|  |
| --- |
| T27 |
| SFD Dispatch  Scope and Mission |
| [Surrey Food Delivery] |
|  |
| **Daniel Park**  **Bill Xue**  **Kent Huang**  **Terence Leung** |
|  |

|  |
| --- |
|  |

**Table of Contents**

[1.](#_gjdgxs) Document Version 3

[2.](#_30j0zll) Team Contact Information 3

[3.](#_1fob9te) Description 3

[4.](#_3znysh7) Proposed Solution 4

[5.](#_2et92p0) Stakeholders and Users 4

[6.](#_tyjcwt) Proposed Technology 5

[7.](#_3dy6vkm) Assumptions 5

[7.1.](#_1t3h5sf) Non-Technical Assumptions 5

[7.2.](#_4d34og8) Technical Assumptions 5

[8.](#_2s8eyo1) Project Goals, Tasks, Features 6

[9.](#_17dp8vu) Deliverables 7

[10. Out of Scope](#_3rdcrjn) 8

[11. Existing System](#_26in1rg) 8

[12. Diagrams](#_lnxbz9) 9

[12.1.](#_35nkun2) Client Diagrams 9

[12.2.](#_1ksv4uv) Team Diagrams 9

12.2.1. Overall Design & Activities 9

12.2.2. Sample WorkFlow 11

12.2.3. Wireframes 13

[13. High-Level Schedule](#_44sinio) 15

[14. High-Level Work Breakdown](#_2jxsxqh) 16

[15. Risks and Issues](#_z337ya) 18

16. Appendix 19

## Document Version

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Description** | **Date** | **Author(s)** |
| **1** | Created document, this is the original plan after the team’s first discussion with client | 28/04/2017 | Daniel Park  Bill Xue  Kent Huang  Terence Leung |
| **2** | Added Gantt chart | 01/05/2017 | Terence Leung |
| **3** | Review and proofread and add content | 04/05/2017 | Terence Leung |
| **4** | Added original project requirements as appendix | 06/05/2017 | Kent Huang |

## 

## Team Contact Information

**Team:**

|  |  |
| --- | --- |
| Daniel Park  [daniel51\_1994@hotmail.com](mailto:daniel51_1994@hotmail.com)  604-700-5909 | Kent Huang  [kenthuang426@gmail.com](mailto:kenthuang426@gmail.com)  604-618-8770 |
| Terence Leung  [terence.cy.leung@gmail.com](mailto:terence.cy.leung@gmail.com)  604-338-2232 | Bill Xue  [billxue9141@hotmail.com](mailto:billxue9141@hotmail.com)  778-866-9900 |

**Supervisor:**

Bill Klug

[bill\_klug@bcit.ca](mailto:bill_klug@bcit.ca)

## Description

Surrey Food Delivery is a food delivery service based in Surrey. Currently, customers and restaurants contact the owner/dispatcher to request a delivery for a food or product. The dispatcher receives either a text or a phone call describing the pickup location, the item to be purchased, and the destination of the customer. If the dispatcher receives a text or call from a customer, he allocates a driver to purchase the order. The driver will then complete the order.

The task of the owner/dispatcher is quite heavy and will continue to grow as the business grows. The owner currently receives many text messages and calls per day due to the increasing number of customers. It is therefore difficult for the owner to manage multiple orders from the customers and restaurants. Also, as the business grows, the number of drivers required to meet demands grow making the owner/dispatcher’s job even more unmanageable.

## Proposed Solution

Our team will develop 2 Android apps. One is the customer end (home and restaurant customers) and one is the company end (driver, dispatcher, admin). The customer or restaurant can request a delivery service. A customer will be able to use the app to order food or any other item. The information of this order will then be sent to the customer end. This will be the company app and seen by the dispatcher. The dispatcher can then allocate the order to the driver. For this project, the dispatcher will manually choose the driver. In future iterations, this feature may be automated. The benefit of this system is that all the communication between the different views can communicate between each other in an automated way.

The solution was derived from the original project specification made the clients. It has been reviewed and reorganized and restructured to fit the time allotted for this project. The original requirements by the client can be found in the [appendix](#_vggmsu7vs8dn).

## Stakeholders and Users

**Asad Khan** - Owner of Surrey Food Delivery

Work number: 604-783-3370

Website: [www.surreyfooddelivery.com](http://www.surreyfooddelivery.com)

**Ayesha Khan** - Main contact. Wife of the owner, helps manage the current business.

Cell number: 604-537-8681

Email: [aftab\_ayesha@hotmail.com](mailto:aftab_ayesha@hotmail.com)

**Users**:

Regular customers

Restaurants Customer

Dispatcher

Drivers

Business owner.

**Development Team - T27**:

Bill Xue

Kent Huang

Daniel Park

Terence Leung.

## Proposed Technology

1. **Java**: the language for Android development.
2. **Android studio**: the IDE for Android development.
3. **Firebase**: the backend development platform provides the database storage and account registration and login logout functionalities.
4. **Nosql**: the language used in Firebase real-time database.
5. **Object-oriented design**: the design techniques used in application design.
6. **Whatsapp**: communication tool for the development team.
7. **Github**: collaboration tool for the development team.
8. **Trello**: Scrum board for the development team to organize the development tasks.

## Assumptions

## Non-Technical Assumptions

1. The team will complete phase 1 of the project (developing an Android app).
2. The next phase of the project may be completed by another team.
3. Development team will not be responsible for publishing the app to app store.
4. Student team is responsible for purchasing any equipment or paid software not provided by BCIT or Surrey Food Delivery.
5. Student team is not responsible for any other platform other than Android.
6. Student team is not responsible for providing the Google account registered for Firebase platform services; Surrey Food Delivery is responsible for registration of Firebase services in production phase of the application.

## Technical Assumptions

1. The team will determine the backend and front-end technologies.
2. Project will be developed using Java and Firebase.
3. The system will consist of two Android applications; one application is for customers and restaurants and another application is for the dispatcher and drivers.

## Project Goals, Tasks, Features

The goal of the current project is to create an app so that customers and restaurants can easily request a delivery to Surrey Food Delivery, and to also layout the foundations for an automated system that will reduce the tasks of the owner. In order to achieve this, a customer, dispatcher, and driver views will be created.

The clients have specified the overall project requirements including the end goal of the entire project. The most important requirement is to develop a system that would reduce the tasks of the owner/dispatcher. Therefore, an automated system is needed to automatically assign drivers tasks. Another requirement is to send a notification to the dispatcher through the app about any new deliveries, and notifications to the driver and customer about the order and the order status. There must also be a payment system and a customer, dispatcher, and driver views in which the communication should be handled by the dispatcher.

The final delivered project will be an Android app that will be used by the customers, dispatcher, and the drivers. The app from the customer’s point of view will be able to request a delivery by filling out a form within the app. The customer may choose to register, login, or login as a guest. Once the form is completed, it will viewable by the dispatcher. A notification will be sent out to the customer via the app with regards to their order status.

As for the dispatcher, he will log in and be able to manage all the new deliveries and drivers. The dispatcher will also receive notifications about new deliveries and view the status of each order and the drivers.

The drivers will also be able to log in and receive notifications whenever a new delivery is assigned to them. Through the app, the driver can accept incoming deliveries, set the estimated arrival time, and complete the given task.

## Deliverables

|  |  |
| --- | --- |
| **Deliverable** | **Description** |
| Design application wireframes | Read through the project description and come up with the application interfaces that work for the business logic and data communication |
| Implement application UI | Implement the UI agreed on the application wireframe design |
| Implement application logic control (Java code) | Look into Firebase Android development  and design and implement the system that can handle mainly the account management and real-time communication or notification among different ends. |
| Implement application database on Firebase platform | The database on Firebase Platform will be used to handle the data storage of account management and transactions recording. |
| Set up Firebase environment | The Firebase configuration, the backend services, will be finished. |
| GPS integration for the driver’s current location | Drivers’ location will be uploaded to real-time database and will be sent to dispatcher at certain intervals. |
| Working prototype | integrate the UI with the Firebase services to produce the working prototype of the dispatch system. |
| handoff source code, compile apps | We will meet up with our clients and discuss the development decisions and present progress with them throughout the project.  At the end, we will formally handoff the source code and application to SURREY FOOD DELIVERY. From that on, we won’t be making changes to the code. |

## 

## Out of Scope

The project is to be separated into different phases. Our team will complete phase 1 of the project which is to complete an Android app and layout the foundations for the other phases to come. The following tasks and features will not be included in our project scope:

1. iOS version of the app
2. Web version of the app
3. Payment system
4. Algorithm to determine the best suitable driver for the delivery
5. Automated assignment of drivers to deliveries
6. A delivery orders queue for drivers (only 1 order at a time)
7. Delivery tracking for customer

## Existing System

The current system consists of customers (whom are home or restaurants) calling or texting Asad (the owner) with the information regarding a delivery they want. The text or phone call would contain name, number, address, products to be picked up, and from where the product should be picked up. The company’s service is not limited to food delivery, but anything that the customer asks, such as laundry, alcohol or cigarette. From there, Asad would choose a driver close to that delivery location and text that information to the driver. The driver then text Asad back when he/she has picked up the products allowing Asad to text or call back the client with an estimate time of arrival and price. As of now, Asad does not have a system to show where the drivers are, so he keeps track of his drivers by memorization.

For customers to find information (phone, price etc.) about the delivery service, it can currently be found on the company’s website at <http://www.surreyfooddelivery.com>. This system on the web site is currently fully manual and require Asad to oversee all transactions constantly.

The current scheduling method is done via a honor system. However, there are systems on the market in which this company can benefit from purchasing opposed to developing one from scratch,

## Diagrams

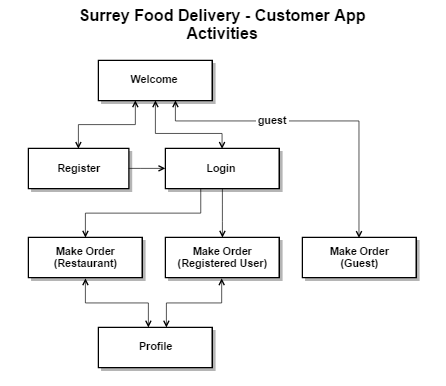
## Client Diagrams

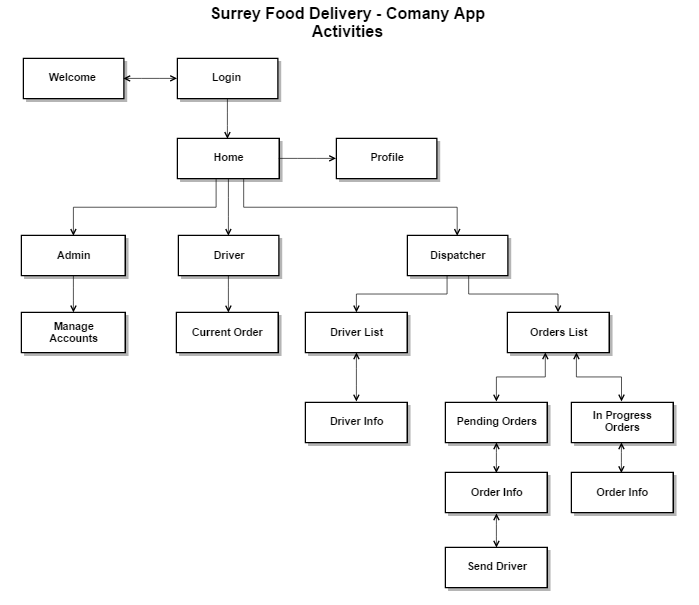
Client did not provide any diagrams, they would like the team to build an app from scratch.

## Team Diagrams

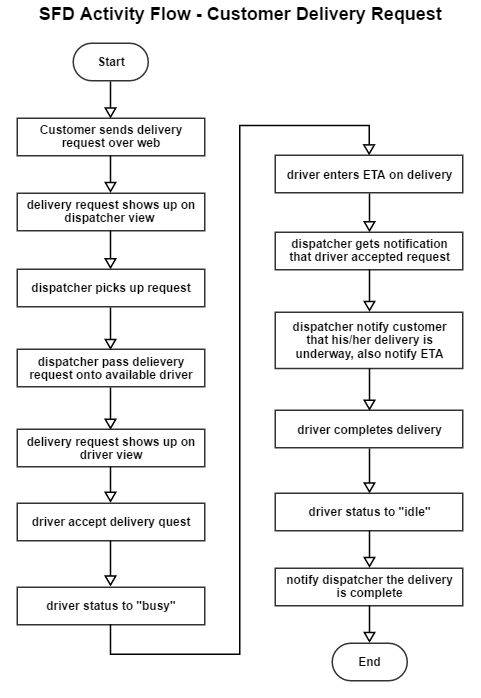
Below are some sample designs for the app activities, workflow, and wireframes.

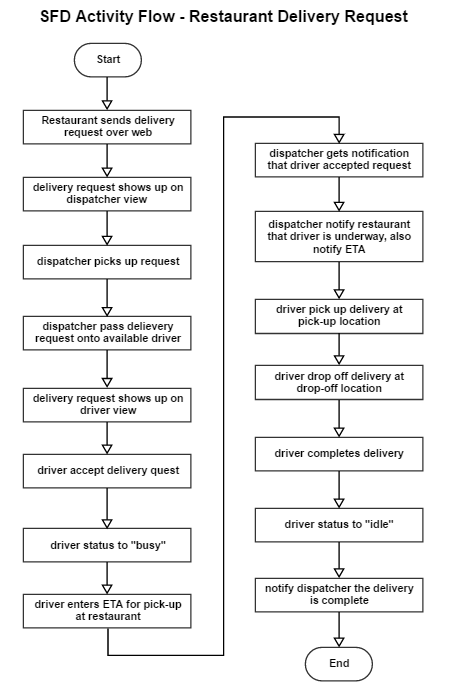
***12.2.1. Overall Design & App Activities***





***12.2.2. Sample Workflow***





***12.2.3. Wireframes***

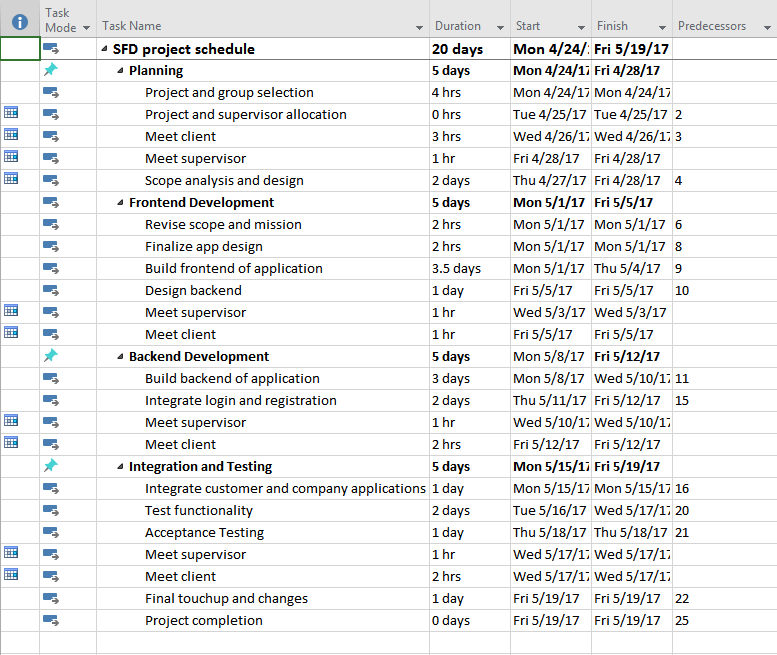
|  |  |  |
| --- | --- | --- |
| *Customer App*  *Welcome Page* | *Customer App*  *Login* | *Customer App*  *Register* |
| *Customer App*  *Guest View*  *New Order* | *Customer App*  *Customer View*  *New Order* | *Customer App*  *Restaurant View*  *Request Driver* |
| *Company App*  *Driver View*  *Idle* | *Company App*  *Driver View*  *Take Order* | *Company App*  *Driver View*  *Take Order (accept)* |
| *Company App*  *Driver View*  *Current Order* | *Company App*  *Driver View*  *Profile Page* | *Company App*  *Dispatcher View*  *In Progress Orders List* |
| *Company App*  *Dispatcher View*  *Select Pending Order* | *Company App*  *Dispatcher View*  *Select Driver* | *Company App*  *Dispatcher View*  *Drivers List* |
| *Company App*  *Dispatcher View*  *Pending Orders List* |  |  |

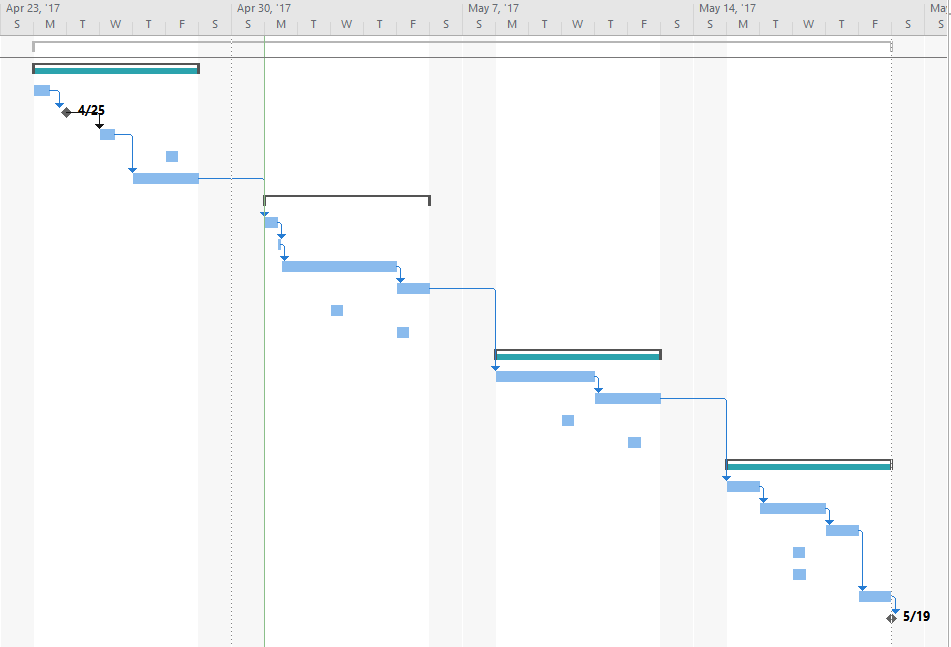
## High-Level Schedule

|  |  |
| --- | --- |
| **Stage** | **Schedule** |
| Week 1 - Planning | - Project and group selection  - Client and supervisor meetings  - Gather requirements and establish scope |
| Week 2 - Frontend | - Revise scope and mission.  - Finalize app design.  - Build frontend of app (both customer and company app).  - Learn Firebase real time database.  - Design backend database.  - Present frontend to client for revision |
| Week 3 - Backend | - Implement backend database.  - Integrate login & register functions into database.  - Present system iteration to client for revision. |
| Week 4 - Integration and Testing | - Integrate database into both app (customer and company).  - Test functionalities.  - Acceptance testing.  - Final changes and touch up.  - Project completion and delivery. |
| Week 5 - Presentation | - This week is dedicated to the practice and delivery of the presentation. |

## High-Level Work Breakdown

|  |  |  |
| --- | --- | --- |
| **Owner** | **Description** | **Completion Date** |
| Terence | Communication between team, client and supervisor | N/A |
| Daniel | Documentations | N/A |
| Bill | Design | N/A |
| Kent | Problem solving, find out new technology | N/A |
| Everyone | Wireframes for both app | Week 1  Saturday |
| Terence  Daniel  Kent  Bill | Scope and mission document - final version | Week 2  Wednesday |
| Daniel  Terence | Frontend implementation | Week 2 |
| Daniel  Kent | Customer App UI | Week 2  Tuesday |
| Terence  Bill | Company App UI | Week 2  Thursday |
| Everyone | Present frontend to client for revision | Week 2  Friday |
| Everyone | Learn how to setup and use Firebase database. | Week 2  Sunday |
| Kent  Bill | backend implementation | Week 3 |
| Kent  Bill | Database design | Week 3  Monday |
| Kent  Bill | Database implementation | Week 3  Wednesday |
| Terence  Daniel | Integrate login & register functions into database | Week 3  Thursday |
| Everyone | Present system iteration to client for revision | Week 3  Friday |
| Kent  Bill | Learn how Firebase real time database works | Week 3  Sunday |
| Terence  Daniel | Find out how to send notification to device when the database updates | Week 3  Sunday |
| Kent  Bill | Integrate database into both app (customer and company) | Week 4  Monday |
| Terence  Daniel | Integrate notifications into both app when database updates | Week 4  Wednesday |
| Kent  Bill | Acceptance testing | Week 4  Thursday |
| Terence  Daniel | Functionality testing | Week 4  Thursday |
| Everyone | Build app apk  Handoff app to client | Week 4  Friday |
| Everyone | Final presentation powerpoint slides | Week 4  Sunday |
| Everyone | Final presentation practice | Week 5  Monday |





## Risks and Issues

|  |  |  |
| --- | --- | --- |
| **Risk or Issue Description** | **Potential Impact** | **Mitigation/Resolution Steps** |
| Team is inexperienced with Firebase | May take up more time than expected to learn and use Firebase | Team will come together and start learning early |
| Project is extremely large with only 3 weeks | Project may not be completed on time | Project has been divided into multiple modules to be completed by other projects |
| Non-technical clients | Clients request functionality incorporated to the program which promotes high coupling | Persuade clients that certain functionality should not be included into the system |
| Client may not be accepting of only developing on Android platform | Project cannot be done | Negotiate and explain how the project must be done in Android |
| Communication between customer and company app may fail | If the 2 applications cannot communicate. The customers will not be able to make orders | Heavy testing on this particular area |

## Appendix, Original Requirements of Project

* Customer should be able to order through
* Webpage: When the orders are placed through web page, it should send a notification to the dispatcher through the app or text about a new delivery
* App
* Phone call: Dispatcher can enter the info into the app (dispatcher module) (similar to the case as a customer would through the app) and that would notify the driver through the driver app module.

Thus, customer places the call for delivery services either through phone call, webpage, or App.

* The avail driver accepts the order for new delivery and this in turn sends the notification to the customer through the app with the following info:
* that the delivery has been accepted
* ETA
* cost of the delivery charge depending on
* #of stops (1 stop - 12$, 2 stops - 20$ and so on.)
* urgency of the delivery e.g. Rush etc.
* Amount of the order, plus the cost of food
* Requesting customer to indicate the mode of payment: Cash / Debit / Email transfer (The functionality should be incorporated into the app and should have security feature)

The driver then purchases/picks the item from the Restaurant or Store E.g. Fast food restaurant / gas stations / corner stores etc.

Indicates through the app that the delivery has been picked up and then delivers to the customer.

Once the driver is done the delivery he should be able to notify through the app: Delivery is done; And be able to bring himself in the avail mode for the next delivery.

3 App Module and Webpage Development:

* Customer

Fields and functionality:

* Customer can place order for home delivery/restaurant. Initial info needed: Name, address, Phone number, Address of Restaurant
* Make a drop down for common stores/restaurants/corner stores/gas stations etc. (Store the 10 last recent orders)
* Distance (how would this be done???)
* If more than 6 km then calculate automatically.
* Mode of payment: Cash/debit/wire transfer/mcard (3% surcharge if paying by mcard). Stops, amount of food, urgency/short notice, Approximate time needed.
* Misc field: Customer should have the functionality of giving feedback/rating for the service and the driver - once the delivery is done (Unsatisfactory/good/ok/best, Fast accurate order, Courteous friendly service-driver points, to be able to like us on FB, Twitter etc.)
* Dispatcher

What fields and functionality:

* should be able to see which driver is logged on,
* should be able to see what all drivers are currently doing deliveries,
* should be able to see the location of each driver at any given time;
* should be able to get notification once the deliveries are picked up and once done.

At the end of the day:

* should be able to sort the deliveries by driver;
* sort the deliveries by mode of payment; sort the deliveries by stars given;
* sort the deliveries by each day; sort the deliveries by restaurant etc.
* should be able to see the customer stored info;
* be able to schedule (in a spreadsheet) the drivers for the week and send the notification to the drivers about the assignment
* indicate the penalties for missing days
* rules for drivers
* indicate the penalties for short notice
* Driver
* Name to be assigned
* number to be assigned
* ID to be assigned
* Should be able to log in.

The login will only be granted by the dispatcher on a day to day basis (meaning administrator/dispatcher should give the OK before the driver can login)

What fields and functionality:

* once the customer places the order, it should make the delivery available to the driver;
* once the driver accepts the delivery, it should be able to recommend the nearest locations for the services asked: e.g. nearest McDonald, chevron etc.
* Once the delivery has been picked up, the driver should be able to indicate to both the customer and the dispatcher (through the app).
* the driver should be able to alert the customer 5 mins, before arrival with the total amount due and the mode of payment, so that the customer can be ready with the payment.
* driver should be able to enter the cost of food upon pick up and this