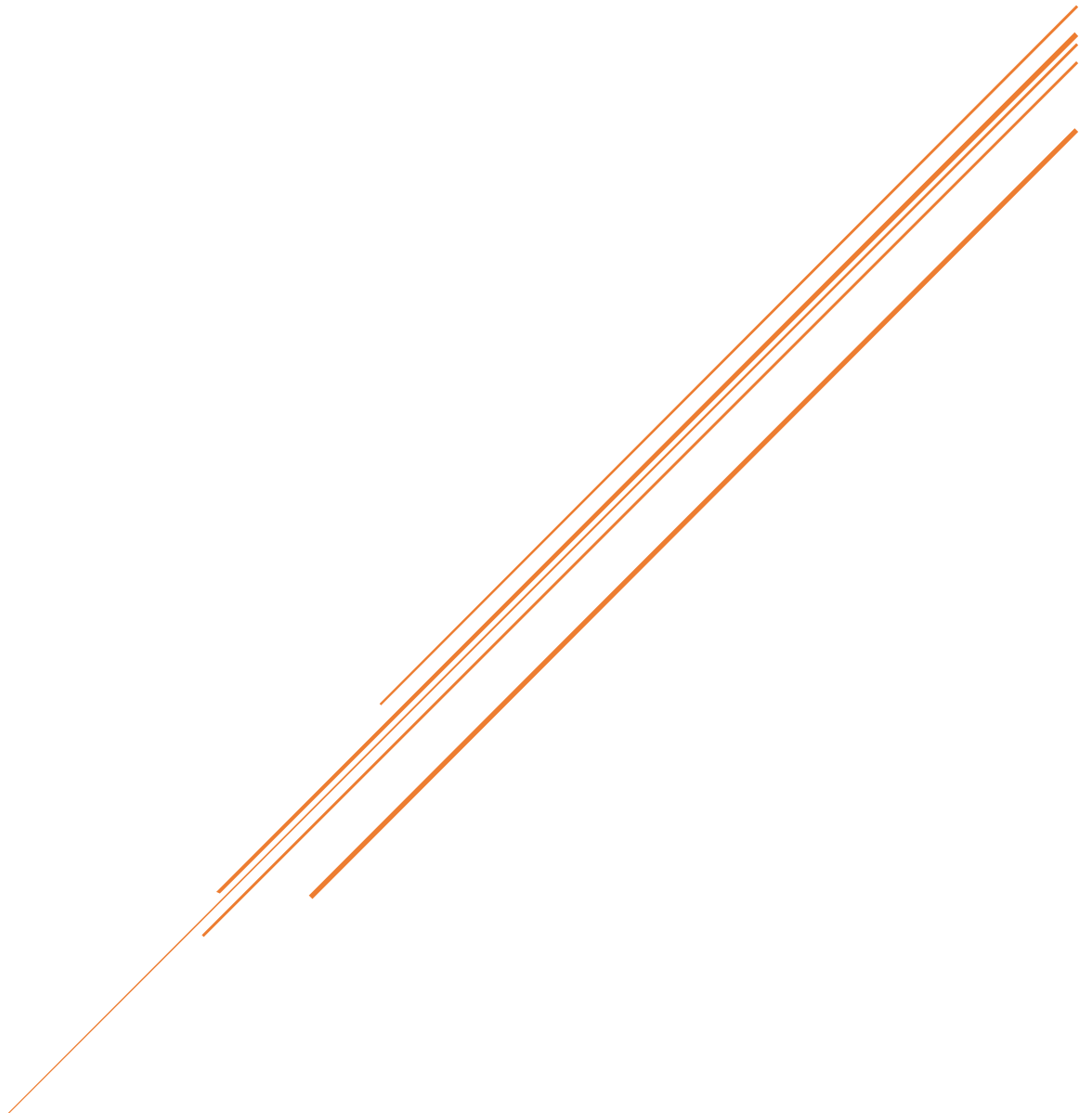


REQUIREMENT ANALYSIS AND WORKFLOW DOCUMENT

SOFTWARE REQUIREMENT SPECIFICATION (SRS) v1.0

BEST WEATHER APP [REACT NATIVE (IOS, ANDROID), ADMIN PANEL (WEB), BACK-END]



INTRODUCTION

PROJECT TITLE

This document includes requirements and Scoping for development of **Best Weather App** for iOS & Android platform using React-Native. The development will comprise on developing Mobile App using React-Native which will yield iOS and Android apps. The system will accompany an Admin Panel and its functional requirements are enlisted in Scope. Furthermore, this complete system will be accompanied with a back-end for Mobile Apps and Admin's connectivity and synced functioning.

PROJECT DETAILS

OVERVIEW AND SCOPE

The primary scope of Best Weather App is to enable the end user to view Best Locations with respect to Weather of those locations (and prioritized via Admin). The app will accompany the users in selecting any location in the app that they find suitable and would like to find out the details of that destination. The app will help the user to get idea of calculations on the following schemes:

1. Flights
2. Hotels
3. Cruises
4. More detail about Location
5. Weather Updates of that Location with a weekly forecast

Moreover, a feature named “Guru” will be able to help user to get estimates for a certain tour.

The app will not include any bookings or payments, this app is meant for estimation to certain locations enlisted within the app or to which user want to travel.

Further details of the project are mentioned below:

SCOPE

The scope of this system includes these things:

1. MOBILE APP (IOS, ANDROID)

- 1.1. Login via Email/Facebook/Google
- 1.2. Show “Top 10” Places [With Temperature, Price for Visit, Image, Name]
- 1.3. Details includes:
 - 1.3.1. Name and Country of that Place
 - 1.3.2. 1 Week Weather Forecast
 - 1.3.3. Flights (button)
 - 1.3.3.1. Open Popup
 - 1.3.3.2. Request Data from User (e.g.; Round Trips, One Way, Flying from/to, Depart/Return Date)
 - 1.3.3.3. Data will be requested from **Orbitz**
 - 1.3.4. Hotels (button)
 - 1.3.4.1. Open Popup
 - 1.3.4.2. Request Data from User (e.g.; Hotels/Airbnb, Filters, Check in/out, etc.)

-
- 1.3.4.3. Data will be requested from **Orbitz**
 - 1.3.5. Cruises (button)
 - 1.3.5.1. Open Popup
 - 1.3.5.2. Request Data from User (e.g.; Going to, No. of Travelers, etc.)
 - 1.3.5.3. Data will be requested from **Orbitz**
 - 1.3.6. More (button)
 - 1.3.6.1. Description of Place
 - 1.3.6.2. Activities – Will show respective Activities
 - 1.3.6.3. Maps – Will show Location over Google/Apple Maps
 - 1.3.6.4. Gallery – Will show Images of that place
 - 1.3.6.5. Share – Shareable through Text
 - 1.4. About
 - 1.5. Filters (Will allow saving Filters based on following parameters)
 - 1.5.1. Original Settings OR My Settings
 - 1.5.2. Select between Weather | Water | Location
 - 1.5.3. Temperature Range
 - 1.5.4. [Filter will be saved if the User is Logged-in Already. Non-Signed-In user will not have option to save Filter.]
 - 1.6. Favorites
 - 1.7. FAQ
 - 1.8. Contact Us
- ## 2. ADMIN PANEL
- 2.1. Admin will be able to Login using valid Email Address and Password
 - 2.2. Admin will be able to:
 - 2.2.1. Manage Users
 - 2.2.2. Manage “Top 10” places using list of places from “**Orbitz**”
 - 2.2.2.1. Admin will be able to Add Place Name. That place will be searched through Orbitz and its relevant details will be fetched. (Need to verify with Orbitz regarding place verification)
 - 2.2.2.2. Admin will be able to Add Images (for Gallery on Mobile App), Description of Place, Lat/Lng (to show on Map), Activities. Once Added/Edited, that will be saved to system and will be available in Mobile App.
 - 2.2.2.3. Admin will be able to Sort Current Places in “Top 10” List.
 - 2.2.2.4. Admin will be able to Add/Remove Places from “Top 10” List.

TECHNICAL DETAILS

BACK-END AND APIS

In addition to APIs that will be developed to manage App's connectivity with Admin Panel and Back-end, we'll be using the following 3rd Party APIs:

1. **Orbitz** (Can be accessed here: <https://hackathon.expedia.com/docs/public/api/>)
[Needs to request access to them on their official Expedia page]
2. **Weather.com** (for fetching Weather Forecast details)

Both of these API's access will be shared by client. Once approved to be used, further exploration of APIs and its integration in the app will be carried out.

TECHNOLOGY STACK

Following tech-stack will be used for Development:

1. Mobile Apps: React-Native
2. Back-end: Laravel
3. Admin Panel's front-end: Bootstrap (HTML5, CSS3)

MILESTONES & TIMELINE

No.	Milestone Division	Milestone Description	Timeline	Expected Date of Delivery
1.	Milestone 1	Design Integration in Mobile App	1 – 2 Weeks	March 27, 2020
2.	Milestone 2	Back-end Development, Admin Panel Development, Orbitz API Exploration + Integration in Back-end & Admin Panel	2 – 3 Weeks	April 17, 2020
3.	Milestone 3	50% APIs Integration in Mobile App	2 – 3 Weeks	May 8, 2020
4.	Milestone 4	100% APIs Integration in Mobile App	2 – 3 Weeks	May 29, 2020

A total of 6 – 7 Weeks will be required to complete the React-Native App (Android + iOS) and Back-end along with “Orbitz” API Integration.

The mentioned Expected Date of Delivery holds if Development begins from *March 13, 2020*. Furthermore, only working days are calculated in Expected Date of Delivery (Saturday & Sunday are excluded from the dates mentioned). Any Public Holidays or leaves announced will be communicated beforehand to make sure client is updated with current situation of development.