**Requirements Specification Document**

**Skip the Lines**

**Puzzled Software Solutions**

**October 22, 2019**

**<REDACTED>, <REDACTED>, <REDACTED>, <REDACTED>, James Ryan<REDACTED>, <REDACTED>, <REDACTED>**

**Contents**

[**Revision History**](#_8gw66q1lm3zj) **3**

[**1 Introduction**](#_1tk3zdu8w41m) **6**

[1.1 Purpose](#_lh2ppmf6cece) 6

[1.2 Project Scope](#_td5und95xcpo) 6

[1.3 Glossary of Terms](#_tgbl38x7pp7a) 6

[1.4 References](#_vf8rp37kwn4r) 7

[1.5 Overview](#_z15yaxrv7y00) 8

[**2 Overall Description**](#_whtjxbfww7vp) **8**

[2.1 Product Perspective](#_25i5auhspw2p) 8

[2.2 Product Features](#_31u2p772avlj) 8

[2.3 User Classes and Characteristics](#_c4ikdtn16hvj) 8

[2.4 Operating Environment](#_ebuc9ptnfbwy) 9

[2.5 Design and Implementation Constraints](#_yim9ornezjzd) 9

[2.6 Assumptions and Dependencies](#_xpmyciloqwwr) 10

[**3 System Features**](#_ht83se1r3pz6) **10**

[3.1 Customer Interface](#_ouipwm5bdmda) 11

[3.1.1 Description and Priority](#_x9pv86sbwkp9) 11

[3.1.2 Functional Requirement](#_pjiw4sa6l3ga) 11

[3.2 Reservation Booking System](#_f0lz2s6n64uf) 13

[3.2.1 Description and Priority](#_cn58ohoonyi4) 13

[3.2.2 Functional Requirements](#_a6jup3ofhnwh) 13

[3.3 Restaurant Employee Interface](#_fbzrdtwi9jyg) 16

[3.3.1 Description and Priority](#_a8t10if5kf6a) 16

[3.3.2 Functional Requirements](#_1nzeveycojih) 16

[3.4 Restaurant Owner Interface](#_3ti2hdia6149) 18

[3.4.1 Description and Priority](#_a099sxten7zr) 18

[3.4.2 Functional Requirements](#_kytfgoflvft7) 18

[**4 External Interface Requirements**](#_ezyxol2stib4) **20**

[4.1 User Interfaces](#_qqa8678kc4v7) 20

[4.2 Software Interfaces](#_gx4nd1w5idps) 21

[4.2.1 Koogle Servers & Databases](#_obckpwn2dbki) 21

[4.2.2 Customer Devices (Android or iOS or Supported Web Browser)](#_ta78eattc1zj) 21

[4.2.3 Restaurant’s Tablet](#_sf7nollzsgb4) 22

[4.3 Communications Interfaces](#_8c75gjgrtq9r) 22

[**5 Other Non-Functional Requirements**](#_2vvhsroqpee) **22**

[5.1 Accuracy Requirements](#_2zs435hf11xu) 22

[5.2 Security Requirements](#_pf9writvxpf4) 22

[5.3 Software Quality Attributes](#_4mv0qoiiqcun) 23

[**6 Diagrams**](#_lj03bfle9fdh) **23**

[6.1 Entity Relationship Diagram](#_55xuw3j1qc62) 23

[6.2 Context Diagram](#_71dssrcmm58u) 25

[6.3 Data Flow Diagram Level 1](#_rgw50u2wbnvk) 26

[6.4 Data Flow Diagram Level 2](#_u3gl2kffpeon) 27

[**Appendix: Issues List**](#_tx7ctnc0tndr) **27**

# 

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for Changes** | **Version** |
| Puzzled Software Solutions | Sept. 25 2019 | Initial draft creation | RD 0.1 |
| Puzzled Software Solutions | Sept. 26 2019 | Continual Initial Draft | RD 0.2 |
| Puzzled Software Solutions | Sept. 28 2019 | Continual Initial Draft | RD 0.3 |
| **<REDACTED>** | Sept. 30 2019 | Added functional requirements detailing the cancellation of reservations held past the hold time. 3.1-17, 3.1-18 | RD 0.3 |
| **<REDACTED>** | Sept. 30 2019 | Edited and arranged contents headings and page numbers.  Worked on 1.2, 2.3, 4.1, 6, Appendix | RD 0.4 |
| **<REDACTED>** | Sept. 30 2019 | Worked on 1.3, 2.5, 2.6, 4.2, 5.1, 6, Appendix | RD 0.4 |
| James | Sept. 30 2019 | Work on 4.2.1, 5.3 and document formatting. | RD 0.4 |
| Puzzled Software  Solutions | Sept. 30 2019 | RD Version 1.0 | RD 1.0 |
| James | Oct. 5 2019 | Initial document creation from RD 1.0 | RSD 0.0 |
| James | Oct. 7 2019 | Worked on modifications based on TA feedback and client RD 1.1 | RSD 0.1 |
| Jordan | Oct. 7 2019 | Made modifications based client feedback for RD1.1 including sections: Glossary of Terms, 1.0, 2.1, 2.2, 2.5, 2.6, 3, 3.1.2, 4.3  Listed potential Use Cases (at the end of the document) and created an example of one. | RSD 0.2 |
| Jordan | Oct. 8 2019 | Added UX/UI and Koogle Account to the Glossary of Terms | RSD 0.3 |
| Puzzles Software Solutions | Oct. 8/9 2019 | Work on use cases. Continue improvements to document based on RD 1.0 feedback. | RSD 0.4 |
| **<REDACTED>** | Oct. 10 2019 | Rearranged sections 3.1, 3.2, 3.3. Added use case templates in sections 3.1, 3.2, 3.3 | RSD 0.6 |
| **<REDACTED>** | Oct. 14 2019 | Worked on use cases 2, 2.1, 3, 4 | RSD 0.7 |
| **<REDACTED>** | Oct. 15 2019 | Revised sections 2.2, 2.3. Worked on uc-1, uc-2 | RSD 0.8 |
| James | Oct. 15 2019 | Revise uc-3, 1.5, REQ-3.2-14, uc-3.1 main flow | RSD 0.9.0 |
| **<REDACTED>** | Oct. 15 2019 | Added REQ 3.3-8 - 3.3-11 and uc7-10  Made the Use Case Diagram | RSD 0.9.1 |
| **<REDACTED>** & James | Oct. 15 2019 | Revised: References, 1.5, 2.6, table formatting, UC-7, REQ-3.2-13 | RSD 0.9.2 |
| Puzzles Software Solutions | Oct. 15 2019 | Proofreading, editing, finalizing use cases. | RSD 1.0 |
| **<REDACTED>** | Oct. 21 2019 | Made a DFD level 0 and DFD level 1 | RSD 1.0.1 |
| James | Oct. 22 2019 | TA feedback up 2.6 inclusive. | RSD 1.0.2 |
| **<REDACTED>** | Oct. 22 2019 | Modified the use case diagram and included the same titles as the use cases. Went through to remove as many plurals as I could find. | RSD 1.0.3 |
| Puzzles Software Solutions | Oct. 22 2019 | Corrections based on TA feedback. | RSD 1.0.4 |
| **<REDACTED>** | Oct. 23 2019 | Added the additional use cases | RSD 1.1.0 |
| Puzzles Software Solutions | Oct. 23 2019 | Use cases corrections, and wait queue functionality removed | RSD  1.1.1 |
| **<REDACTED>** | Oct. 24 2019 | Revised use cases 3, 5, 6. Added use case “Update table status” in section 3.3 | RSD 1.1.2 |
| **<REDACTED>** | Oct. 25 2019 | Read the RSD and commented on areas that need improving | RSD 1.1.3 |
| **<REDACTED>**, James, **<REDACTED>**, **<REDACTED>** | Oct. 25 2019 | Use case rewrites, additions and removals. | RSD 1.1.4 |
| **<REDACTED>** | Oct. 25 2019 | Reviewed paper and made comments on areas that need improving | RSD 1.1.5 |
| **<REDACTED>** | Oct. 27 2019 | Overall read through the document, added comments and fixed punctuation. Added to sections 5.2, 5.3 | RSD 1.1.6 |
| **<REDACTED>** | Oct. 27 2019 | Changed UC-7 and reordered functional requirements. | RSD 1.1.7 |
| **<REDACTED>** | Oct. 29 2019 | Updated the use case diagram | RSD 1.1.8 |
| **<REDACTED>** | Oct. 29 2019 | Updated unfinished use case, review documentation, added diagram description, draft dfd level 2 | RSD  1.1.9 |
| Puzzles Software Solutions | Oct. 29 2019 | Data Dictionary, Descriptions, Diagrams | RSD 2.0 |

# 

# 1 Introduction

## 1.1 Purpose

Skip the Lines is an application that will provide a customer with the ability to make a reservation at a restaurant, as well as provide an estimation of wait times for the nearby restaurants. The consequent aim of Skip the Lines is to reduce the size of lineups among all restaurants, cut a customer’s wait-time in line at a restaurant, and assist restaurant owners to better allocate resources, such as food supply. This Requirements Specification Document (RSD) describes the intended purpose, requirements and nature of the software that is to be developed, intending to satisfy the required effects.

## 1.2 Project Scope

The features of Skip the Lines are to be implemented as an extension of Koogle Maps, one of Koogle’s existing services. Skip the Lines’ main feature will let each Koogle user make a table reservation at a restaurant listed on Koogle Maps. Skip the Lines features are expected to reduce wait times among affected restaurants, provide high accuracy wait-time estimations, reduce the frequency and size of lineups outside restaurants and reduce the amount of time restaurants must spend to coordinate reservations.

## 1.3 Glossary of Terms

|  |  |
| --- | --- |
| **Customer** | A user that is looking for a restaurant to dine at. |
| **Device Location Detection** | The mechanisms for which a device uses to detect its own geographical location. |
| **End Time** | The time that a group’s session with a table ends. |
| **Group** | A collection of one or more customers that eat together at a table. |
| **Koogle Account** | Provides Koogle-wide access to Koogle products such as Kmail, using the same username and password. |
| **Koogle Maps** | A web mapping service that offers satellite imagery, aerial photography, street maps, 360° panoramic views of streets, real-time traffic conditions, and route planning for traveling by foot, car, bicycle and air, or public transportation. |
| **Open Table** | An online reservation service that allows a user to find and book a table at a desired restaurant. |
| **Patron Count** | The number of patrons currently in a restaurant. |
| **PII** | Personal Identifiable Information is any data that could potentially identify a specific individual. |
| **Reservation** | A request made by a customer to hold a table for a group where the group’s size and the start time are both specified by the customer. |
| **Reservation Hold Time** | The amount of time a reservation is held after the reservation's start time before it is cancelled. This will be specified by each restaurant. |
| **Restaurant** | An establishment where people go to eat meals that are cooked and served on the premises. |
| **Restaurant Details** | List of information relating to a restaurant including: hours of operation, total restaurant seat count, menu items, reservation hold time, address, contact information and restaurant genre. |
| **Restaurant Analytics** | Information available to restaurant owners which pertains to restaurant performance in Skip the Lines. This information includes data such as reservations per hour, peak reservation times, and reservation history. |
| **Start Time** | The time that a group’s session with a table begins. |
| **Table** | A table in a restaurant that can hold one group of customers at a time. |
| **UI** | User Interface involves the look and feel, or the presentation and interactivity of the system. |
| **UX** | User Experience is the process of enriching customer satisfaction by improving the ease of use provided in the interaction between the customer and the system. |
| **Wait Time** | The amount of time a group has to wait before receiving a table at a given restaurant. |

## 1.4 References

“Koogle's Website”, Home. [Online]. Available: https://skipthelines.vernonliu.com/. [Accessed: 26-Sep-2019].

“OpenTable”, Find your table for any occasion. [Online]. Available: https://www.opentable.ca [Accessed: 28-Sep-2019].

“RFP”, RFP. [Online]. Available: https://skipthelines.vernonliu.com/rfp.html/. [Accessed: 15-Oct-2019].

“RD 1.1”, RD 1.1. [Online]. Available: https://skipthelines.vernonliu.com/rd.html. [Accessed: 15-Oct-2019].

## 1.5 Overview

RSD 2.0 aims to provide clarification on the aspects of Skip the Lines, such as UI/UX interface features, functional, and non-functional requirements. Included in the System Features section are use case diagrams for each feature. The RSD will outline what has been proposed by the client in consideration of the Request for Proposal (RFP) and Requirements Document (RD) 1.1 documents on Koogle’s website and various client meetings. This document contains an overall description, list of system features, external interface requirements and other non-functional requirements.

# 2 Overall Description

## 2.1 Product Perspective

Currently, Koogle doesn’t have a system for reserving a table at a restaurant. Although resources exist for measuring restaurant quality and reservation capabilities, there is not a service that provides real-time estimations of wait times. Koogle wants Skip the Lines to provide such real-time estimates, along with an integrated table reservation request system. The table reservation request system will hold tables for a customer and notify that customer when a table becomes available. There is an application called OpenTable which allows a customer to make a reservation at a restaurant. Skip the Lines will implement the functionality of OpenTable with additional features such as: integration into Koogle Maps, real-time wait estimates, and notifications for when a restaurant has a table available. Skip the Lines will be a new system integrated into Koogle Maps.

## 2.2 Product Features

The main feature of Skip the Lines will allow a customer to make a table reservation at a nearby restaurant. Additionally, when a customer selects a restaurant in Koogle Maps, Skip the Lines will display that restaurant’s associated wait time estimation.

## 2.3 User Classes and Characteristics

The four user classes in order of their importance to be satisfied by Skip the Lines are: customer, restaurant employee, restaurant owner, and administrator.

* *Customer*

A customer is any person with a phone, a tablet or a computer and an internet connection who has a Koogle account. Through the customer interface, a customer can view each nearby restaurant, view each restaurant's respective wait time estimate, and make a reservation at a restaurant. A customer is expected to have the lowest level of technical expertise and are the most important class to satisfy.

* *Restaurant Employee*

The restaurant employee class is representative of an employee of a dining establishment that is registered with Skip the Lines. This user class will use an interface to perform tasks such as: viewing reservations, adding a customer to the restaurants reservation list, viewing food that is pre-ordered for a reservation and cancelling reservations. A restaurant employee is expected to have a moderate level of technical expertise.

* *Restaurant Owner*

The restaurant owner class is representative of an owner of a dining establishment that is registered or wishes to be registered with Skip the Lines. This user class will use an interface to perform tasks specific to restaurant owners such as: create a restaurant Skip the Lines user account, delete the restaurant’s Skip the Lines user account, and view restaurant analytics. A restaurant owner is expected to have a moderate level of technical expertise.

* *Administrator*

This user class represents a Skip the Lines employee who has full access to Koogle’s database and the administrative menus of Skip the Lines. This user has the highest level of technical expertise and is a representative of Koogle.

## 2.4 Operating Environment

Skip the Lines will run on mobile phones through the Koogle Maps mobile phone application, and web browsers through the Koogle Maps website for customers and restaurants. Administrators will have a dedicated web interface for managing Skip the Lines. The web interface for administrators is not part of the scope of this document.

Skip the Lines will operate on the following browsers: Internet Explorer 11+, Edge Browser 44.18362.267.0+, Chrome 77.0.3865.90+ , Opera 2019+ and Firefox 69.0.1+.

Skip the Lines will operate through the Koogle Maps mobile phone application on the following mobile operating systems: iOS 11+ and Android 7+.

## 2.5 Design and Implementation Constraints

The following design and implementation constraints highlight the issues that will limit the options available:

* All PII must be securely stored in Koogle’s encrypted MySQL 8.0 database.
* Skip the Lines must be maintained by Puzzled Software Solutions.
* Skip the Lines must be hosted on Koogle’s Ubuntu Server 18.04.3 LTS servers and MySQL 8.0 database systems.

## 2.6 Assumptions and Dependencies

In order for Skip the Lines to excel, the following assumptions about the environment are made:

* Each Skip the Lines user has a smart device or computer with a mobile browser.
* Each Skip the Lines user must have an internet connection.
* Koogle servers and databases will host Skip the Lines system, details outlined in section 2.5 Design and Implementation Constraints.
* The Puzzled Software Solutions team will have access to servers for Koogle Maps and Koogle Account in order to test, deliver, and maintain the Skip the Lines software.
* Skip the Lines requires a registered restaurant to have access to the internet.

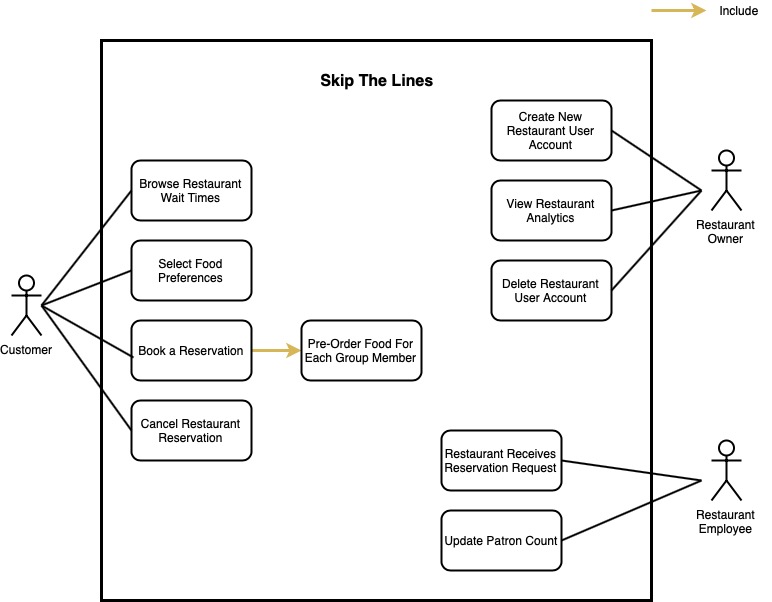
# 3 System Features

With respect to the four user classes, Skip the Lines will have the following features:

* Customer
  + Check wait times for a restaurant.
  + Reserve a table at a restaurant.
  + Pre-order food while creating a reservation (Requires customer payment information to be on file).
* Restaurant Employee
  + View the restaurant’s reservation list.
  + For each customer in their respective reservation booking, a restaurant can view that customer's pre-ordered food.
  + A restaurant employee will be able to update their restaurant’s patron count as customers come and go.
* Restaurant Owner
  + Create a restaurant account on Skip the Lines for their restaurant.
  + Remove their restaurant’s Skip the Lines account.
  + View analytical information
* Administrator
  + This document does not cover the scope for administrator use of Skip the Lines.

**Use Case Diagram:**

The following use case diagram shows how all the use cases and actors interact with each other.



## 3.1 Customer Interface

The Customer Interface system allows a customer to access Skip the Lines’ features.

### 3.1.1 Description and Priority

The Customer Interface is a high priority feature that allows a customer to access Skip the Lines. A customer must be able to access each restaurant’s details. A customer with a Koogle account must be able to access the reservation service and receive related notifications.

### 3.1.2 Functional Requirement

REQ-3.1-1: Each customer must be able to set their food style preferences.

REQ-3.1-2: Each customer must be able to view the corresponding wait time estimate of each restaurant that customer selects in Koogle Maps.

REQ-3.1-3: Each customer must be able to select the restaurant of their preference from Koogle maps.

REQ-3.1-3: Each customer must be able to make a reservation at each restaurant selected in Koogle Maps.

|  |
| --- |
| **Use Case: Browse Restaurant Wait Times** |
| **ID:** UC-1 |
| **Brief description:** A customer views a restaurant’s wait time estimates. |
| **Actor(s):** Customer |
| **Preconditions:**   1. The customer must have access to Koogle Maps. 2. The selected restaurant is registered with Skip the Lines. |
| **Main flow:**   1. Through Koogle Maps, the customer selects the Skip the Lines link for a specific restaurant. 2. The wait time estimate for the restaurant is displayed to the customer. |
| **Postconditions:**   1. The customer knows the wait time estimate for the restaurant. |
| **Alternative flow(s):** None |

|  |
| --- |
| **Use Case: Select Food Preferences** |
| **ID:** UC-2 |
| **Brief description:** A customer sets their food preferences for their Koogle account. |
| **Actor(s):** Customer |
| **Preconditions:**   1. The customer must have a Koogle account. |
| **Main flow:**   1. The customer logs in to his or her Koogle account. 2. The customer is displayed a list of food types. 3. The customer selects all food types that they prefer. 4. The customer’s food preferences are saved to their Koogle account. |
| **Postconditions:**   1. The food preferences saved in the customer’s Koogle account reflect the customer’s desired food preferences. |
| **Alternative flow(s):** None. |

## 

## 3.2 Reservation Booking System

This feature facilitates the ability for a customer to create a reservation at a restaurant registered with Skip the Lines.

### 3.2.1 Description and Priority

The reservation booking system is a high priority feature that operates an online reservation service through Koogle Maps. The reservation booking system also allows a customer to pre-order food for their booked reservation.

### 3.2.2 Functional Requirements

REQ-3.2-1: Each customer must have a Koogle account in order to make a reservation.

REQ-3.2-2: Each customer must be able to make a reservation in Skip the Lines from Koogle Maps.

REQ-3.2-3: Each customer can only make a reservation for the same day they will book the reservation.

REQ-3.2-4: Each customer must not be allowed to create more than one reservation on the same day with a start time that is within 90 minutes of each other.

REQ-3.2-5: Each customer must be able to enter their name, phone number, and group size while making a reservation.

REQ-3.2-6: Each customer must be able to pre-order food for each member of his or her group.

REQ-3.2-7: Each customer must have a payment method registered with their Koogle account to be able to pre-order food with their reservation.

REQ-3.2-8: Each customer must agree to the Terms of Service provided by Koogle in order to make a reservation.

REQ-3.2-9: Each customer must agree to the Privacy Policy provided by Koogle in order to make a reservation.

REQ-3.2-10: Each customer must be shown the respective reservation hold time for a reservation before the reservation is submitted.

REQ-3.2-11: When a customer’s reservation request is confirmed by the respective restaurant, a reservation confirmation notification must be sent to the phone number associated with the customer’s Koogle account.

|  |
| --- |
| **Use Case: Book a Reservation** |
| **ID:** UC-3 |
| **Brief description:**  A customer can reserve a table at a restaurant for him or herself or for a group of people including themselves with their Koogle accounts. |
| **Actor(s):**  Customer |
| **Preconditions:**   1. The customer must have a Koogle account and access to Koogle Maps. 2. The customer can only book a reservation that takes place on the same day they make the reservation. |
| **Main flow:**   1. The customer logs in to his or her Koogle account. 2. Through Koogle Maps, the customer selects the Skip the Lines reservation link for a specific restaurant. 3. The customer enters the start time and the size of their group. 4. The customer submits the reservation. |
| **Postconditions:**   1. The customer booked a reservation for a table at a restaurant. 2. The customer received a confirmation notification through text and email that the reservation booking was successful, including:    1. The Restaurant that the customer selected.    2. The booked reservation time.    3. If the customer pre-ordered a meal:       1. The group’s meal details. 3. The customer will receive a notification at the start time of their reservation. |
| **Alternative flow(s):**   1. Include: Pre-order food   1.1. See Main flow 3-6   1. A customer decides to cancel the reservation before the reservation has been made. 2. A customer's reservation is automatically cancelled if he or she does not claim their table before the end of the reservations hold time |

|  |
| --- |
| **Use Case: Pre-Order Food For Each Group Member** |
| **ID:** UC-4 |
| **Brief description:**  A customer pre-orders food during their reservation creation process. Food shall be ready at the time of the reservation. |
| **Actor(s):**  Customer |
| **Preconditions:**   1. The customer must be in the reservation creation process. 2. The customer must have chosen to preorder food during the reservation creation process. 3. The customer must have specified the number of people in their group during the reservation creation process. 4. The customer’s payment information must be on their Koogle account. 5. The pre-ordered item must be in the restaurant’s menu. |
| **Main Flow:**   1. For each group member of the reservation the customer enters the following information:    1. Group member name    2. Menu item(s) to be pre-ordered    3. Modifications to pre-ordered item(s)    4. Allergies or other dietary restrictions 2. The customer reviews the pre-ordered item(s) 3. The customer moves onto the next step in the reservation creation process. |
| **Postconditions:**   1. Once the reservation is submitted, the pre-order is sent to the restaurant. |
| **Alternate Flow(s):**  None |

|  |
| --- |
| **Use Case: Cancel Restaurant Reservation** |
| **ID:** UC-5 |
| **Brief description:**  After a customer has booked a reservation at a restaurant, he or she no longer wants that reservation at the selected restaurant. |
| **Actor(s):**  Customer |
| **Preconditions:**   1. The reservation must already be booked. 2. The reservation start time to be seated at the restaurant has not gone by. |
| **Main flow:**   1. A customer selects his or her booked reservation through Koogle Maps. 2. A customer clicks on “My Bookings” button on his or her dashboard. 3. A customer clicks on the “Cancel” button next to his or her booking confirmation number that he or she wishes to cancel. 4. A customer is prompted to confirm cancellation.    1. Either the customer cancels the cancellation,    2. Or the customer confirms the cancellation. 5. The customer cancels the reservation. |
| **Postconditions:**   1. The reservation at the selected restaurant is removed. 2. The customer receives a cancellation confirmation on his or her respective email. |
| **Alternate Flow(s):**  None. |

## 3.3 Restaurant Employee Interface

Restaurant Registration facilitates the ability for a restaurant to create an account with Skip the Lines which then allows the restaurant to start taking reservations from customers through Skip the Lines. The restaurant employee interface will

### 3.3.1 Description and Priority

Restaurant Registration is high priority. It facilitates the ability for a restaurant to register with Skip the Lines which allows a restaurant employee to view the restaurants reservation list, view pre-ordered food if applicable, update the restaurants table availability and allows the restaurant to take reservations through Skip the Lines.

### 3.3.2 Functional Requirements

REQ-3.3-1: Each restaurant must be able to enter their primary contact name, primary contact phone number, restaurant name, restaurant address, mailing address, operating hours, restaurant type, reservation hold time, menu, and reservation type at the time of registration.

REQ-3.3-2: Each restaurant employee must be able to view each customer’s reservation who has made a reservation for their restaurant.

REQ-3.3-3: Each restaurant must be able to view the pre-ordered food options for a customer’s reservation.

REQ-3.3-4: A restaurant employee must be able to update their restaurant’s current patron count.

|  |
| --- |
| **Use Case: Restaurant Receives Reservation Request** |
| **ID:** UC-6 |
| **Brief description:** A restaurant employee receives a reservation request notification. |
| **Actor(s):** Restaurant employee, customer |
| **Preconditions:**   1. The employee’s restaurant is registered with Skip the Lines. 2. A customer made a reservation request at the employee’s restaurant through Skip the Lines. |
| **Main flow:**   1. A restaurant employee is displayed a reservation request notification through the restaurant employee interface. 2. If the reservation request includes pre-ordered meal details:    1. The restaurant employee is displayed the meal details. 3. The restaurant employee interface prompts the restaurant employee to confirm the reservation request. 4. If the restaurant employee confirms the reservation request:    1. The customer that made the reservation request is sent a text message confirming their reservation. |
| **Postconditions:**   1. The request of making a reservation at the selected restaurant is confirmed. 2. The customer receives a confirmation notification on his or her respective email. |
| **Alternative flow(s):** None. |

|  |
| --- |
| **Use Case: Update Patron Count** |
| **ID:** UC-7 |
| **Brief description:** When a group sits down at a table in a restaurant, a restaurant employee will update the patron count. In turn, the restaurant employee will update the availability of seats when a group is finished dining at the restaurant. |
| **Actor(s):** Restaurant employee |
| **Preconditions:**   1. A group enters or leaves the restaurant. |
| **Main flow:**   1. The restaurant employee manually determines the current patron count. 2. The restaurant employee updates the current patron count for their restaurant’s Skip the Lines account through the restaurant employee interface. |
| **Postconditions:**   1. The restaurant’s patron count is updated. |
| **Alternative flow(s):** None. |

## 

## 3.4 Restaurant Owner Interface

The Restaurant Owner Interface facilitates the ability for a restaurant owner to create a new account with Skip the Lines which then allows the restaurant to start taking reservations from customers through Skip the Lines. Additionally, the restaurant owner can view restaurant analytics and has the ability to delete his or her restaurant user account.

### 3.4.1 Description and Priority

The Restaurant Owner Interface is high priority. It facilitates the ability for a restaurant to register with Skip the Lines which enables the restaurant to have access to Skip the Lines services. The Restaurant Owner Interface will allow the restaurant owner to view details on the restaurant analytics to analyse and investigate. If the restaurant owner no longer wishes to obtain Skip the Lines services, the Restaurant Owner Interface provides the option to delete his or her account.

### 3.4.2 Functional Requirements

REQ-3.4-1: Each restaurant owner must be able to create an account with Skip the Lines.

REQ-3.4-2: Each restaurant owner must enter their restaurant account information at the time of account creation.

REQ-3.4-3: Each restaurant owner must be able to delete his or her registered restaurant account from Skip the Lines.

REQ-3.4-4: Each restaurant owner must be able to edit their restaurant details account information when logged in.

|  |
| --- |
| **Use Case: Create New Restaurant User Account** |
| **ID:** UC-8 |
| **Brief description:** A restaurant owner creates a new and unique Skip the Lines account for his or her restaurant. |
| **Actor(s):** Restaurant owner |
| **Preconditions:**   1. The restaurant owner owns a restaurant that does not have an associated Skip the Lines account. |
| **Main flow:**   1. The restaurant owner wants to create a new Skip the Lines account. 2. The restaurant owner enters his or her name, email, date of birth, restaurant address, city, and up to 5 additional contact names and contact phone numbers. 3. While the restaurant owners details are invalid.    1. The restaurant owner is prompted to enter their details again for confirmation.    2. The restaurant owner’s details are validated. 4. The restaurant owner creates a new Skip the Lines account for his or her restaurant. |
| **Postconditions:**   1. A new account has been created for the owner’s restaurant. 2. A confirmation email was sent to the restaurant owner verifying the new restaurant user account was created. |
| **Alternative flow(s):**  2.1 a) The restaurant owner is prompted with an invalid email address exception  2.1 b) The restaurant owner is prompted with an invalid password exception   1. Account creation is cancelled. |

|  |
| --- |
| **Use Case: View Restaurant Analytics** |
| **ID:** UC-9 |
| **Brief description:** The restaurant owner views restaurant analytics for their restaurant. |
| **Actor(s):** Restaurant owner |
| **Preconditions:**   1. The restaurant owner has registered their restaurant with Skip the Lines. 2. The restaurant owner is logged into their restaurant account. |
| **Main flow:**   1. The restaurant owner selects the analytics view of the restaurant interface. 2. If there are no analytics to view:    1. Go to alternate flow 1. 3. The restaurant owner views the restaurant analytics. |
| **Postconditions:**   1. The restaurant owner has viewed the restaurant analytics. |
| **Alternative flow(s):**   1. The restaurant has no analytics for the restaurant owner to view. |

## 

|  |
| --- |
| **Use Case: Delete Restaurant User Account** |
| **ID:** UC-10 |
| **Brief description:** The restaurant owner wants to remove their restaurant’s Skip the Lines account. |
| **Actor(s):** Restaurant owner |
| **Preconditions:**   1. The restaurant owner logs in to their restaurant’s Skip the Lines account. |
| **Main flow:**   1. The restaurant owner selects the option to “delete account”. 2. A confirmation for deletion is prompted to the restaurant owner 3. If the restaurant owner does not want to delete account:    1. Account deletion is cancelled. 4. The restaurant owner’s account is removed. |
| **Postconditions:**   1. The restaurant’s Skip the Lines account has been removed. |
| **Alternative flow(s):** None. |

## 

# 4 External Interface Requirements

## 4.1 User Interfaces

The customer interface is integrated into Koogle Maps. Below, Figure 1 shows an example of what Koogle Maps looks like when a customer searches for a restaurant(s). Koogle Maps will contain the UI specific elements for Skip the Lines.

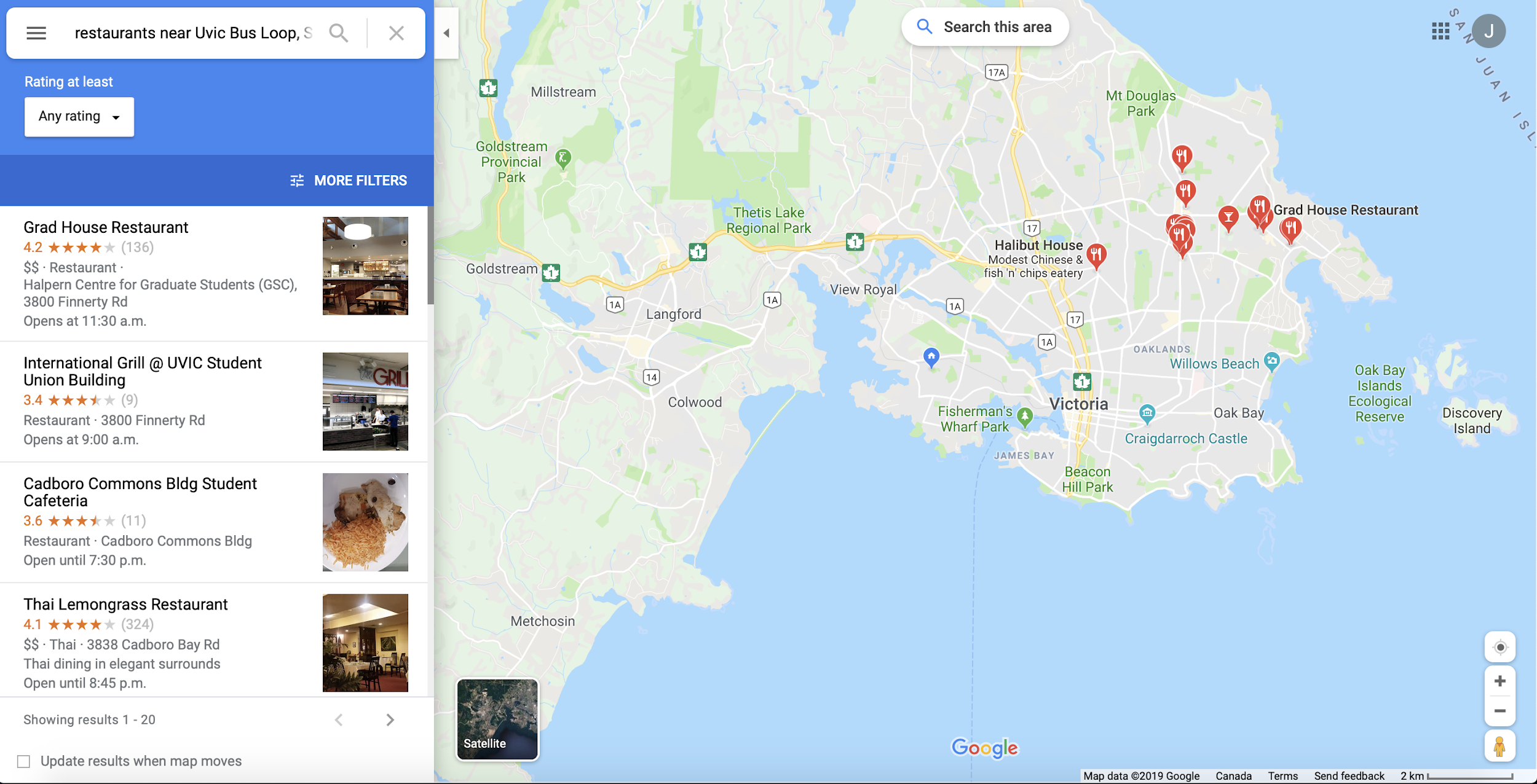


Figure 1. An example of a search for a restaurant in Koogle Maps.

## 4.2 Software Interfaces

This section describes the interactions between Skip the Lines and external software systems.

### 4.2.1 Koogle Servers & Databases

Data in from restaurants and customers to Koogle servers and databases:

* Information requests from users such as: wait times, available reservations, restaurant information.
* Pull requests from a restaurant, such as: requests to be taken off a reservation list.
* Reservation requests from a customer.
* Restaurant information for restaurants registering with Skip the Lines that are not already in Koogle’s databases.

Data out from Koogle servers and databases to restaurants and customers:

* Reservation information & accompanying relevant customer information to restaurant such as: when a reservation is created or edited.
* Display of information on Koogle Maps (Integration into Koogle Maps)
* Notifications such as: specials, and alerts to when your table is ready or reservation is nearing)
* Up to date information about restaurants such as: address, hours of operation and menu.

### 4.2.2 Customer Devices (Android or iOS or Supported Web Browser)

Data in from Koogle servers and databases:

* Up to date restaurant information such as: wait times, available reservations, restaurant information.
* Reservation creation/altering confirmation.
* Notification SMS messages such as: specials, and alerts to when your table is ready or reservation is nearing.
* Notification email message to confirm a customer’s reservation booking.

Data out to Koogle servers and databases:

* Phone Number, Email, Name upon account creation.
* Reservation information such as: creating or altering of said reservation.

### 4.2.3 Restaurant’s Tablet

Data in from Koogle servers and databases:

* Reservation information & accompanying relevant customer information to restaurant such as: when a reservation is created or edited.

Data out to Koogle servers and databases:

* Up to date information such as: wait times, available reservations, and restaurant information.

## 4.3 Communications Interfaces

Skip the Lines offers a communication interface for automatically sending notifications regarding on the wait time and table availability. Each registered restaurant provides their real-time wait time information into Skip the Lines. A notification related to the current wait time will be sent to a customer’s Koogle account after this customer makes a reservation. Skip the Lines does not support direct communication interfaces for a restaurant to contact a customer. However, a customer must enter his or her contact information while booking a reservation. The restaurant will then receive the customer’s contact information and can contact the customer if necessary. Skip the Lines must allow a customer user to select if they want to be contacted by the restaurant via email, a phone call, or SMS while creating a reservation.

# 5 Other Non-Functional Requirements

## 5.1 Accuracy Requirements

Projected wait times listed for a restaurant on Skip the Lines must be within 10 minutes (+/-) of actual wait times.

## 5.2 Security Requirements

REQ-5.2-1: A customer must not be able to view any other customer’s personal information.

REQ-5.2-2: A customer must not be able to view any other customer’s reservation information.

REQ-5.2-3: A customer's geographic location must not be accessed without the customer's permission.

REQ-5.2-4: User information must be encrypted and protected using a cryptosystem.

REQ-5.2-5: User financial information must be encrypted and protected using a cryptosystem.

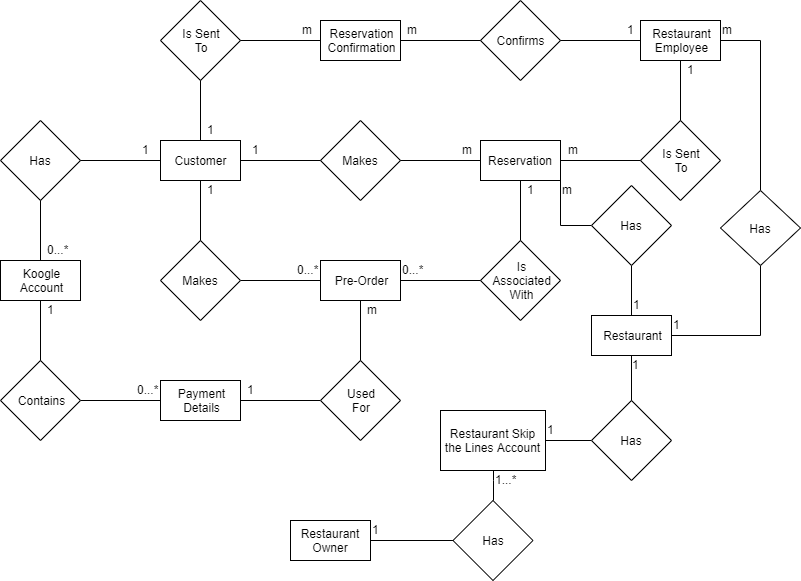
## 5.3 Software Quality Attributes

Skip the Lines must provide a well designed UI, and UX for customers and restaurants. UX is not a quantitative characteristic that can be measured, but rather a qualitative characteristic that will be improved over time based on feedback. A user (customer or restaurant employee) should be able to use the application with basic technology skills. It is expected that within 2 or 3 times of using the Skip the Lines app, the user will be considered proficient on the application.

# 6 Diagrams

## 6.1 Entity Relationship Diagram

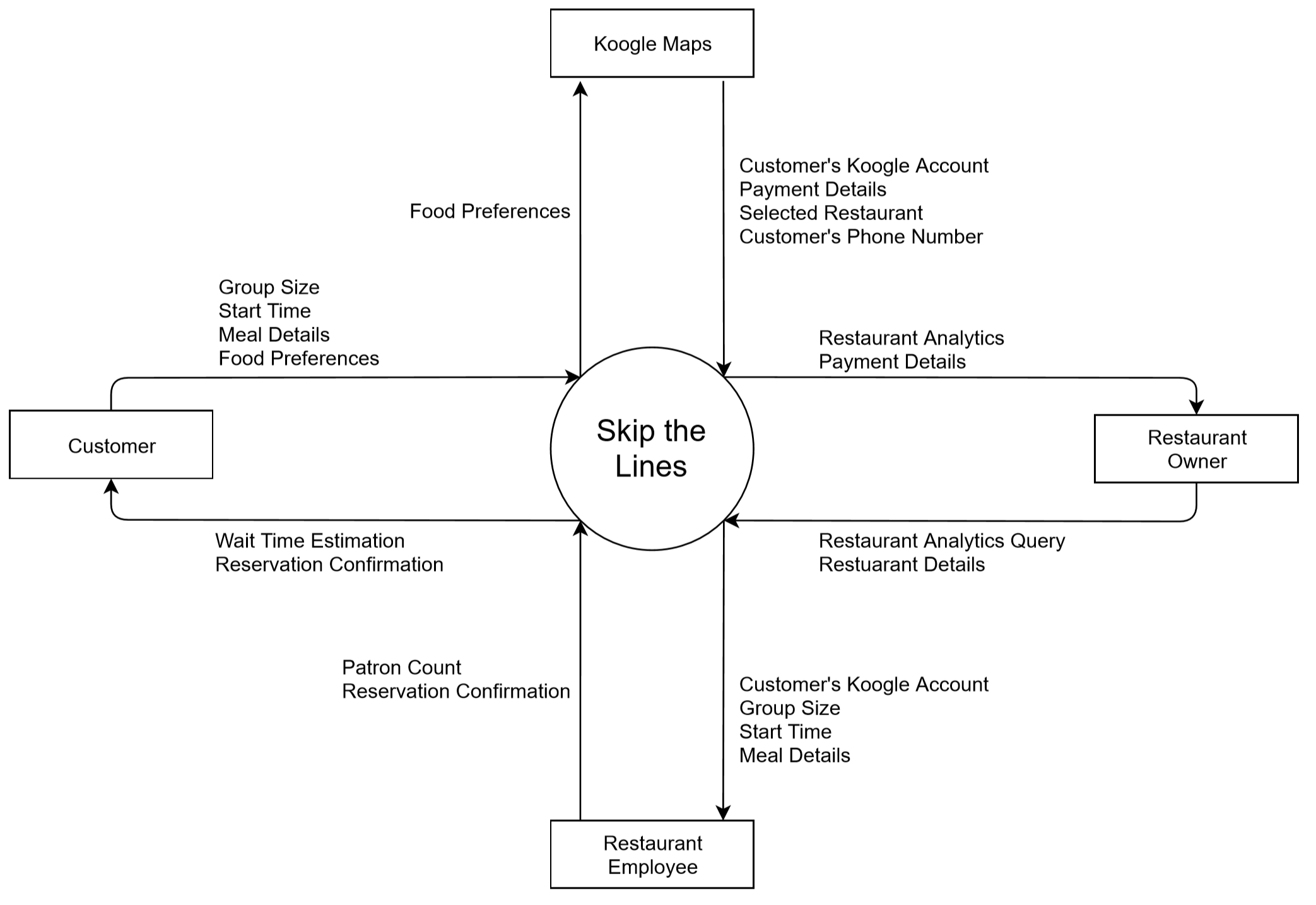
The entity relationship diagram below visually describes relationships between entities within Skip the Lines. The entities within the entity relationship diagram are described within data dictionary.



|  |  |
| --- | --- |
| **Data Dictionary** | |
| **Object Class** | **Attributes** |
| Customer | Name, Koogle account, email, and phone number of the customer |
| Koogle Account | Customer’s information, phone number, and payment details. |
| Payment Details | Customer’s payment method for pre-ordering food, which is contained in customer’s Koogle account. |
| Pre-Order | List of food ordered by the customer, which is used for their restaurant reservation. |
| Reservation | Customer makes a reservation, which may contain some pre-order information, and it is sent to Restaurant employee. |
| Reservation Confirmation | Confirmation notification with reservation time, holding time. |
| Restaurant Employee | Gets reservation request from a customer, the reservation gets approved, and a notification confirmation is sent back to the customer. |
| Restaurant Owner | Creates and manages Restaurant Skip the Lines account. |
| Restaurant | Location, menu, and hours of operation. |
| Restaurant Skip the Lines Account | Contains Restaurant’s attributes, and Restaurant’s analytics, which is managed by Restaurant owner. |

## 6.2 Context Diagram

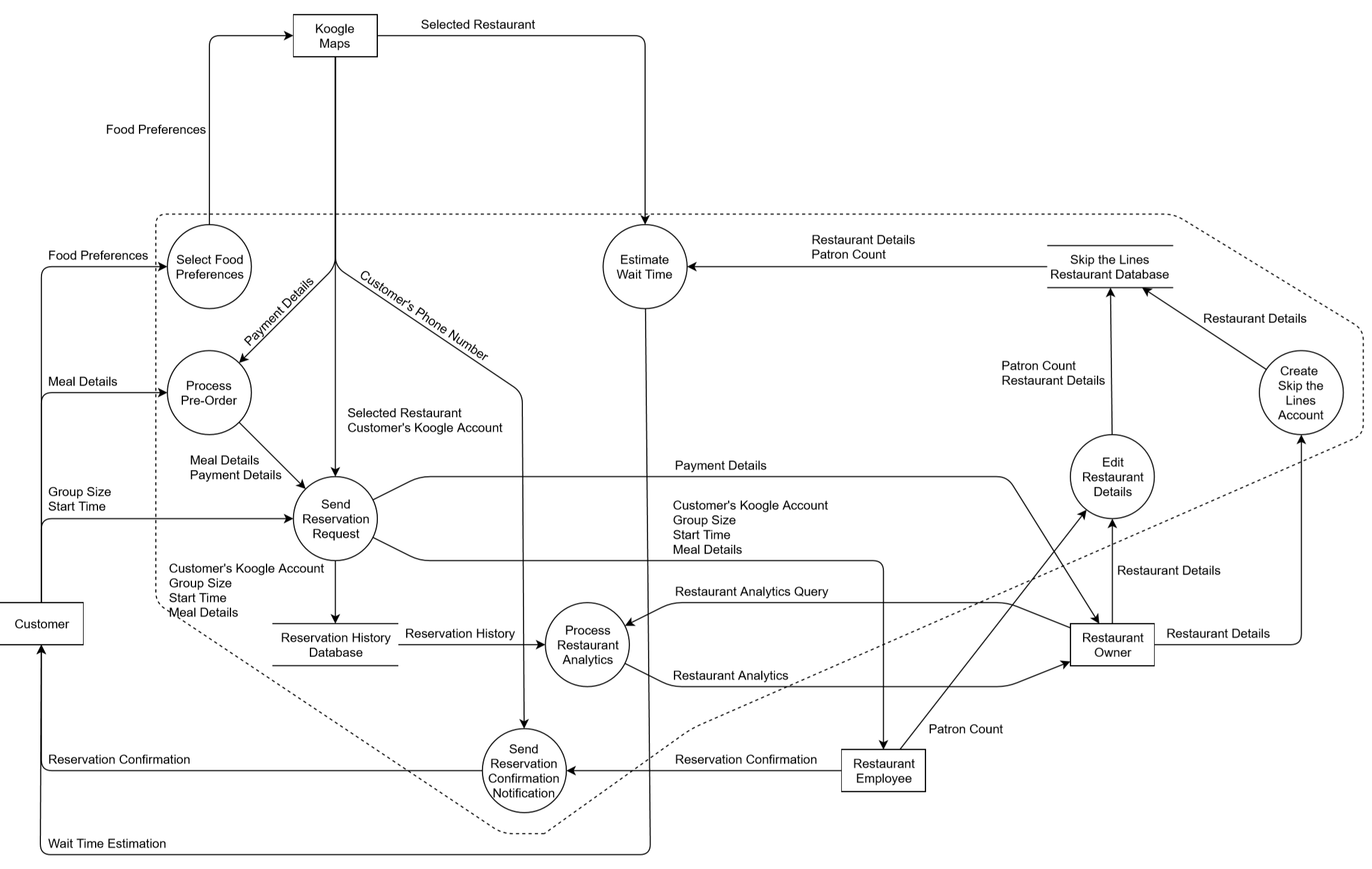
Below is the context diagram, which shows the general relationship and interaction between our external entities and Skip the Lines system.



## 

## 6.3 Data Flow Diagram Level 1

Below is the Level 1 Data Flow Diagram showing how the external entities (Customer, Koogle Maps, Restaurant owner, and Restaurant user) interact with the processes included in Skip The Lines. It provides a detailed internal structure of Skip the Lines.



## 

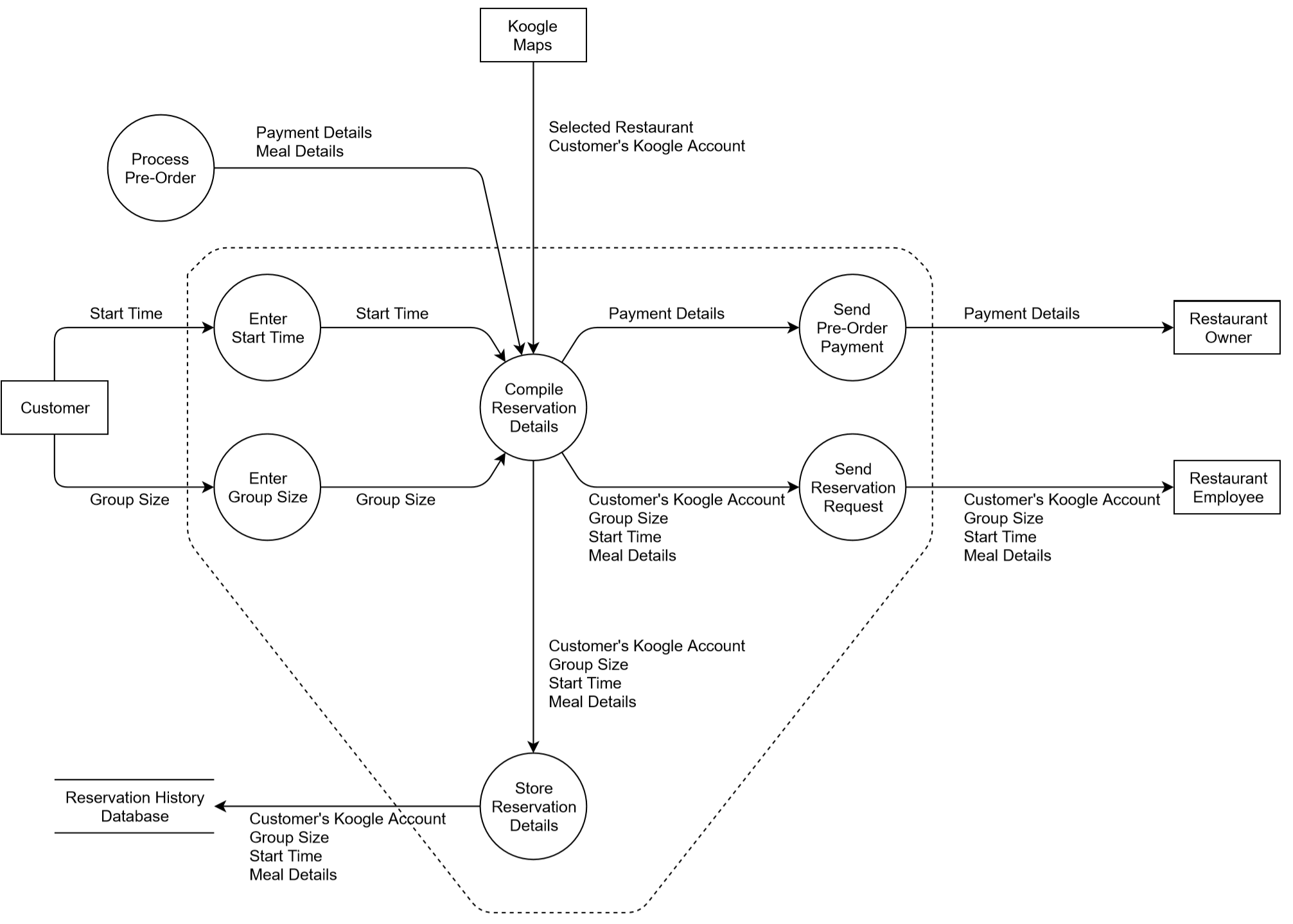
## 

## 

## 

## 6.4 Data Flow Diagram Level 2

Below is the Level 2 Data Flow Diagram showing how the external entities (Customer, Koogle Maps, Restaurant owner, and Restaurant user) interact with the “Send Reservation Request” process bubble in the Data Flow Diagram level 1. It provides a more detailed internal structure of Skip the Lines than the Level 1 Data Flow Diagram.



# Appendix: Issues List

Unresolved issues:

* Discuss the layout of a customer’s interface, including screen layout constraints, standard buttons and functions.