49786 Software Engineering Management

Milestone 1 Delivery, Team LiteWait, Sept 26th, 2018

This document is the milestone 1 delivery for the product LiteWait, entailing the requirement definition, UI Design, and Architecture Design.

1. Stakeholders and team members

Roles	Stakeholders
Angel Investor	Catherine Fang
Product Manager	Siva Shankaran Vasanth (Shiva)
Ideator and Engineering Lead	Joseph Ho (Joe)
Project Manager	Yuki Wang
Design and Experience Lead	Yizhong

2. User Stories (MVP)

View to the Complete Product Portfolio of LiteWait, here. Following are the user stories that will be accomplished in the minimal viable product (MVP). Prioritization to Sprint Map - Sprint 2: P0 Sprint 3: P1 Sprint 4: P2 + Demo.

User Story	Priority	UserStory Title	UserStory Description	Acceptance Criteria
US 1001	P0 EPIC	[EPIC] Splash Screen and App Landing Page	This EPIC designs and creates the splash screen. This page should appear when user clicks app icon on the operating system launcher.	A desirable Splash screen with elegant Logo, that stays for 2 seconds and leads to list view (List view render not part of story)
US 1103	P0	User views Splash Screen	As a user, I would like to view the home page after the splash screen.	1. Logo Design 2. Design Layout of Splash Screen 3. App Launch development 4. displayed for max 2s
US 1002	P0 EPIC	[EPIC] Home Page - View Nearby Restaurants and Corresponding Waiting Times	This EPIC designs and creates the Home Page which enables user to view nearby restaurants, the corresponding waiting times and links to these restaurants profiles (implicitly). The waiting times on this view page should update every min.	A convenient home page to view nearby restaurants based on current location of user 15 miles around the location - with structured views of waiting times.
US 1105	P0	User is able to view the Home Page	As a user, I would like to view the home page after the splash screen.	1. Appealing Design Layout for with a background and layout.
US 1107	P0	User is able to view the top ten restaurants and view as scrollable with max of 50 restaurants	As a user, I am able to view the top 10 restaurants near me so that I am aware of what restaurants I can choose to view. I would like to use scrolling feature to view more restaurants nearby.	1. Load the top 10 restaurants near the current location. (Demo). 2. Show variations by changing location (mock up).
US 1108	P0	User is able to view the waiting time of all the restaurants in the nearby restaurant list layout on	As a user, I would like to view the waiting time of the top 10 restaurants nearby. On scrolling, I would like to see the waiting time of the loaded restaurants.	1. Load the waiting time for each of restaurant on the table. 2. Explain the Design logic behind storing the waiting time (DB, Cache, Keys, etc)

		home page.		
US 1005	EPIC P1	[EPIC] User can view Restaurant Page and view the details and current waiting time.	This EPIC allows the user to exit the application either by gesture or by a click of a link available on the pages of the application.	Requires Decision - TBD
US 1113	P0	User is able to view restaurant page with restaurant name, with address, with location displayed	As a hungry user, I would like to click the restaurant of my choice on the home page, and view the corresponding restaurant page to see waiting time and details about the restaurant.	1. Appealing Design Layout of Restaurant Page 2. Demo -User able to view Restaurant details such as name, address, basic cuisine details, website link to restaurant or yelp and waiting time (refreshed every 1 minute).
US 1114	P1	User is able to go back to home page from restaurant page.	As a choosy user, I would like to go back to home page to view the restaurants again.	1. Demo redirection to home page from restaurant page
US 1006 Epic	EPIC P1	[EPIC] User can update observed waiting time if nearby and waiting time of restaurants may be updated.	This EPIC enables the user to enter waiting time if found nearby.	1. Demo to check if user is near the restaurant of the restaurant page 2. Demo to check if user can update the waiting time 3. Show how waiting time of restaurant is updated/not updated based on logic used to calculate cumulative waiting time in the last 1 min.
US 1115	P1	User clicks update waiting time and system senses the current location of the user and shows error if not nearby	As the system, it is necessary to obtain current location of customer, so as to decide whether to display waiting time entry fields for the customer.	1. Console Logic to show how it is determined to display the waiting time entry or not to the customer 2. Demo - on application this experience
US 1116	P1	If user is nearby, User is able to enter the waiting time and the waiting time is calculated	As a collaborating user, I would like to update the waiting time of the restaurant with necessary information on how to arrive at that value, and view any changes to the waiting time if deemed as logical.	1. Demo waiting time entry and updation of persisted data or calculated data 2. Demo page refresh with updated time with console log showing calculation
US 1007	EPIC P2	[EPIC] User is able to sort the list based on waiting time for the restaurants	This EPIC enables the user to sort the nearby restaurant list on the basis of waiting time.	1. Demo the sorting experience.
US 1117	P2	Enable sorting logic on the restaurant list table	As a choosy and hungry user, I would like to view the 'nearby' restaurants with lowest waiting times.	1. Demo the sorting experience.

3. Project Timelines and Ceremonies for Release 1 (MVP)

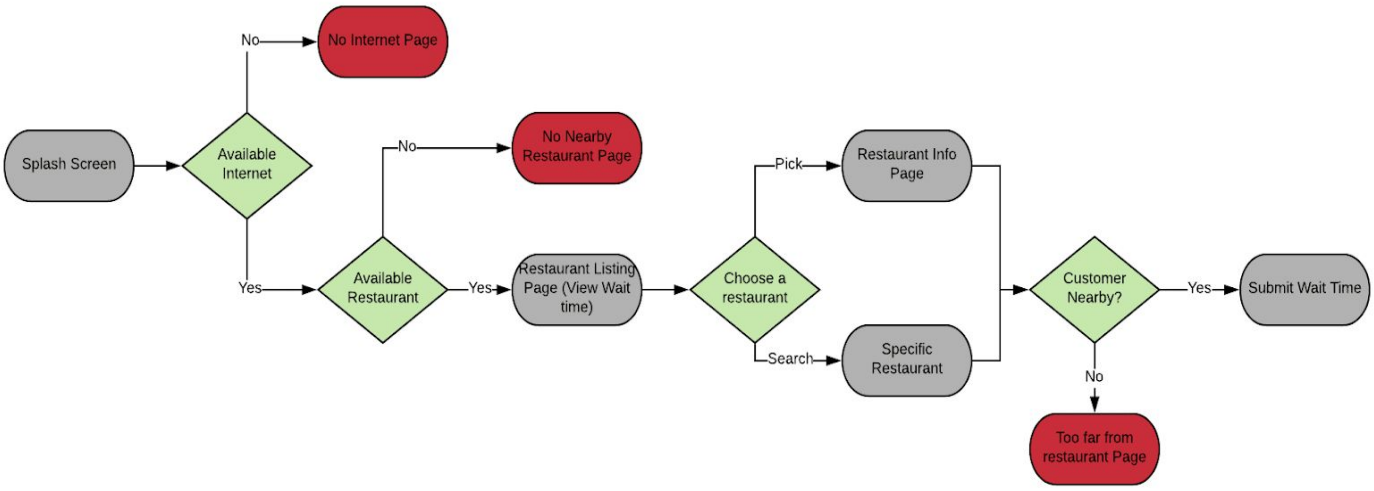
Sprints	Sprint TimeLine	Sprint Planning	Backlog Grooming	Sprint Demo	Sprint Retrospective
Sprint 1	Sept 13th to Sept 26th	Sept 13th	Sept 19th	Sept 25th	Sept 26th
Sprint 2	Sept 27th to Oct 10th	Sept 27th	Oct 3rd	Oct 9th	Oct 10th
Sprint 3	Oct 11th to Oct 24th	Oct 11th	Oct 17th	Oct 23rd	Oct 24th
Sprint 4	Oct 25th to Nov 7th	Oct 25th	Oct 31st	Nov 6th	Nov 7th
	Daily standup - 10 minutes everyday				

4. Resource Planning

Skill Set (1-3)	Joe	Yizhong	Shiva	Yuki
mvc design	2	2	3	1
mobile develop	2	2	3	1
Front End	2	3	2	1
Back End	3	3	3	1
PRD	2	3	3	1
DataBase	2	2	2	2
Swift	1	3	1	1
Andriod Studio	3	3	3	2

Time Availability	Joe	Yizhong	Shiva	Yuki
Total Hours Committed per sprint	15	15	15	15

5. Wireframe Design



6. Architecture Design

Layers	Structures	Technology
Presentation	Tools/Frameworks	Xcode/Swift
Middle	API's	Google Places API (Autocomplete API, Search, Place Details), Google Map API
Backend	Firebase	Authentication, Real-time (Cloud-hosted) database
	Others	Data Elements (JSON): Restaurant name, Location Details, Default Wait Time

7. External dependencies

7.1. Areas with Technical Challenges

Since only a few members have iOS experience, there will be some obstacles developing our product. More specifically, we expect to run into some technical challenges when combining each technical component, due to our lack of familiarity with the backend, and specific API usage.

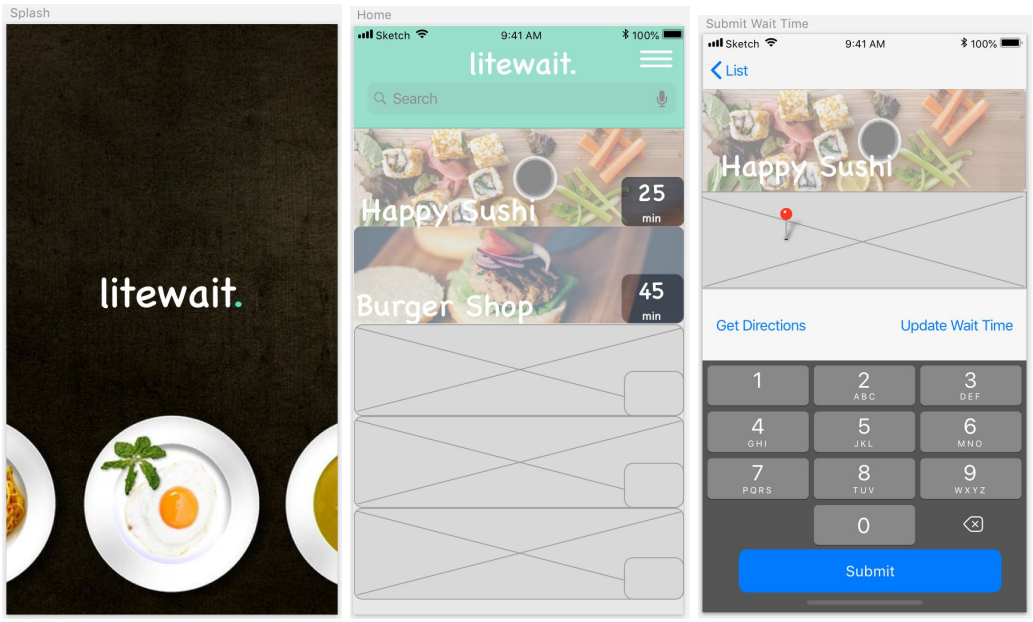
7.2. Areas not explored

Firebase is new to all of the team members.

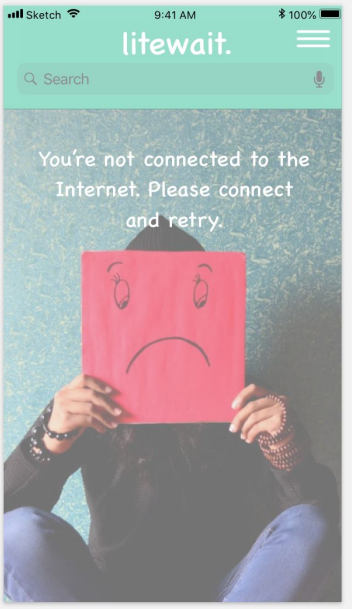
7.3. Areas with algorithms

In terms of the minimum viable product, are sorting by waitime and sorting by distance would be two algorithms we need to control.

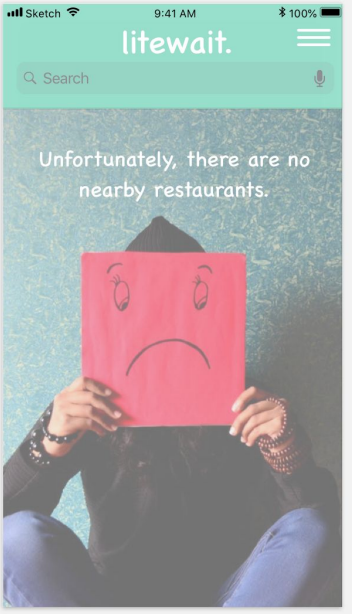
8. Optimized Mock-Frames



No Internet



No nearby restaurants



Distance Error Popup