Matthew Rathwell and Zakaria Rahman

CSC 472 Mobile Application Development Final Project

11/24/2015

Zoom Truck



Table of Contents

[ Contact Information 2](#_Toc436149180)

[1. Purpose 2](#_Toc436149181)

[2. Design Requirements 2](#_Toc436149182)

[2.1 High Level Design Requirements 2](#_Toc436149183)

[2.2 Detail Design Requirements 3](#_Toc436149184)

[2.2.1 Home 3](#_Toc436149185)

[2.2.2 Order Name Activity 4](#_Toc436149186)

[2.2.3 Food Item Activity (Fragment View) 5](#_Toc436149187)

[2.2.4 Food Item Details (Fragment View) 6](#_Toc436149188)

[2.2.5 Master/Detail tablet view 7](#_Toc436149189)

[2.2.6 Summary Activity 8](#_Toc436149190)

[2.2.7 Mail Intent 9](#_Toc436149191)

[3. System Requirements 10](#_Toc436149192)

[4. Team Contribution Distribution 10](#_Toc436149193)

[5. Conclusion 11](#_Toc436149194)

# Contact Information

The primary contact for questions regarding this document may be directed to the Attention of personnel given below:

Matthew Rathwell

Zakaria Rahman

Project Builders

Email: [mrathwel@mail.depaul.edu](mailto:mrathwel@mail.depaul.edu), [zrahman2@mail.depaul.edu](mailto:zrahman2@mail.depaul.edu)

# Purpose

The purpose of this document (Design Specification, Functional and Technical Requirements) is to formally define the high-level requirements for the system in terms of the needs of the end users as per the final project requirements.

# Design Requirements

## 2.1 High Level Design Requirements

|  |
| --- |
| Design Description |
| Home |
| Food Truck Locator Map |
| List Food Item |
| Food Item Details |
| Order Summary |
| Mail Intent |
| Gestures |
| Master/Detail fragments |

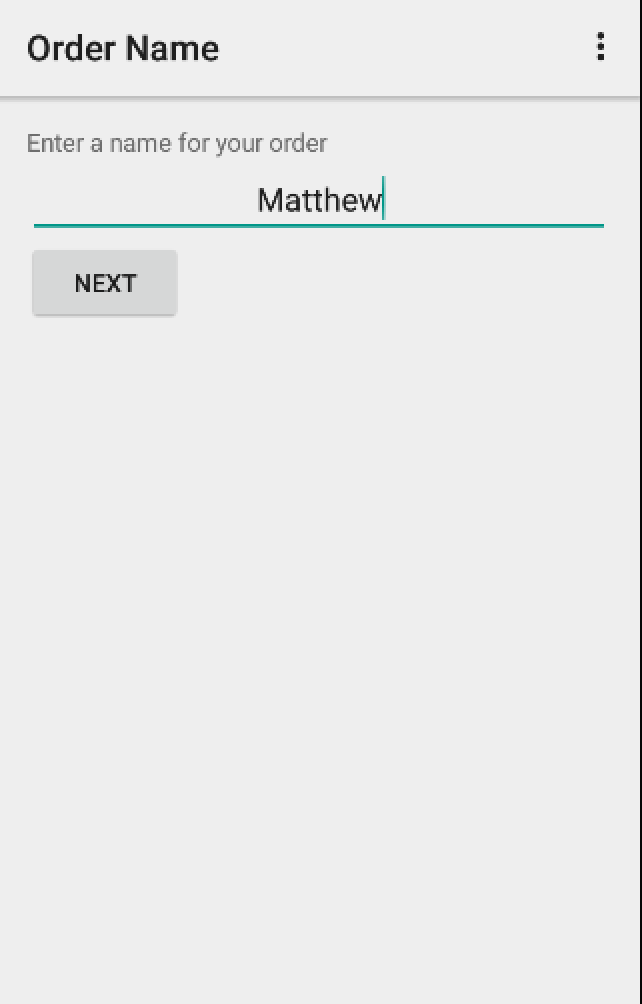
## 2.2 Detail Design Requirements

### 2.2.1 Home

The Home screen contain the following 2 buttons (Linear Layout -> vertical)

* “Find Zoom Truck” => Button -> implicit intent to open maps application which currently passes a static addess to the google maps API which then can be used to get directions to the zoomtruck. Later iterations will include contacting a geolocation web service that tracks the Zoomtruck, that is currently not possible because the zoomtruck doesn’t exist.
* “Place an Order” => Button -> This button navigates to the Name Activity.

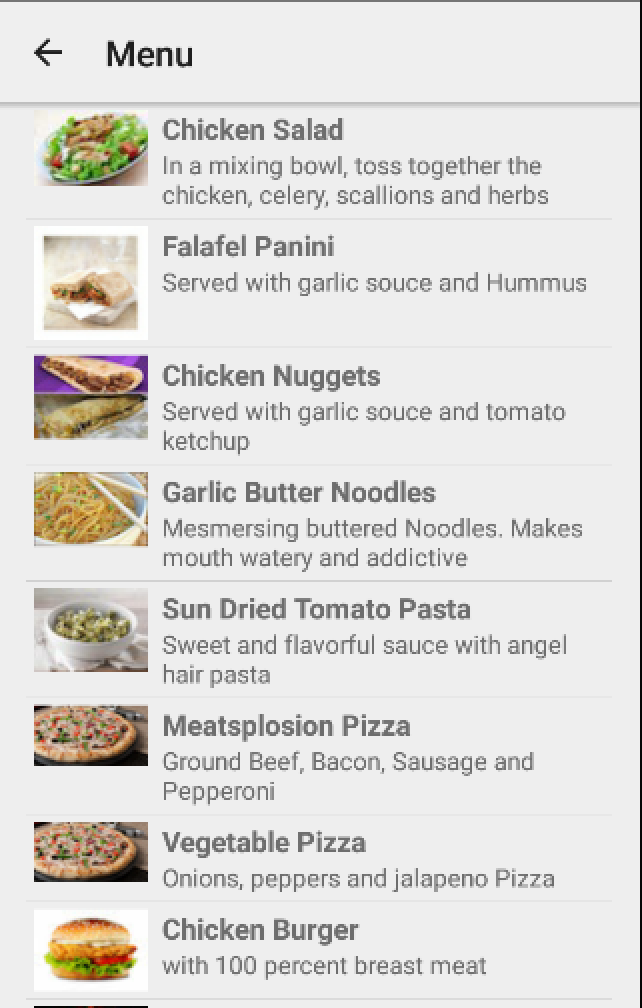
### 2.2.2 Order Name Activity



This activity gives names to the order. This activity should have the following: (Linear Layout -> aligned vertical)

* Label Description: “Enter a name for your order” => TextView
* Customer Name => EditText
* Next => Button -> stores the customer name editText field into the Order Singleton’s name field and starts the Food Item Details Fragment.

### 2.2.3 Food Item Activity (Fragment View)

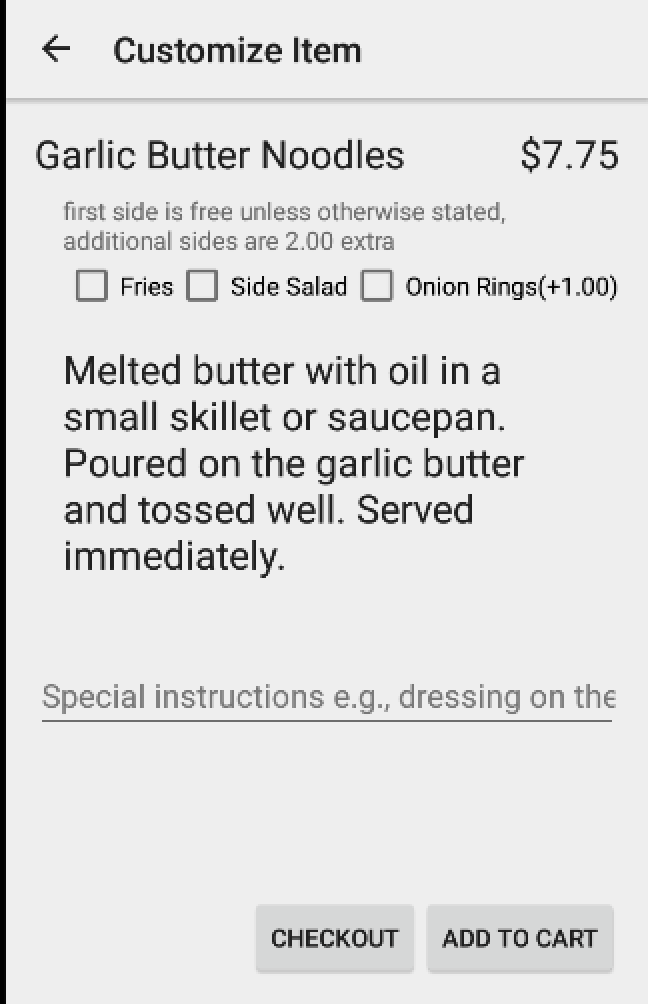


This activity contains the list of Food Items. Each food item has the following sections in its layout: (Linear Layout -> aligned horizontal)

* Image of the Food Item on the left of each list item. => ImageView
* Label of the food item. => TextView
* Short description of the food item below the label. => TextView

Each list item is a link to the Food Item Details activity of the Food Item.

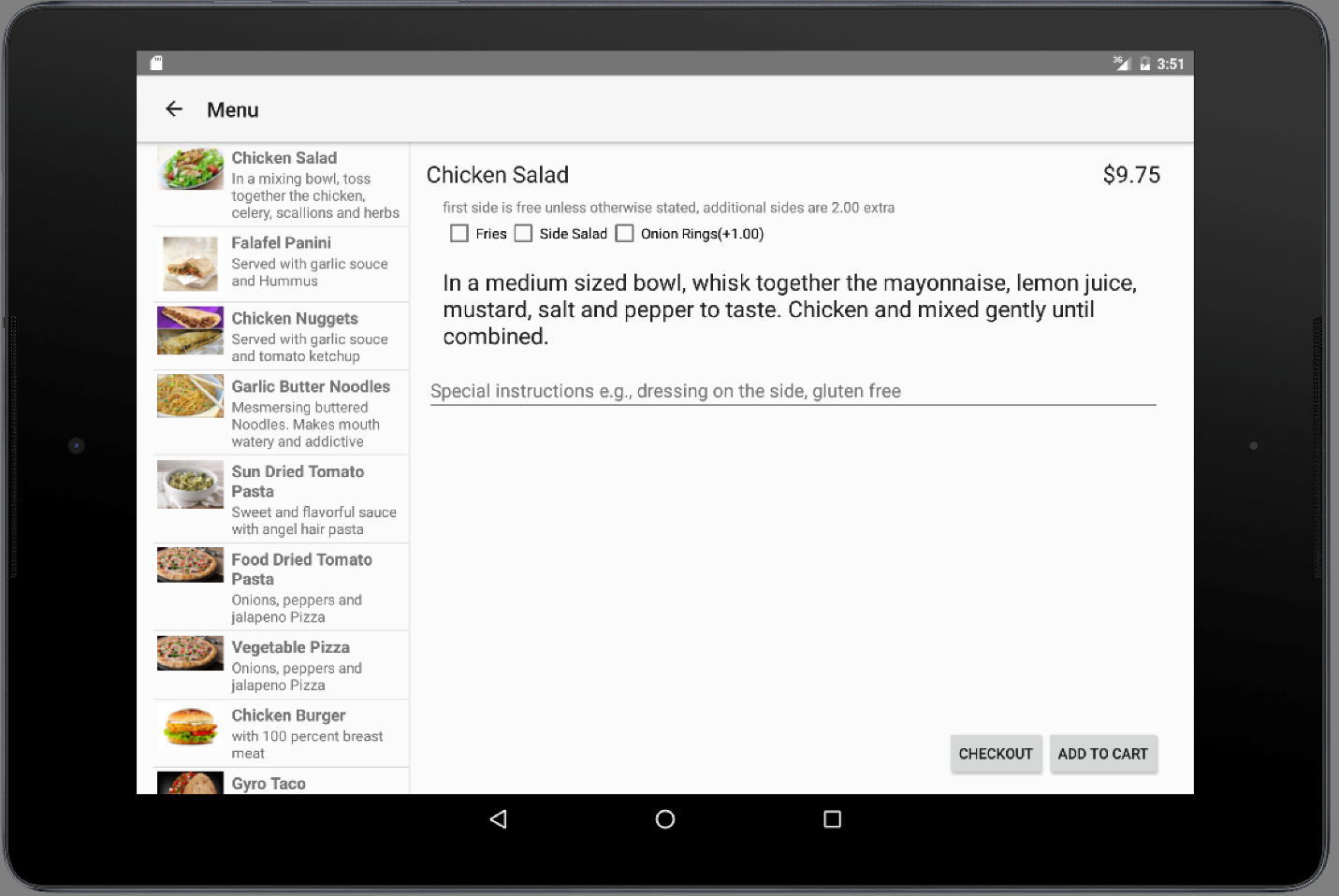
### 2.2.4 Food Item Details (Fragment View)



This Activity contains the following sections: (Relative Layout)

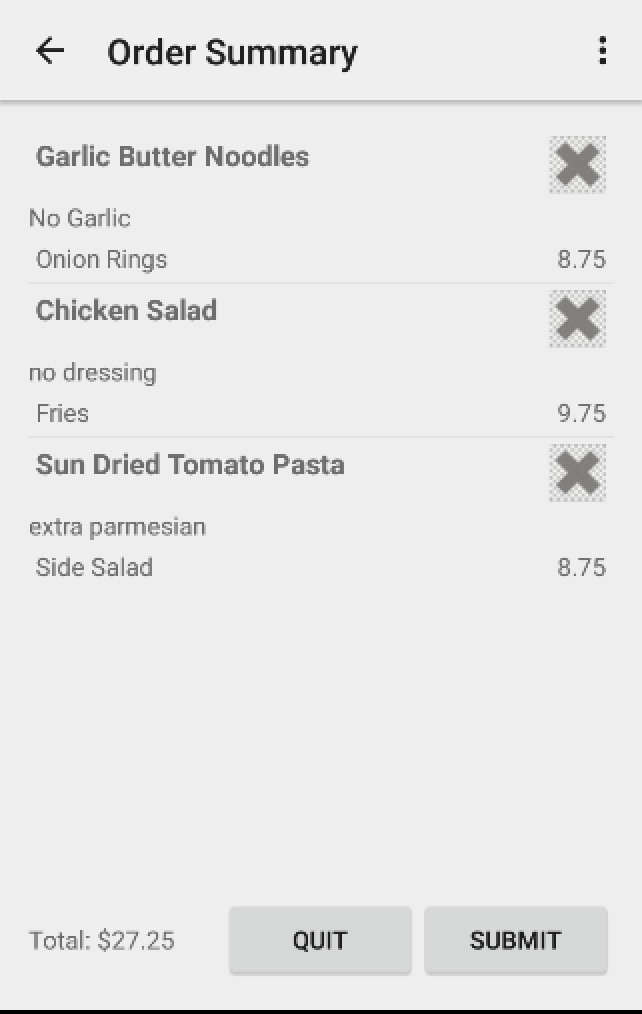
* Label of the Food Item => TextView
* Price of the Food Item and related add ons with the Food Item. => TextView
* Briefing about the checkboxes => TextView
* Sides => Checkbox group: aligned horizontally, selecting onion rings adds $1.00 to the total and each side past the first adds an additional $2.00
* Long Description of the Food Item => TextView
* Special Instructions => EditText
* Button group (Relative Layout)
  + Checkout => Button -> link to the Summary Activity taking the current item selected.
  + Add to Cart => Button -> adds food item to Order Singleton and the Summary Activity.

### 2.2.5 Master/Detail tablet view



See above for details

### 2.2.6 Summary Activity



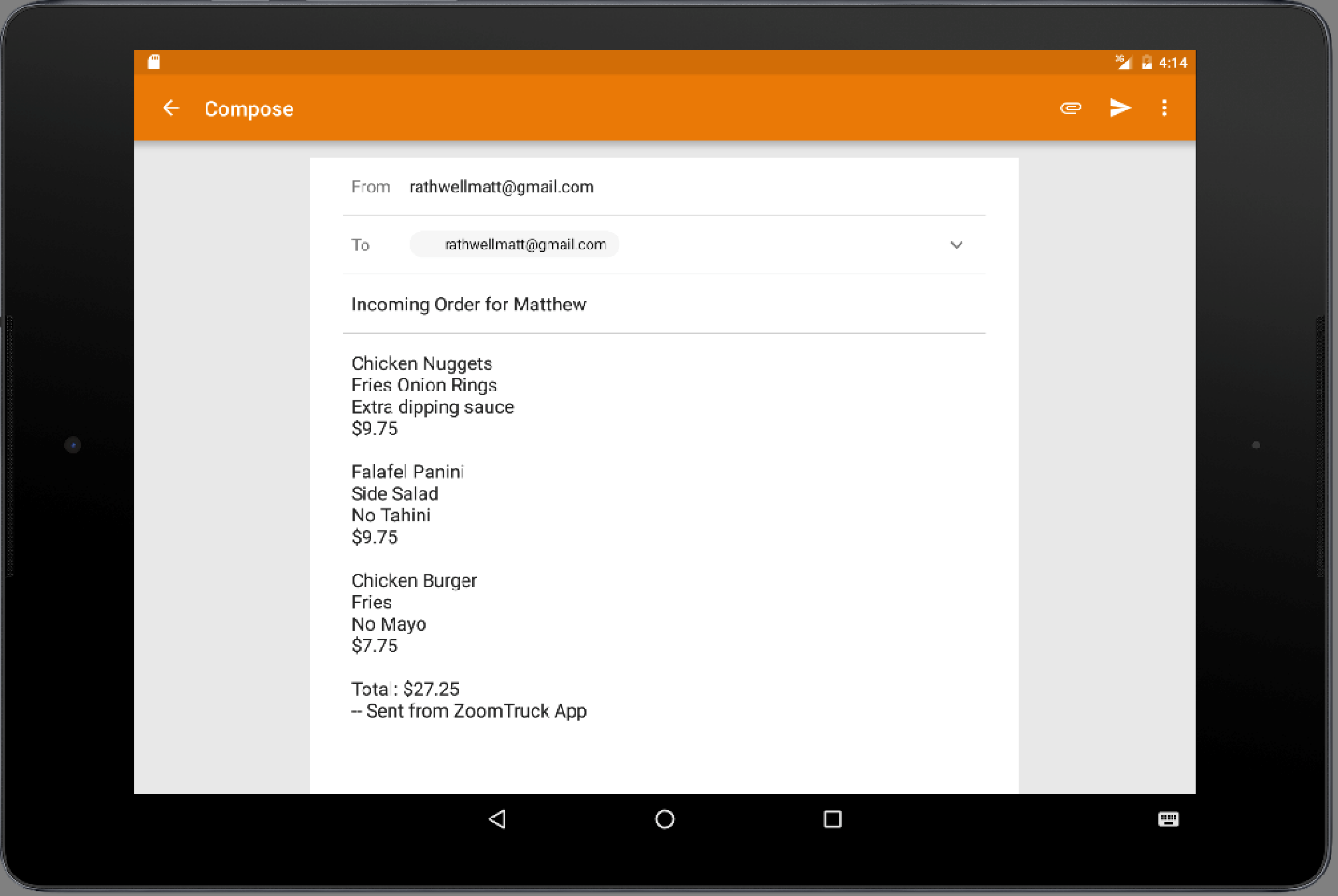
This activity contains the following: (Linear Layout -> vertical)

* Label of Food Item => TextView
* Close image => ImageView -> This image has a click listener which removes this item from the Order singleton and rebuilds the list on the summary activity.
* Special Instruction => TextView any special instructions/customizations the customer wants for that particular item.
* Sides => TextView : side items listed
* Cost => TextView : in line with sides. Total cost of the item including sides
* Total => TextView : at the bottom adds up the total cost of all order items, repopulates when one of the order items is removed.
* Quit => Button : when pressed appropriately finishes all of the intents in the program.
* Submit => Button : sends the order with the customer’s name at the top using an email intent.

This activity also supports swiping if one is too lazy to use the back button at the top or submit button at the bottom. Merely swipe right to submit and swipe left to go to go back to previous page.

### 2.2.7 Mail Intent

The Mail Intent uses whatever mail client the customer selects on their device as an implicit intent is used to send it to the food trucks email address(currently my email rathwellmatt@gmail.com).



# System Requirements

* Devices Compatibility: Phone and Tablets
* Minimum SDK Version Level : 22
* Target SDK Version: 23
* Compile SDK Version: 23
* Minimum Android Version: 5.1

# Team Contribution Distribution

|  |  |  |  |
| --- | --- | --- | --- |
| Sr Num | Module | File Type | Completed by |
| 1 | Main Activity | XML Layout | Matthew |
| 2 | Name Activity | XML Layout | Matthew |
| 3 | Activity Food Item List | XML Layout | Matthew |
| 4 | Food Item List Item | XML Layout | Matthew |
| 5 | Activity Food Item Detail | XML Layout | Matthew |
| 6 | Fragment Food Item Detail | XML Layout | Zak/Matthew |
| 7 | Activity Summary | XML Layout | Matthew |
| 8 | Summary List Item | XML Layout | Zak |
| 9 | Mail Intent | XML Layout | Matthew |
| 10 | MenuContent | Java | Zak |
| 11 | Food | Java | Zak |
| 12 | FoodItemDetailActivity | Java | Matthew |
| 13 | FoodItemDetailFragment | Java | Matthew |
| 14 | FoodItemListActivity | Java | Matthew |
| 15 | FoodItemListFragment | Java | Zak |
| 16 | Gestures | Java | Zak |
| 17 | NameActivity | Java | Matthew |
| 18 | Order | Java | Matthew |
| 19 | OrderItem | Java | Matthew |
| 20 | SummaryListActivity | Java | Matthew |
| 21 | Documentation | Doc | Zak/Matthew |

# Conclusion

A great Deal of this project was pair programmed so our Division of labor really is an approximation of who was primarily responsible for the file’s completion. Also sometimes as in the case of gestures one of us would work on it for hours but fail to get it working and the other would actually make it work.

If given longer we would have implemented gestures to work on all of the screens, not just the summary screen and setup a webserver that would feed different locations to represent the food truck moving around.

Also the email intent was a bit of a hack since there is no food truck to send order to. If given more time I also would prefer to set up a webserver that will receive and display an order queue and provide mechanisms for the end user of that to “Clear orders” and send an update to the app that it’s ready.