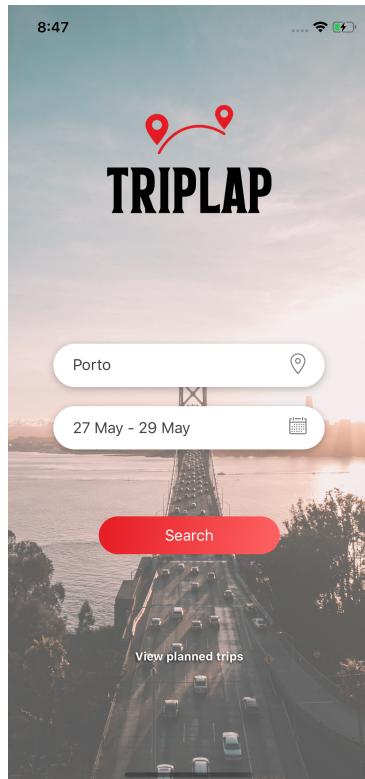


Figure 2: Search screen

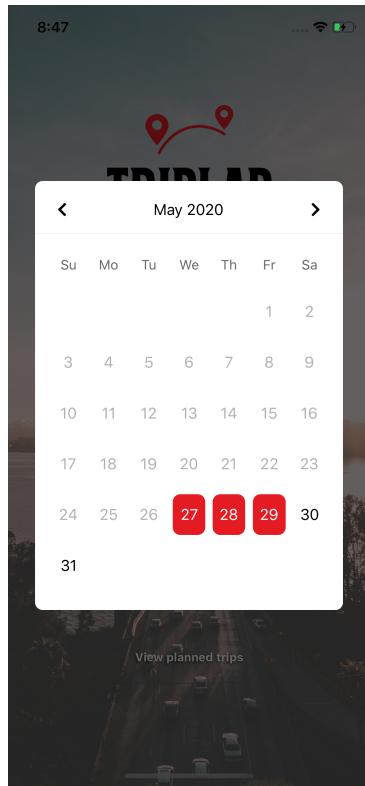


Figure 3: Select date range pop-up

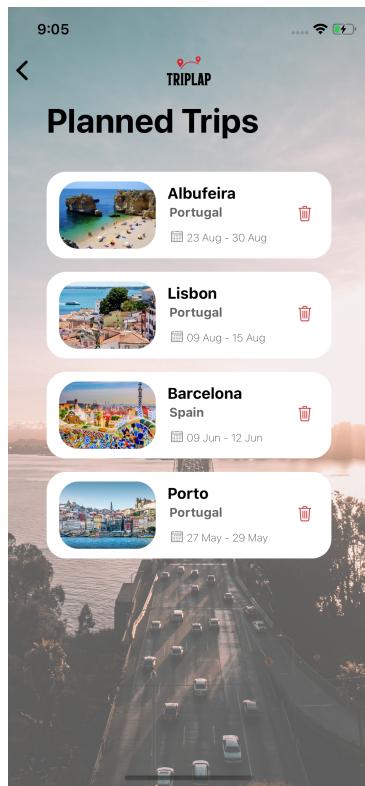


Figure 4: Planned trips screen

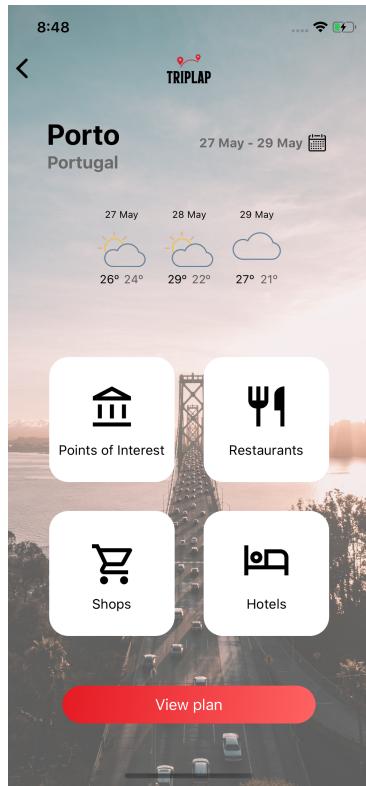


Figure 5: Trip main screen

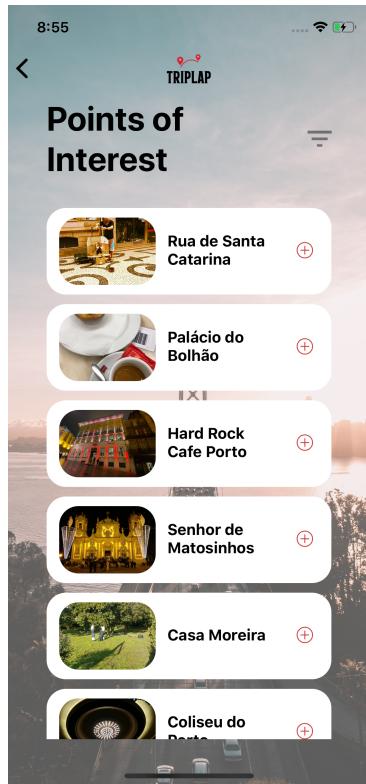


Figure 6: List of points of interest screen

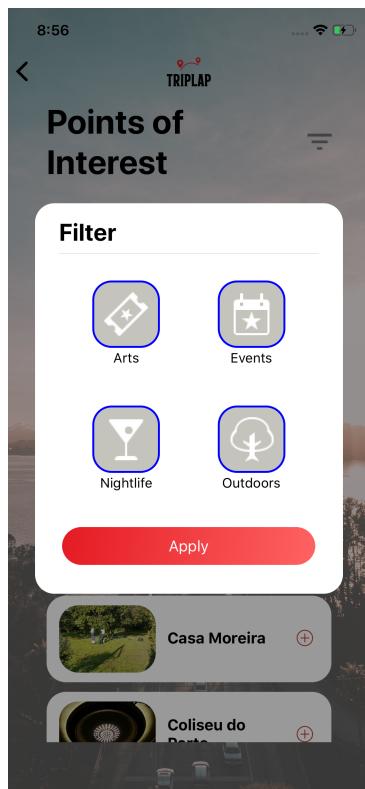


Figure 7: Filter points of interest pop-up

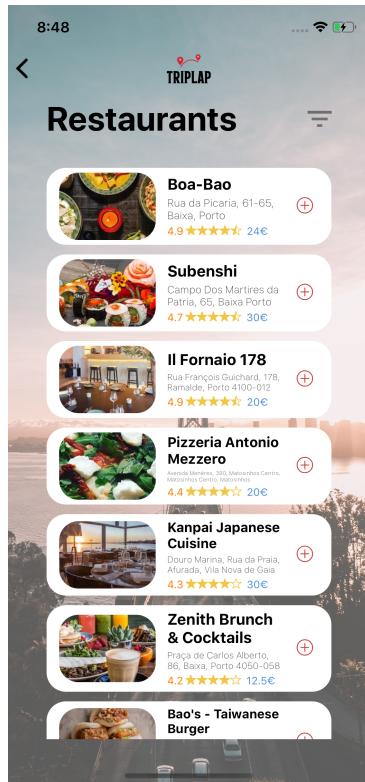


Figure 8: List of restaurants screen

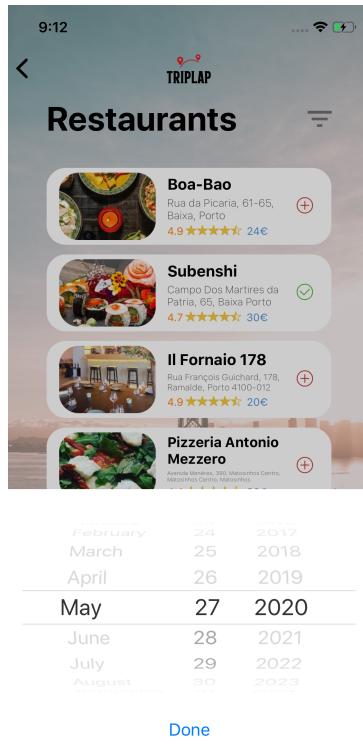


Figure 9: Select date for a place pop-up

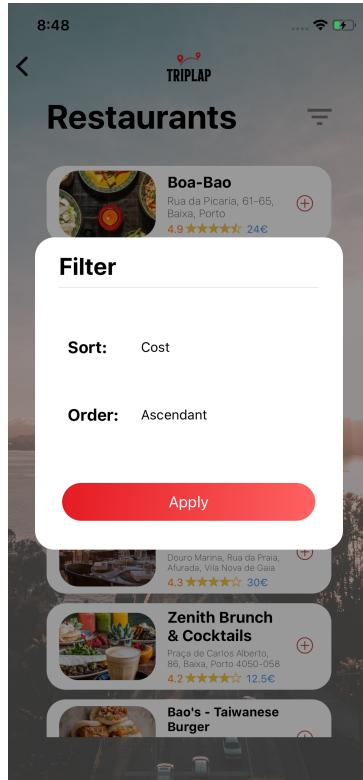


Figure 10: Filter restaurants pop-up

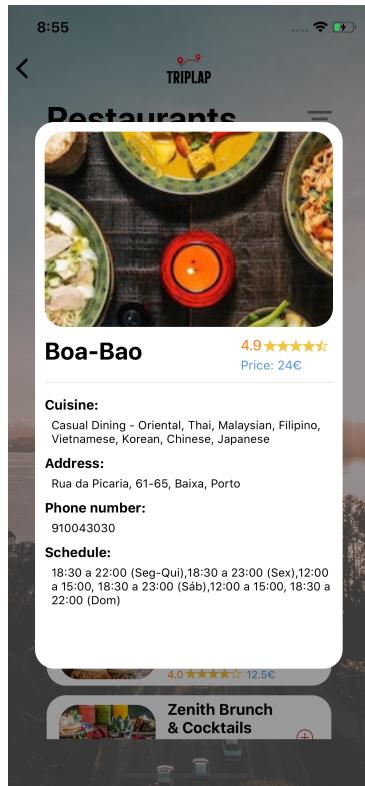


Figure 11: Detailed information of a restaurant pop-up

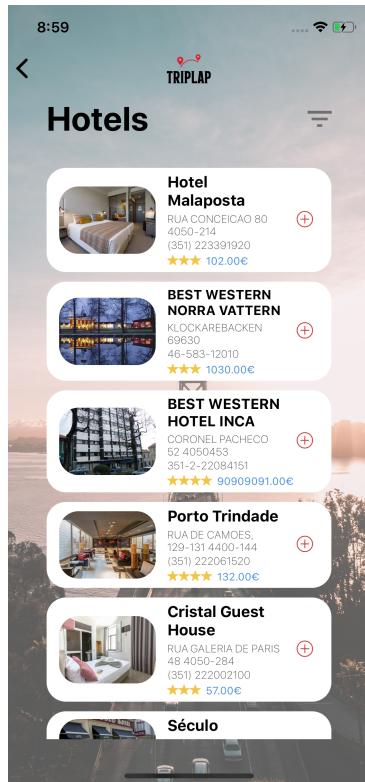


Figure 12: List of hotels screen

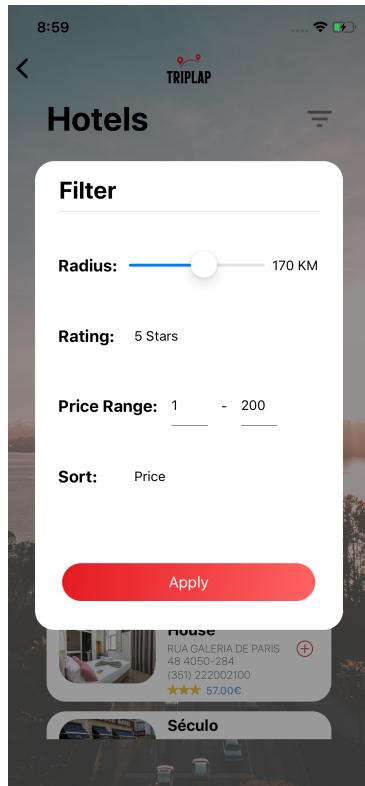


Figure 13: Filter hotels pop-up

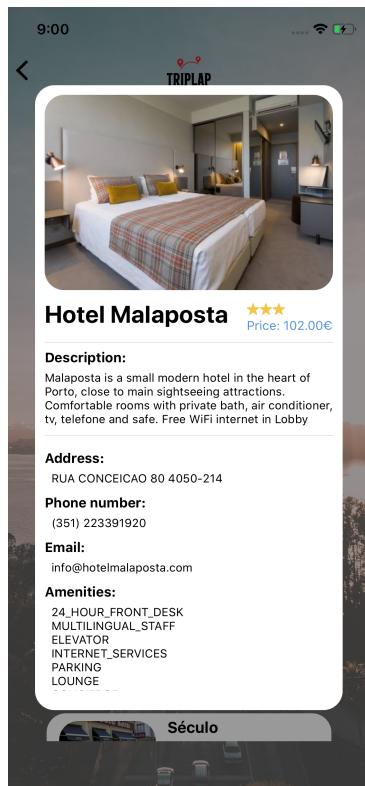


Figure 14: Detailed information of a hotel pop-up

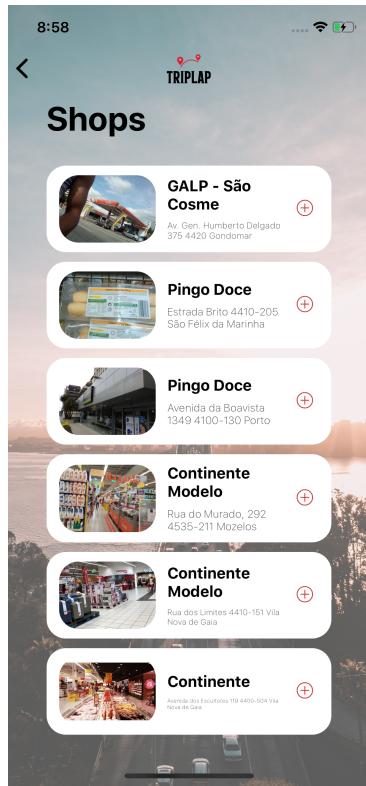


Figure 15: List of shops screen

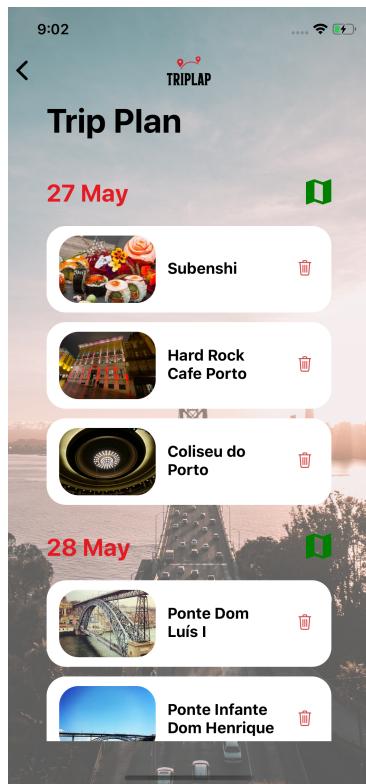


Figure 16: Trip plan screen

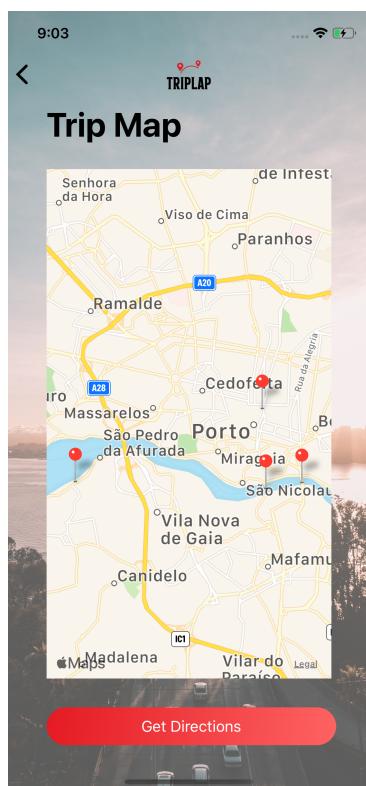


Figure 17: Trip map screen