
Software Requirements Specification

for

TravelGavel

Version 1.0 approved

Prepared by Joe Godinez

Kodiak Coders

02/14/2023

Table of Contents

1. Introduction	1
1.1 Purpose	1
1.2 Intended Audience and Reading Suggestions	1
1.3 Product Scope	1
1.4 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	3
3.1 User Interfaces	3
4. System Features	4
4.1 System Feature 1: Travel Logs	4
4.2 System Feature 2: Exploration of local restaurants and attractions	5
4.3 System Feature 3: Packing Lists	5
4.4 System Feature 4: Budgeting	5
5. Other Nonfunctional Requirements	5
5.1 Performance Requirements	5
5.2 Safety Requirements	6
5.3 Security Requirements	6
5.4 Software Quality Attributes	6

Revision History

Name	Date	Reason For Changes	Version
Joe Godinez	02/14	Beginning TravelGavel SRS	0.1
Whole Team	02/18	Developing TravelGavel SRS	0.2

1. Introduction

1.1 Purpose

The product described by this SRS is TravelGavel, of which the first version is being developed. While the vision for TravelGavel is complex and multi-functional, the focus of this document is on the MVP that can be accomplished within the given time-frame. The MVP includes the following features: travel logs, locating local attractions and restaurants, packing lists and budgeting.

1.2 Product Scope

TravelGavel is a vacation planning app available on Android devices. TravelGavel has everything you need to plan and enjoy your perfect trip. Book flights, hotels, and car rentals with ease. Use TravelGavel's packing list to make sure you don't forget anything at home. Research the best restaurants and attractions at your destination, while keeping track of your expenditures. Keep track of your plans with TravelGavel's travel logs. Stay connected with others by sharing your trip with friends or planning a trip with a group. Store all your digital needs, from reservation and tickets at your fingertips to pictures and videos to commemorate your journey.

1.3 References

<https://developers.google.com/maps/documentation>

2. Overall Description

2.1 Product Perspective

This app is being developed for a group-based college project. It is a new system that is not connected to preexisting software.

2.2 Product Functions

- Travel logs
- Exploration of local restaurants and attractions
- Packing Lists
- Budgeting

2.3 User Classes and Characteristics

- Users that travel for vacation:
 - Highly used features: travel logs, restaurant finder, attraction finder, picture/video saving, packing lists, budgeting

- Minimally used features: none
 - Other considerations: might be traveling as family (need to accommodate in packing lists)
- Users that share travel on social media:
 - Highly used features: multi-trip planning, travel log sharing, picture/video saving, packing lists, restaurant finder, attraction finder, budgeting
 - Minimally used features: none
 - Other considers: will most likely be younger (more tech-savvy)
- Users that travel for work:
 - Highly used features: multi-trip planning, packing lists, restaurant finder, budgeting
 - Minimally used features: travel log itinerary/sharing, attraction exploration, picture/video saving, travel logs
 - Other considerations: will most likely be older (less tech-savvy)

2.4 Operating Environment

TravelGavel is a mobile Android app. It is being developed with SDK x and requires an Android device running at least Android 9.0 (Pie).

2.5 Design and Implementation Constraints

This project is constrained by the time limit of the semester. We must maintain the users' right to security and privacy by asking permission to access and use their data for the features of the app. This is an Android app, so its code and database must be compatible with Android standards.

2.6 Assumptions and Dependencies

It is assumed that the project is constrained by the availability of Google's maps software, however it is a rare occurrence for that software to be shut down.

3. External Interface Requirements

3.1 User Interfaces

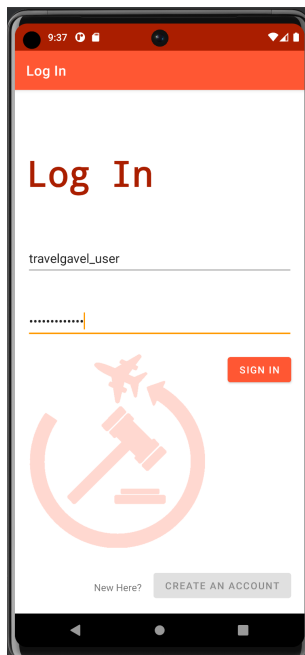


Figure 1: This image shows a mock-up of logging into an existing user account. There are plain text input fields for the account username, as well as an input field for the user's password. Additionally, there is an optional button at the bottom of the UI which leads to the "Create New Account" interface.

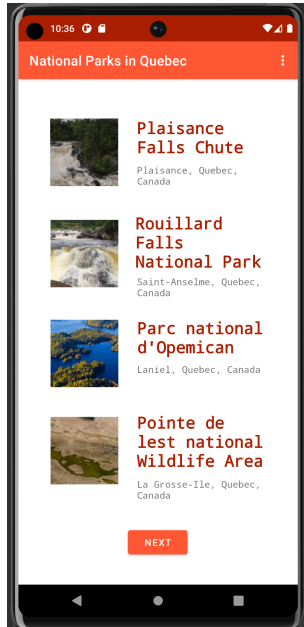


Figure 2: This image shows an example search return for a user looking for national park attractions in Montreal. Four example suggestions are displayed, with park names and locations formatted as headline text next to an image button which the user can tap to view a detailed description of the location. The user has the option to tap “Next” to view more suggestions other than the four displayed.

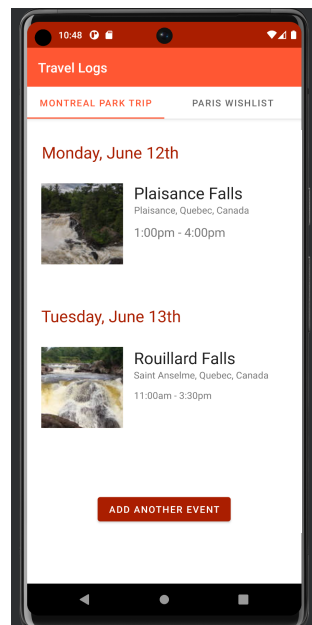


Figure 3: This image shows an example of a travel log. Two items have been added to the “Montreal Park Trip” travel log, and the details of each event are accessible through an image button displaying a picture of the location. Towards the top of the interface is a scrollable tab group containing the titles of any other travel logs that have been created and saved by the user. Within a given travel log, the user is also able to add another event by selecting the “Add Another Event” button at the bottom of the interface.

4. System Features

4.1 System Feature 1: Travel Logs

4.1.1 Description and Priority

Travel logs display important events and times those events occur for the user's travel plan. Expansion of this feature will include sharing options, multi-user editing of a single travel log, saving of important documents like tickets, saving of pictures and videos, and downloading a travel plan for offline access. It is a High priority feature.

4.1.2 Functional Requirements

- REQ-1: The travel app shall allow users to export their itinerary as a spreadsheet.
- REQ-2: The travel app shall sync travel log times with the user's calendar.
- REQ-3: The travel app shall display the local time of locations in the user's travel logs.
- REQ-4: The travel app shall allow friends to have editing access to a travel log.
- REQ-5: The travel app shall allow the user to share their itinerary to social media.
- REQ-6: The travel app shall allow users to record a list of items for their trip.
- REQ-7: The travel app shall allow users to opt in to sharing their travel data with 3rd party users.

4.2 System Feature 2: Exploration of local restaurants and attractions

4.2.1 Description and Priority

Exploration of local restaurants and attractions allows users to find places to go at their destination of choice. Expansion of this feature will allow users to filter their search based on price, reviews, and other preferences. Future implementation will allow the user to make reservations and pay for tickets through the app. It is a high priority feature.

4.2.2 Functional Requirements

- REQ-1: The travel app shall display restaurants within 75 kilometers of the user.
- REQ-2: The travel app shall record the location of the user.
- REQ-3: The travel app shall allow users to store preferences for types of restaurants they want highlighted.
- REQ-4: The travel app shall show the current weather for locations on their itinerary.

4.3 System Feature 3: Packing Lists

4.3.1 Description and Priority

Packing lists allows users to track what they will bring for their trip. Expansion of this feature will include app recommended packing lists, that will be dynamic based on expected weather, planned activities, and trip length. This is a medium priority feature.

4.3.2 Functional Requirements

- REQ-1: The travel app shall provide the user with a recommended packing list.
- REQ-2: The travel app shall adapt the recommended packing list to match the expected weather.

4.4 System Feature 4: Budgeting

4.4.1 Description and Priority

Budgeting allows users to plan and track their spending for their trip. Expansion of this feature will include using the local currency of the user's destination. Future implementation will include syncing with purchases made through the app. This is a medium priority feature.

4.4.2 Functional Requirements

REQ-1: The travel app shall allow users to track their budget.

REQ-2: The travel app shall convert currency for locations on their itinerary.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

REQ-1: The travel app shall remain under 50 MB in size.

REQ-2: The travel app shall have a response time of less than eight seconds under the typical workload 99% of the time.

REQ-3: The travel app shall become available within 120 seconds of a crash 99% of the time.

REQ-4: The travel app shall save its state when interrupted by a call 95% of the time.

REQ-5: The travel app shall be able to maintain optimal performance 10,000 visits 95% of the time.

REQ-6: The travel app shall have error rates for viewing itinerary which must not exceed 10%.

5.2 Safety Requirements

5.3 Security Requirements

REQ-1: The travel app shall require each user to have a 12 character password when signing up.

REQ-2: The travel app shall encrypt the user's entered information.

5.4 Software Quality Attributes

REQ-1: The travel app shall be compliant with Google Play Store requirements.

REQ-2: The travel app shall have a date format as follows: month.date.year.