Hop (A Social Application)

by

Harper Lamb
Patrick Mileham
Zack Smith
Dylan Silbernagel

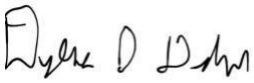
Submitted to
the Faculty of the School of Information
Technology in Partial Fulfillment of the
Requirements for the Degree of Bachelor of Science
in Information Technology/Cybersecurity

Harper Lamb _____04/24/2022

Zack Smith _____04/24/2022

Dylan Silbernagel _____04/24/2022

Patrick Mileham _____04/24/2022



_____04/24/2022

University of Cincinnati
College of
Education, Criminal Justice, and Human Services
April 2022

Table of Contents

Abstract

The Hop social application was created to solve problems that the group found when socializing on the weekends. People were not able to meet up with friends, or spent most of their evening trying to, because they didn't know which location their friends were at. When that person found their friends, they had to wait in a line outside the bar or restaurant to get in. After this individual got through the line and found their friends, they may have to wait again for a drink. At this point, their friends may be ready to move on to the next location. Hop has been created to solve these problems and to make going out on the weekends a more fun and simpler process. Our team researched these problems and spoke with people in our target market to better understand their problems and what type of solution they would like to see. We also researched competitors that had similar solutions or those that users had heard about to understand how we could differentiate our product. Our group members downloaded other applications and used them so we would have firsthand experience in what worked well and what could be improved on to solve our users' problems. We collected our insights as well as data received from conversations with target users. Our decision was to create an iOS application using the swift programming language, and the group had to conduct research to successfully utilize this language. As a group, we had no knowledge of this language, so we spent time researching the fundamentals and learning how the language works. We focused on how it interacts with other platforms and languages, and we learned how to use Xcode as our development environment. This research took a significant amount of time as swift is a fairly new language. Apple does have resources that we used and referred to as we worked on creating the application. After completing research and work on understanding the coding language, development platform, and other platforms used in the project, the group moved to look at the features that would be included. The research focused on wait times at bars and restaurants and how that could be minimized. The group also looked into different APIs and resources for implementing the features. We worked in Google Developer to understand their location data API and how that could be implemented into our application. We also completed research on how we could get another user's location data, a friend the user has added, to display on their map. This was a more difficult task that led to the decision of supplying friend location data through their status. The majority of our research was completed based on each group members' experiences and through talking with other people at bars and restaurants. We created Hop as a solution to these problems and as a way to make going to bars and restaurants easier. Our solution makes the experience easier and more efficient for users while allowing the bars to benefit through promotion and partnerships.

Introduction

Project Summary:

Hop is a social media application that lets the user find friends, deals, wait times and bars all from an application on their phone.

Problem Statement:

People on college campuses are constantly waiting in lines or trying to contact their friends to see which bar to go to. They are also trying to find the best deal that night while figuring out where they will have the most fun. These issues can lead to lack of communication and wasted time waiting on people, lines to clear, or going from bar to bar. There are applications (Find My Friends/Beer with Me) that help find your friends, but they only offer the location. There isn't an option to understand lines, wait times, or where your friends might be headed next. There are other applications (No Wait) that let you get in line on your way, but don't tell you where your friends are or if there are deals offered at that location.

Solution:

Hop is an application on your phone that will report wait times to users, which bar users' friends are at, and drink prices along with drink menus. Each bar location can promote different events on their specific page for example, a karaoke night or if part of the bar is rented for a private event. To eliminate wasting time or wondering where your friends may be, all the user would do is check HOP which gives them vital information before going out. This application would not only be helpful to the users, but it will also be helpful to the bar managers and owners as well. The app would be able to give bar managers a good idea of the amount of people that are coming to their bar on a certain day. It will also show statistics based on certain days that reflect the success of their business or deals they are promoting. Hop allows users to see where their friends are, where they are going, where the best deal is, and the shortest wait time.

Project Source:

As a group we came up with this project as a solution to the typical weekend bar scene. Our group formed as we all have worked in class before together and thought we would work well as a group this year. The inspiration came from problems that people have when going out which includes where their friends are, if there is a line or wait time, if there are drink deals, and if the bar is even open.

Discussion

Project Objectives/Goals:

We hope our solution will help by making weekend plans easier and saving time for people that are going out.

Major Features:

- Location Services to find friends
- Map feature
- Real time wait tracking
- Messaging capabilities

Expected Impact:

- Easier weekend experience
- Connection between people
- Profit benefits and exposure for bars
- Exclusive deals for users

Project Scope:

Hop will allow anyone going to a bar to accurately assess line lengths along with different prices and selection of products. It will be able to accurately track this information from your phone in the comfort of your home.

Quick Project Timeline:

Task #	Task Name	Duration	Start Date	End Date
	Fall Semester			
1	Project scope and objectives meeting	1 hour	9/20/2021	9/20/2021
2	Design Phase I	1 month	9/21/2021	10/21/2021
3	Create app wireframes	1 week	9/20/2021	9/27/2021
4	Coding for location services feature	2 months	9/27/2021	11/27/2021
5	Coding framework from wireframes	2 months	10/4/2021	12/4/2021
6	Creation of use cases and use case diagrams	1 week	10/4/2021	10/11/2021
7	Design Phase II & III	1 month	10/21/2021	11/21/2021

8	Testing for location services	2 weeks	11/8/2021	11/22/2021
9	Coding for friends feature	1 month	10/18/2021	11/18/2021
10	Coding for login feature	2 weeks	11/15/2021	11/29/2021
11	Testing for friends and login features	2 weeks	11/29/2021	12/6/2021
12	Coding for friend location feature	1 month	12/6/2021	1/6/2022
13	Coding for real-time data	2 months	12/6/2021	2/6/2022
	Spring Semester			
17	Design framework implementation	1 month	01/10/2022	02/10/2022
18	Design Phase II	1 month	02/10/2022	03/10/2022
19	Design Phase III	1 month	03/10/2022	04/10/2022
20	Testing/Implementation	2 month	2/10/2022	4/10/2022

Technologies Used:

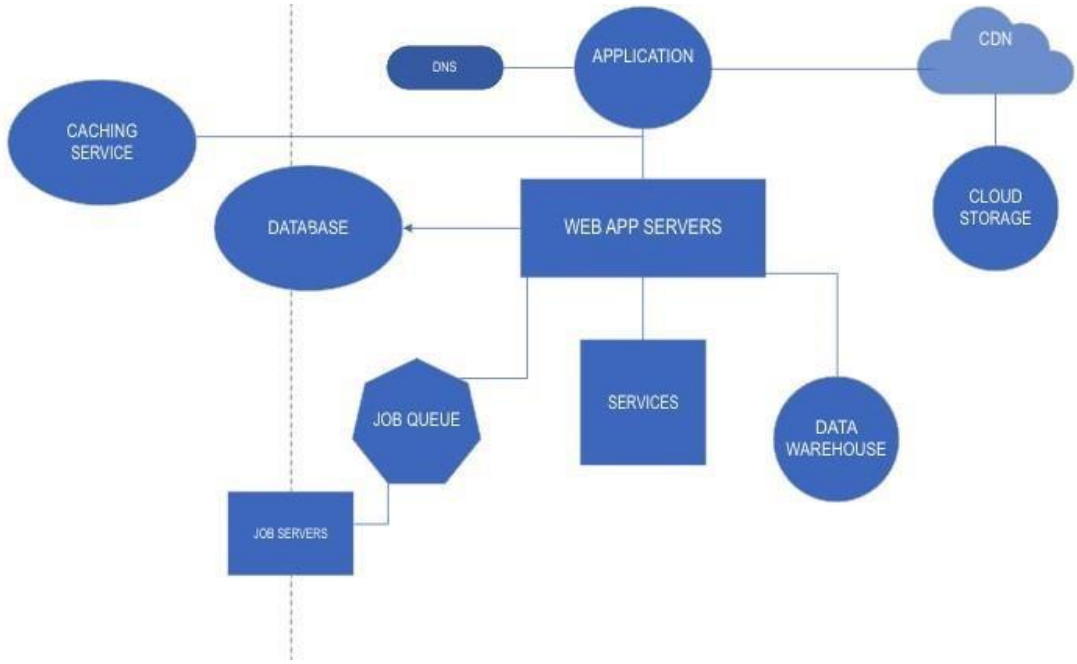
App Development:

- Swift
- Xcode
- GitHub
- SourceTree

App Hosting:

- Microsoft Azure
- Google Map Platform

Technical Architecture Diagram:



User Personas:

Table 1 User Persona Table

User Persona: 1	
	College Student/Part-time employee
	Hunter
	23
	Male

Behavior	Hunter lives near campus and likes to go to bars on the weekends to meet up with friends. He also wants to find the best deals and happy hours to try and save money. He is a full-time student and works part-time so his budget is important in his decisions.
Pain	He works late so he doesn't always know where his friends are once he gets off work. He also is on a budget and is trying to save money by finding discounts.
Needs & Goals	Needs: - Easy way to find friends on the weekends - Way to find the best deals Goals: - Have a social life outside of work hours and school - Not waste time or money when going out to the bars

Table 2 User Persona Table

User Persona: 2	
	Bar Owner
	Jim
	42

	Male
Behavior	Jim has always wanted to own a bar, and his new bar is his full-time job. He is typically at the bar serving drinks or managing staff. Jim is open to ways to increase traffic and establish more regulars at his bar.
Pain	The number of people that come in is inconsistent each weekend depending on how busy they were the night before. He also gets complaints about long lines and wait times for drinks.
Needs & Goals	Needs: - To understand how to help frustrated customers - To create a fun and casual environment where people want to come back Goals: - To have a consistent, steady group of customers - To make customers feel better about wait times and decrease lines

Use Cases:

Table 3 Use Case Table

Use Case ID	001
Use Case Name	Create User Account
End Objective	To have a user account that saves information and allows the user to access their customizations.
User/Actor	App user

Trigger	The user enters login credentials into the create account prompt
Frequency of Use	This is a one-time event
Preconditions	User opens the Hop app
Basic Flow	1. App prompts user to login or create an account 2. User selects create account 3. User enters prompted information 4. User confirms and logs on 5. System saves the information 6. System takes user to home page
Alternate Flow	1. User opens the app 2. System identifies user saved information 3. User logins in from stored information
Postconditions	The app saves the account information and stores the user. The app knows the user by login credentials and saves their customizations.

Table 4 Use Case Table

Use Case ID	002
Use Case Name	User login
End Objective	To login using the saved credentials from saved information in the app
User/Actor	App user
Trigger	The user enters login credentials into the login prompt
Frequency of Use	This will occur every time the user opens the application to login. The user may choose to save their login and will only enter the credentials once.
Preconditions	User opens the Hop app

Basic Flow	1. App prompts user to login or create an account 2. User selects login 3. User enters login credentials 4. User opts in to app saving their information for quicker login 5. User confirms and logs on 6. System saves the information 7. System takes user to home page
Alternate Flow	1. User opens the app 2. System identifies user saved login credentials
	3. User logins in from stored information 4. System takes user to home page
Postconditions	The app saves the account information and stores the user. The app knows the user by login credentials and saves their customizations.

Table 5 Use Case Table

Use Case ID	003
Use Case Name	User adds friends
End Objective	To add friends on the app
User/Actor	App user
Trigger	The user selects the add friends option
Frequency of Use	Each user will use this feature in some capacity. The use will be more frequent near when the user signs up.
Preconditions	User decides to add friends
Basic Flow	1. User selects the add friends option from the home page 2. User clicks in search bar to search for friend by username 3. User selects option to add friend 4. System sends a friend request to selected user

Alternate Flow	1. User selects the add friends option from the home page 2. User selects the QR option 3. User scans other users QR code to add them as a friend 4. System sends a friend request to selected user
Postconditions	The app saves the account information of the added user. The app saves the connection between the two accounts and changes the privacy settings to allow certain features to be shared.

Table 6 Use Case Table

Use Case ID	004
Use Case Name	User accepts friend request
End Objective	To add friends on the app
User/Actor	App user
Trigger	The user selects the notifications option
Frequency of Use	Each user will use this feature in some capacity. The use will be more frequent near when the user signs up.
Preconditions	User received a friend request
Basic Flow	1. User opens app and selects the notifications option 2. User selects the friend request notification 3. User selects between the accept or reject options from the notification 4. If accepted, system sends confirmation notification to initial user that sent the request 5. System stores connections and updates customizations.
Alternate Flow	1. User opens app and selects notifications option 2. User selects the friend request notification

	3. User selects between the accept or reject options from the notification 4. If rejected, system deletes notification of friend request 5. System does not make connection
Postconditions	The system saves the friend information and updates privacy settings to share selected information. The system saves the connection as part of the user account.

Table 7 Use Case Table

Use Case ID	005
Use Case Name	User enables location services on app
End Objective	To utilize feature to find friends and bar locations
User/Actor	App user
Trigger	The user is prompted to enable or disable feature
Frequency of Use	If enabled, this feature will be used each time the app is opened
Preconditions	User receives a prompt to select option
Basic Flow	1. User accepts location services
	2. User access location services to see where they are and what bars their friends are at.
Alternate Flow	1. User does not accept location services and cannot use the application.
Postconditions	The system saves the location services preference per user.

Table 8 Use Case Table

Use Case ID	006
Use Case Name	User selects bar on map to get information
End Objective	To receive information on wait times and view the menu

User/Actor	App user
Trigger	The user selects the bar on the map
Frequency of Use	The frequency will vary based on user
Preconditions	User sees the bar as an option to select
Basic Flow	1. User opens application and navigates to the map function 2. User selects location they want more information about 3. System opens information page with menu and wait times 4. User closes page and returns to map
Alternate Flow	1. System promotes location to user based on location services 2. User selects bar from prompted message 3. System opens informational page 4. Users closes page and returns to map
Postconditions	The user closes the bar information and returns to the map.

Table 9 Use Case Table

Use Case ID	007
Use Case Name	User checks in at a bar
End Objective	The bar will have data on how many people check-in and the user's friends will be notified.
User/Actor	App user
Trigger	The user is promoted to check-in through location services.
Frequency of Use	Each time the user enters a bar in person.
Preconditions	User receives a prompt to select the option to check-in.
Basic Flow	1. User checks into the bar to see what friends are there. 2. Once the user checks into the bar, they will see what menu items are available.

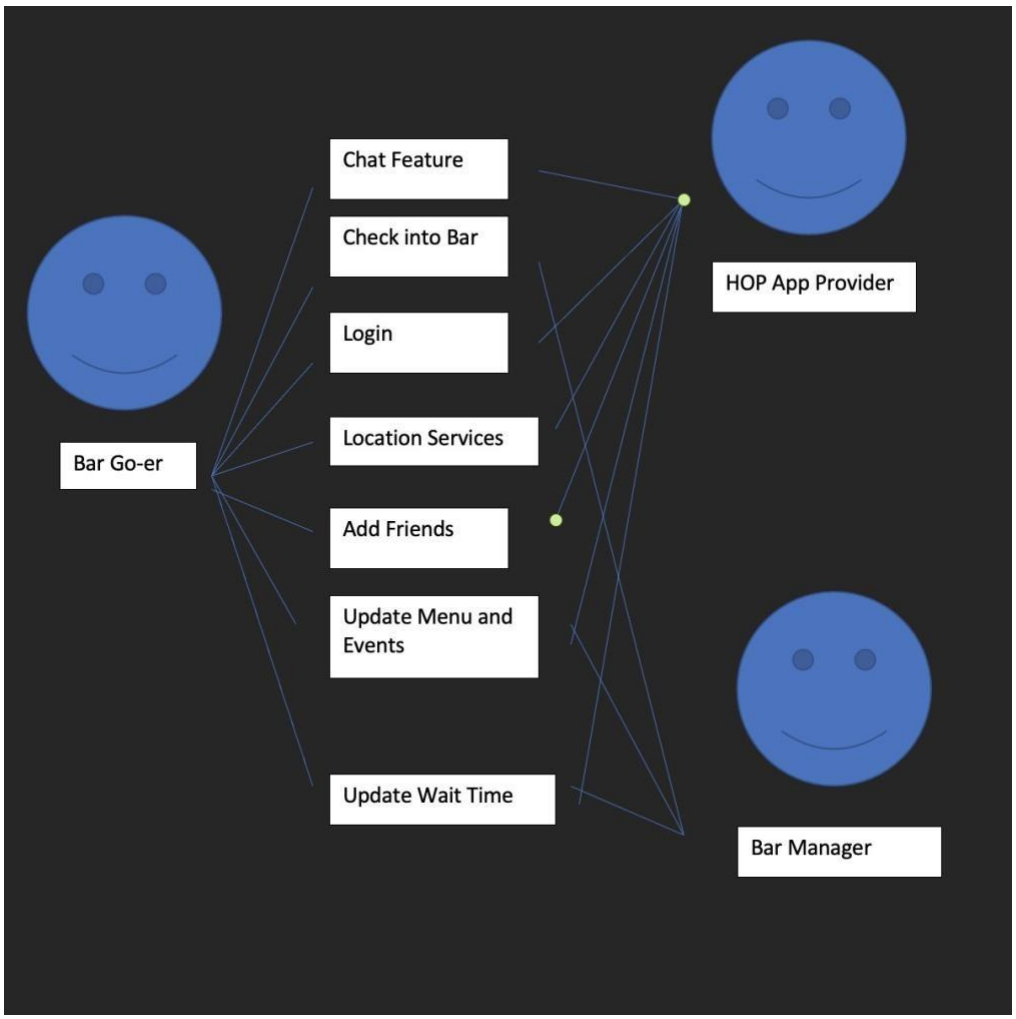
	3. Updates wait time along with any missing or changing menu items.
Alternate Flow	1. Prompt for user to check into the bar. 2. Prompt for user to update wait time and menu items. 3. Shows user what friends are at that particular bar.
Postconditions	The system saves the location services preference per user.

Table 10 Use Case Table

Use Case ID	008
Use Case Name	User chats with friends
End Objective	The user can talk with friends through a chat feature.
User/Actor	App user
Trigger	The user selects the chat option.
Frequency of Use	This will be used frequently based on the user.
Preconditions	User will receive a notification that there is a message or will open the chat function.
Basic Flow	1. User opens chat page to initiate a message 2. User selects recipient 3. Compositions of message and send by using the send icon.
Alternate Flow	1. Notification displays on device and user can be directed to the messages tab. 2. Composition or reply to message can follow.
Postconditions	The system will store the chat so the user can go back to it.

Use Case Diagram:

The use case diagram maps out the different features and capabilities that the application will provide the user.



Testing Plan:

Overview

The Testing section outlines the process of testing when creating the Hop application. This section outlines the testing methodology, the scope of testing, the objectives of testing, and the results and process logs from the testing process. There is also an outline of the team change management plan and how the team agreed upon making changes based on testing.

Methodology

The approach to testing was to test locally within Xcode and then share the code within Github so another team member can pull it down and test it locally as well. Putting the code into Github will allow all team members to access the code and work on the application.

Scope

Use Cases Tested:

- Single user sign-on
- Administrator (bar manager/owner) sign-on Features Tested:

- Sign-up/Sign-on
- Add Friend
- Location Services
- Messaging
- Check-in

Objectives

a. All major features and use cases need to be accounted for and incorporated.
b. All features must be working components for the user.
c. All bugs need to be resolved before the IT Expo.
d. All use cases are incorporated into the application. **Test Logs and Procedures**

Item #	Test Case #	Expected Output	Actual Output	Pass/Fail	Date
001	GitHub	Shared repository of xCode to share with group members	One group member as repository admin to push and pull lines of code for the group	Pass	10/05/2021
002	Welcome page	A functioning welcome page to sign up with email,	A fully functioning welcome page	Pass	09/22/2021
		Facebook or Google			

003	Location Services	Using location services to search for bars in users area	Location services that will be turned on to find the users location	Pass	10/18/2021
004	StoryBoard	Active storyboard to lay out the steps and processes of the application	Storyboard that portrays what features and processes taken to create the application	Pass	09/20/2021
005	Friends Location Feature	Using location services to find friends based in application	Error from code	Fail	02/02/2022
006	Menu Update	Updating menus as they change	Updates to menus that will change from bar to bar and week to week	Pass	02/02/2022
007	Location Services	Current user location	Current user location	Pass	02/07/2022

008	Friends Location Services	Update with an added friend's current location	Error from code	Fail	02/15/2022
009	Comet Chat	Ability to message friends and create chats	Users can chat with friends and groups	Pass	03/9/2022
010	Login page	User is able to login to the app	User can create account and login	Pass	3/18/2022

Testing Review

During the testing process, we were able to use the data collected to understand how to better improve our application. We were able to fix bugs or try different implementation methods to create the best feature possible. Each test led to an improvement of the feature that was being tested which led to the overall success of the application. Different features had to go through multiple testing interactions to understand why it wasn't working in the way the group wanted or how we could improve the code overall. We also completed periodic testing once we felt confident in a feature to ensure its continued success for the application. We learned that it would take multiple interactions and small changes to lead to the end result, and that a feature may work but it may need to be changed when implemented in the application as a whole. After completing the testing process, we would spend more time planning the features and how they may fit together in implementation to allow for an easier testing experience. We recognize that better planning may have led to less interactions before creating the end product.

Change Management Plan:

- We will need to streamline the look and functionality of the application. We will be working together to make sure we are set on how the application will look before we compile code into the application.
- Anyone can request a change. We have been working together on almost all decisions. All decisions are decided as a group and since our team has been accomplishing tasks with this ideology in place, it seems like the best route to continue with.

- We will triage problems together. Our group has been working together to work problems out together. We have been communicating in a group chat and met in person if the situation requires it.
- The approval is done as a democracy. We do not have any roles and we are all peers. We work together and bring up issues within the group and solve the issues as a team.
- For the most part we have been focusing on our project from a Xcode standpoint. Since most of us do not have any experience using Xcode, we have made helping each other figure out how to use the language a number one priority.
- We will update our weekly progress report to display changes we have made. We have also been updating our project plan within 48 hours of changing something within our plan or application.

Budget:

The budget for HOP is used to keep track of expenses and to ensure group members and stakeholders are on the same page financially. The cost of the application has been outlined in the project budget table. This includes the cost of the technology used, labor, and other miscellaneous costs. The group members were not paid for their work, but the estimated cost of their labor per hour has been considered.

Table 3 Project Budget

				Ongoing Annual		
	Rate Per/Hr	**Work Effort (Hours)**	**1 X Costs**	**Rate Per/Hr**	**Work Effort (Hours)**	**1 X Support Cost**
Labor - IT	20	600	$ 12,000	20	200	$ 4,000
Labor External	18	0	$ 0	18	0	$ 0
Software - External	$40/year	4 users	$ 0			$ 160
Hardware External	$1,400	4 computers	$ 5,600			$ 2,000
Misc.	$0	0	$ 0			
TOTAL			$ 17,600			$ 6,160

Problems Encountered and Analysis of Problems Solved:

We encountered problems that changed our approach and plan for the project as these issues changed timeframes and the initial framework of the application. The first problem was the lack of experience our group members had in xCode and iOS development. We were able to overcome this, but it did take a significant amount of time to learn how to use the software and language. We also had a difficult time pushing our code to GitHub and we were able to solve this by using SourceTree as a management software. Another problem was that some of the features we wanted to implement, mainly related to location services, were more difficult than expected. We used Google's API for finding a user's location, but we wanted to implement the capability of seeing another user's location on the same map. This was challenging and an ongoing problem as there aren't resources for this feature. This was a point of frustration for our group, but we focused on our other accomplishments. We worked together to tackle the problems we encountered and learned a new language and software collectively.

Conclusion

HOP is an application that can bring people together and our group was able to create this without prior development experience. We faced challenges in planning, the technical elements, and in the implementation process, but we made progress through our features and remembering our goals. The challenge of learning a new language and how to build out features aligned with our goals and were tasks that we achieved as a group. Each group member collectively navigated new software and brought abilities that contributed to the overall success of the project. We learned skills that we will be able to carry over into our careers and feel accomplished in completing the process of building an application. There is still work to do with the application and there are goals to build and expand upon features, but we made progress with a main framework and a functional interface. Some features are still in development which creates a goal for the group moving forward.

References

Academy, TechRepublic. “Apple Programming: Learn How to Create Apps for IOS 15, IPadOS, Apple Watch and Macos.” TechRepublic*, TechRepublic,* 19 Nov. 2021, https://www.techrepublic.com/article/apple-programming-learn-how-to-create-apps-for-ios15-ipados-apple-watch-and-macos/.

Atlassian. “Free Git Gui for Mac and Windows.” *SourceTree*, https://www.sourcetreeapp.com/.

Yelp Analytics Aayush Agrawal - Rutgers University. https://rucore.libraries.rutgers.edu/rutgers-lib/52036/PDF/1/play/.

Bielen, F. and Demoulin, N. (2007), "Waiting time influence on the satisfaction-loyalty relationship in services", *Managing Service Quality: An International Journal*, Vol. 17 No. 2, pp. 174-193. https://doi.org/10.1108/09604520710735182

Dharmawirya, Mathias and Oktadiana, Hera and Adi, Erwin, Analysis of Expected and Actual Waiting Time in Fast Food Restaurants (June 30, 2012). Industial Engineering Letters, Vol. 2, No. 5, 2012, https://ssrn.com/abstract=2184107

Inc., Apple. “Swift.” *Apple Developer*, https://developer.apple.com/swift/.

Melissa Baker, Suzanne K. Murrmann & Claudia Green (2013) Dining in the City: Server Behaviors, Time Preferences, and the Effect of Urbanization in Restaurants, Journal of Foodservice Business Research

Inc., Apple. *Swift.org*, https://www.swift.org/.

Wells, Garrett, "The Future of iOS Development: Evaluating the Swift Programming Language" (2015). *CMC Senior Theses*. 1179.
https://scholarship.claremont.edu/cmc_theses/1179