



v.1.4

## Scope of Work: HavRez App

## Confidentiality of Important Information

The information in this Document is confidential and is intended solely for the attention and use of **Ms. Jody Rogers**. It might contain privileged information. If it has come to you in error and you are not the intended recipient you must not proceed further, disclose, copy, use or disseminate any information contained therein, please delete it and contact us (**sakshi@sdi.la**) without delay so that we may take whatever action we consider appropriate. Although this document is believed to be free from any virus it remains the responsibility of the recipient to ensure that this document is virus free and we accept no responsibility in this regard.

## Table of Notifications

This Table represents the contacts in both companies as assigned currently.

Project	Name	Company
Project Requirements	Jody Rogers	HavRez
Client Interface (Business)	Sakshi Sharma	Software Developers Inc.
Client Interface (Technology)	Raj Srivastav	Software Developers Inc.

## Aim

- To build a restaurant finder app which will allow users to find a restaurant based on their preferences (Profile) instantly in real time. This app will be modeled on Cognitive and Standard behavior.
- The users will be provided an algorithmic result based on answers to some question and based on their tastes and preferences the app will recommend restaurants.
- The users can select a restaurant from the recommended results. Optional features like reservations may be available. They can also skip the recommendations and start search again.
- This document contains all of the suggested main modules/functions requirements.

## Development Principles

- Mobile app for iOS platform (iPhone)
- Combination of latest UI and UX principles to provide a clean, Intuitive Interface for the HavRez app
- Smart Navigation Tabs for easy and fast access to all functions
- Language of Development – English
- Language of Data Entry - English

## Coding Standards

- Apple iOS SDK's will be used to create the code and user interface design.

## Requirements from client

- **Client will provide upto 50 questions which can be in a series or in sets.**  
**Note : The client can provide these in an excel spreadsheet or can directly add these to the database through the backend interface.**
- We will create an algorithm/logic which will pop-up questions based on users responses (Cognitive behaviour)

## FEATURES/ FUNCTIONS:

### MODULE 1 : App Features/Functions

1. Sign In with your Email and Password or connect via your Facebook account.
2. Sign Up if the user does not have a Facebook account
  - 2.1. Name
  - 2.2. Email
  - 2.3. Password
  - 2.4. Confirm password

Note : First time users will have an option to skip the sign up step to check out the app.
3. **Complete profile: After the user signs up, “Complete your profile” screen populates. It will help the app to understand the user’s basic tastes and preferences. These will be provided by the client and we will include them into the app/back-end.**
4. Find a Restaurant

- 4.1. Click on the “Find a restaurant” button which will start a round of questions.
- 4.2. User will select a response and move to the next question.

Note: The users cannot skip the questions. They have to provide a response in order to move forward. The questions will be based on an algorithm which will be sequenced based on the aspirational profile, the next question will appear only after and if it is needed based on the previous answer.
- 4.3. The app will provide an option to the users to select a radius within which they would like to dine in. For example 5miles, 10miles or 30miles.

Note : The app will recommend restaurants based on the users geolocation and selected criteria.
- 4.4. **A list of restaurants will pop up as soon as the questions round is over. Each restaurant will also display an image next to its name.** The list may consist of 3 different restaurants. If users are not satisfied with the best recommended restaurants then they can swipe to the next best recommendations.
- 4.5. The recommended restaurants will provide following pieces of information:
  - 4.5.1. Restaurant name
  - 4.5.2. Attributes (which will be pulled from the database, these attributes are going to be 2 best dishes of that particular restaurant)

**The restaurants database can consist of unlimited restaurants (List of restaurants to be provided by the client in a spreadsheet) and upto 50 attributes can be associated with each restaurant.**
- 4.6. Once the user selects a restaurant the following information shows up:
  - 4.6.1. Image icon
  - 4.6.2. Address (map integration to get directions)
  - 4.6.3. Contact number
  - 4.6.4. Cuisine type
  - 4.6.5. Reservation option to make a reservation

**We will Integrate up to 3 APIs from different providers if needed to provide easy restaurant reservations.**

- 4.7. If the app couldn't generate results based on users response then a message will popup saying "Sorry, we couldn't find any restaurants based on your preferences."
- 4.8. The next screen after the above message will display a default list of restaurants nearby users current location (users can change the location if they would like to search for restaurants in a specific location)

Note: The app will show results based on the backend database.

- **The app will pull the GPS location data of the user, convert the data into geo-coordinates & then query the location field in the restaurants database table.**
- **It will then filter the above results by comparing the user data stored in the user profile database table & user preference attributes database table (quiz answers fields, food preference field etc ) with the restaurants attributes (rankings field (based on the quiz), cuisine field) database table to provide the results as per users profile.**

We will feed the database in a way that users will not be able to search restaurants for locations that do not exist in our database. We will acknowledge their request with a message popup "Sorry, we do not serve this location at the moment".

- 4.9. Trapdoor - If a user visits or is in the vicinity of the restaurant and decides to not go/eat at the restaurant, then he can click on the "Trapdoor" icon and then he will be presented with another set of recommendations. If the visitor decides to visit another non-recommended restaurant then he/she will have the ability to add that restaurant to the app
- 4.10. Favourites - User will be able to mark restaurants as "Favourites" and he will be able to view the Favourites list.

## 5. Settings

- 5.1. Edit profile
- 5.2. Edit preferences
- 5.3. Change password
- 5.4. Delete account

## 5.5. Signout

## 6. Help/ Info

### **MODULE 2 : Web Services**

#### **App Features/Functions for the Admin**

1. Login/ Logout
2. Dashboard
3. Add/edit/delete questions {The algorithms (understanding and creating a logic in terms of what will be the next question and matching the results) will be created and the code will be written accordingly in order to find the restaurant}
4. Manage and control users access to database.
5. Add/ Modify/ delete accounts/ users profile.
6. Users can submit a restaurant and also send comments.
7. Customer loyalty module.

#### **Other salient points**

- We will launch the app on a local (Specific cities) basis initially based on the clients preferences. As more data becomes available, more cities can be added.
- The app will have an integrated Flurry based analytics. This will provide insights into user behavior and app usage data.
- We will provide a 3 -4 page static website for company contact information and app features. This is a requirement to be fulfilled before launch as per Apple's SDK guidelines.
- In-App payments can be integrated at a later date as an optional feature.
- We will focus on designing the app to increase loyalty.
- The design will reflect familiarity with other popular restaurant apps for easy adoption.
- The app will have geo-location identification.

## Investment details - Time & Cost

Tasks	Resources	Duration	Cost
Wireframes and designs	1 Designer	1 month	\$3200
iOS App development	1 iOS developer	3 months	\$12000
Backend + Web services	1 PHP programmer	3 months	\$10500
Testing	1 QA Engineers	1 month	\$0
Beta Launch		1 week	\$0
		<b>TOTAL</b>	<b>\$25700</b>

FYI: Testing and beta launch are the final stages of project completion where we will test the app and make sure that everything is working smoothly without any issues. There is no charge for these services. We have listed this to keep you informed about the time it will require.

### **Payment and revenue sharing terms:**

- 50% upfront payment at contract signing.
- 50% on Beta delivery.

### **Project Management - Collaboration/Review/Feedback**

- We use Active-Collab for Project Management/communications. You will be provided with login credentials and you can provide your feedback, information, and/or ask questions if any. You can communicate with the team of designers and developers who will be working on your project.
- The process will start by going through the scope of work with team. We will provide you with a project plan which will include a list of tasks with their scheduled completion deadlines.
- The project manager will have a meeting with you when you are ready and we will begin work on the wireframes and design work and thereafter regularly update you and receive feedback and suggestions until you approve the designs.

- Once the designs are approved, we will move to the development phase. The team will work on the frontend and backend in parallel. We will have regular meetings and schedule the meetings as per your convenience.
- You will have daily access via phone, email, skype, and gotomeeting to directly communicate with the Project manager and/or other team members.
- When the team starts testing the app, we will provide you the build which you can test on your respective device. We will ask you to share your iPhone device UDID. You can provide your additional UDIDs as well to test the app among your friends and relatives before it goes live.
- All deliveries are covered by a 6 months debugging warranty after delivery.