Personalized Shopping Assistant

Team 4

Thomas Ripp Joseph Mirabile Jose Cruz

Stevens Institute of Technology

May 16, 2021

Contents

1 Mission Statement

For the people who are busy and enjoy purchasing online, which requires their products delivered to their doorstep as fast as possible, the shopping assistant is the fastest and most reliable platform, unlike Shopify or Amazon that takes multiple days to deliver their products.

PSA offers a 2-4 hour delivery window while offering a vast catalog of products, from your nearby stores, so you can buy from your favorite stores without leaving the house.

With PSA, customers are going to be able to make orders, which the drivers are going to either accept or deny. Then the drivers are going to go into the merchant's stores and purchase the products which are going to be delivered to the customers.

2 Key Stakeholders

2.1 Merchants

The major benefits that the merchants will receive from the application are to increase their exposure, while also saving cost because they wouldn't need to hire a delivery service nor there is a need to keep maintenance of the vehicles.

One of the major features is that the application will provide the merchant a delivery service, a bigger pool of customers while also enables insights on what customers are purchasing, giving them information on what items to promote.

The merchants that the application is going to target are small stores and boutiques, one of their constraints is that they are short staff so they can't have a dedicated person to handle the picking up the items, and we don't want to be intrusive in their workflow so the drivers will do the shopping and pick up the items like an everyday customer.

2.2 Customers

The major benefits for customers are that they will be able to find out merchants near their home from where they can purchase, since the merchants that get a display in the user's app, are close to the customers, the delivery time is fast.

Customers also avoid spending time going to the stores, doing the line, and purchasing the items, they only need to make the order and they get it delivered to their doorstep in less than 2 hours.

2.3 Developers

The developers belong to the same company as the requirement engineering team, for them, the main concern is clear, precise, and verifiable requirements.

2.4 Drivers

The main benefit for the drivers is to get another source of income by using their current means of transportation as an asset.

For them, one of the major features is going to display in the map the place where they need to do the shopping and the location where they need to deliver.

They also require to keep track of the milage and time of the order, and also keep track of all the orders that they delivered.

3 Key Drivers

3.1 Small Businesses

The reason that we are building this application is to help small businesses to compete with the big stores. Because of COVID, small businesses are now forced to migrate their operations online. Big companies like Amazon and Shopify were online since the beginning, Walmart, Target, and BestBuy have their main operations in store but they also had an online presence.

Since these companies already have their delivery service, they are now focusing on improving and refining their e-commerce experience. This puts small businesses in peril since they already work with small profit margins, to begin with, they lack the capital to hire the technical skills required to create an online presence.

3.2 Unemployment

Not only small businesses are affected, since the beginning of the pandemic unemployment has been rising, particularly in the US. That's why this application does not only help small businesses but also helps people who have their method of transportation and want to make extra money by fulfilling orders from the platform.

This application will help small businesses and people without jobs while also helping to keep social distancing during the pandemic, now people more than ever are ordering online, so this is a good opportunity for a new player to get into the market.

3.3 Health Risk

Since the beginning of the pandemic, stores are required to shut down or limit the number of people they can have inside the stores, which translates into fewer sales for the merchants. Going to the store also increases customer exposure to the virus.

With the application, we can reduce the number of people inside the stores while also helping the businesses to keep their sales.

3.4 Convenience

We want to enable our customers to have the same experience that they receive while ordering fast food. For this reason, we want to create an application that makes the purchase experience as seamless as possible.

4 Key Constraints

4.1 Business Constraints

CON-1 Drivers are required to have a background check before being applied to be enrolled in the platform.

CON-2 Businesses are required to provide their opening and closing hours.

CON-3 Drivers must have their vehicle, transportation insurance, and health insurance.

CON-4 Drivers must be over 18 years old.

CON-5 Merchants must have a business license.

4.2 Project Constraints

CON-6 Developers can only use open source libraries or tools that are either Apache or MIT license.

CON-7 Google Pay, Apple Pay, and PayPal are the only supported payment providers.

CON-8 The data interchange format must be JSON.

CON-9 The authentication/authorization mechanism between applications is going to be JWT.

CON-10 JWT tokens are going to expired after 24 hours.

CON-11 The database that is going to be used in the project must be MySQL.

CON-12 There is only one driver per session. *A driver can't log in on two devices at the same time.*

CON-13 PSSS will only accept orders from the merchant opening hour to 60 minutes before the merchant closing time.

CON-14 Merchants are only going to be displayed to customers if they are less than 2 miles from each outer.

CON-15 All applications must follow Google Material Design guidelines. [14].

CON-16 Images should take less than 10MB.

5 Key Requirements

5.1 Terms, definitions and abbreviated terms

5.1.1 Terms and Definitions

- **OAuth**: Authentication protocol that allows one application to interact with another one on behalf of the first one.
- **Firebase Cloud Messaging**: Cross-platform message and notification solution developed by Google.
- **Customer**: The user is going to search products and create orders by purchasing products from merchants.
- **Merchant**: The user upload products to the store, inside the platform, from which the customers are going to purchase.
- **Driver**: The user that fulfills the order generated by the customer.
- **Order**: A list of items that were paid by a customer and are going to be fulfilled by a driver.
- **Cart**: Container that holds, temporarily, a group of products that are going to be purchased.
- **Estimated time of Arrival**: The amount of time that an order is going to take before arriving at the customer.
- **Interaction**: Amount of touches required by the user to act, if the device has a touch screen or the number of clicks a user performs if the device has a non-touchscreen.
- **2FA**: Authentication method in which a particular device is used to grand access to a website or application, this method requires two or more pieces of evidence that confirms the user identity.
- **JWT**: An open standard method for representing claims securely between two parties.
- **PCI**: Set of security standards designed to ensure that all companies that accept, process, store or transmit credit card information maintain a secure environment. [1]

5.1.2 Abbreviated Terms

Abbreviation	**Term**
PSS	Personal Shopper System
PSMS	Personal Shopper Merchant System
PSDS	Personal Shopper Driver System
PSSS	Personal Shopper Server System
FCM	Firebase Cloud Messaging
ETA	Estimated time of Arrival
2FA	2-Factor Authentication
JWT	JSON Web Token
JSON	JavaScript Object Notation
PCI	Payment Card Industry
VM	Virtual Machine
AWS	Amazon Web Services
CI	Continuous Integration

5.2 Assumptions and Dependencies

AS-1 The market share that the application is going after, is the online shopping market where the customer and the merchants exist near each other in a 2 mile radius.

AS-2 We will use the HEART model [13] to measure improvements in UI/UX designs.

AS-3 Drivers already sign up in the system when they send their driver application to the company, outside of the project scope.

AS-4 Merchants are already added into the system, outside of the project scope.

AS-5 There is only one account per merchant. *At least in the first iteration.*

AS-6 The merchants are going to submit products that are appropriate for all ages. *They are not going to upload drugs, guns, medical devices.*

AS-7 The merchants already sign a contract that specifies the types of products they can add to the platform.

AS-8 Customers have 2FA [4] as an option.

AS-9 Merchants require 2FA.

AS-10 Drivers have 2FA as an option.

AS-11 A driver session, for a specific driver, can only run on one device at a time.

AS-12 PSS, PSMS, PSDS only talk with PSSS, using a request-response approach.

AS-13 Installing PSSS requires that a trained operator perform the installation.

AS-14 Android is the only mobile platform. iOS is going to be the target in future releases.

AS-15 The customer order would be sent to the nearest driver to the merchant store.

AS-16 A **disabled** is a product that does not get displayed in PSS.

AS-17 Opening and closing times for merchants are going to be set outside of the scope of the project.

AS-18 Merchants are small businesses that usually have no more than 5 employees in a single shift.

AS-19 All the merchants, customers and drivers operate inside the Hoboken area.

AS-20 PSS has two versions of mobile which is Android and a Web version.

AS-21 PSMS has only a web version.

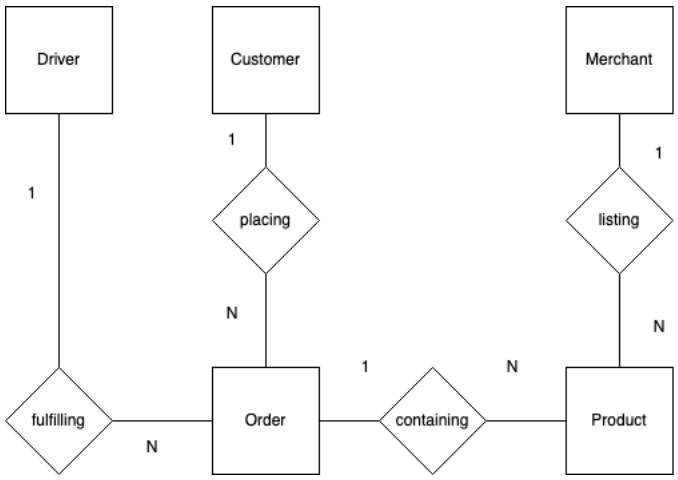

Figure 1: Logical Data Model

AS-22 PSDS has only an Android version.

AS-23 PSSS is an API server only.

AS-24 Notifications in PSS (Mobile/Web) and PSDS are going to use FCM.

5.3 Business/Mission Requirements

BR-1 We want to achieve at least 15 transactions a day, for each merchant, within the first 6 months since the launch date.

BR-2 Keep customer satisfaction as high as posible, a 9 out of a 10 scale within the first 6 month of release.

BR-3 Achieve the break-even point in 6 months.

BR-4 Increase transactions to at least 50 a day, by the merchant, within a year.

BR-5 Achieve positive cash flow on this product within a year.

BR-6 Increase market share in Hoboken to 2% within a year.

BR-7 Increase market share in Hoboken to 4% in the second year.

BR-8 Reduce delivery time by 5% in the first 6 months.

BR-9 Reduce delivery time by 10% in the first year.

BR-10 Comply with federal laws and regulations.

BR-11 Achieve product-market fit [11] within the first year from release.

BR-12 Become the application with the biggest amount of stores partnered with our services in Hoboken within the first year.

BR-13 Be the application with the most drivers in Hoboken within the first two years.

BR-14 Increase the number of stores affiliated with the application to 12 stores after 3 months from release.

BR-15 Increase the number of stores affiliated with the application to 30 stores after 6 months from release.

BR-16 Increase the number of stores affiliated with the application to 100 stores after 12 months from release.

BR-17 Increase the number of stores affiliated with the application to 150 stores after 18 months from release.

BR-18 Expand to nearby cities within the first three years since release.

BR-19 Increase the number of drivers enrollment to 50, within the first 6 months since release.

BR-20 Increase the number of drivers enrollment to 100, within the first 12 months since release.

BR-21 Increase the number of drivers enrollment to 150, within the first 24 months since release.

BR-22 Increase merchant sales by 20% within the first 24 months of launch.

BR-23 Achieve 30, 000 USD in gross profit at the end of the first year.

BR-24 Increase the number of employees by 2 in the first two years.

BR-25 Increase company gross margin at a rate of 10% from the previous year, each year.

BR-26 Improve UI/UX in our products between the first and second year since launch. *We will improve it by iterating periodically.*
Ref: AS-2

BR-27 Reduce monthly support costs by 10% after the first year. *This refers to the customer support provided to support inquiries coming from merchants, drivers, or customers of the platform, this customer support system is outside of the scope of the project.*

5.4 User Requirements

*The requirements that end with an asterisk *****, are requirements going to be developed in future releases and are not expanded in the system requirements.*

5.4.1 Merchants

USR-1 As a merchant, I want to authenticate in the application.

USR-2 As a merchant, I want to be able to update my password.

USR-3 As a merchant, I want to be able to log out.

USR-4 As a merchant, I want to add products into the application.

USR-5 As a merchant, I want to disable existing products.

USR-6 As a merchant, I want to update existing products.

USR-7 As a merchant, I want to remove existing products.

USR-8 As a merchant, I want to re-enable products that are disabled.

USR-9 As a merchant, I want to group products in categories.

USR-10 As a merchant, I want to be able to reset my password. *If the users forgets their password. Categories are a way to group products that share a particular characteristic, a product can only belong to one category at a time.*

USR-11 As a merchant, I want to set possible substitution for related items. *

USR-12 As a merchant, I want to group products in different type of collections. *Collection is a way to group products arbitrarily (Sales, Promotions, Seasons), a product can belong to one or more collections.*

USR-13 As a merchant, I want to be able to reach out for technical support.

5.4.2 Customers

USR-14 As a customer, I want to authenticate in the application.

USR-15 As a customer, I want to be able to update my password.

USR-16 As a customer, I want to be able to log out.

USR-17 As a customer, I want to see the available merchants near me.

USR-18 As a customer, I want to see the available products from a single merchant.

USR-19 As a customer, I want to see where the merchants are on a map.

USR-20 As a customer, I want to see where I'm on a map.

USR-21 As a customer, I want to be able to add my credit card information to my profile.

USR-22 As a customer, I want to upload my photo to my profile. *

USR-23 As a customer, I want to save my favorite merchants. *

USR-24 As a customer, I want to be able to see the information of the driver that is bringing the order.

USR-25 As a customer, I want to know what is the ETA for the order to arrive.

USR-26 As a customer, I want to know when the driver is on their way to the merchant store.

USR-27 As a customer, I want to know when the driver is working on the order.

USR-28 As a customer, I want know when the driver is on their way to deliver the order.

USR-29 As a customer, I want to able to rate the driver.

USR-30 As a customer, I want to be able to tip my driver. *

USR-31 As a customer, I want to be able to cancel an order.

USR-32 As a customer, I want to be able to set replacements in case a product is not found. *

USR-33 As a customer, I want to see all the products in a category, from a merchant page.

USR-34 As a customer, I want to see all the collections from a merchant.

USR-35 As a customer, I want to be able to message a driver.

USR-36 As a customer, I want to know the contact information of a particular merchant.

USR-37 As a customer, I want to be able to reach out for technical support.

USR-38 As a customer, I want to be able to search for a product on the platform. *

USR-39 As a customer, I want to be able to sort products by price.

USR-40 As a customer, I want to be able to sort merchants by distance.

USR-41 As a customer, I want to be able to sign up onto the platform.

USR-42 As a customer, I want to be able to add a product to my cart.

USR-43 As a customer, I want to create an order.

USR-44 As a customer, I want to be able to reset my password.

USR-45 As a customer, I want to mark an order as delivered.

USR-46 As a customer, I want to be able to report an issue with an order.

5.4.3 Drivers

USR-47 As a driver, I need to authenticate in the application.

USR-48 As a driver, I want to be able to update my password.

USR-49 As a driver, I want to be able to sign out.

USR-50 As a driver, I need to notify the system that i am ready to accept orders. *

USR-51 As a driver, I need to accept orders.

USR-52 As a driver, I need to deny orders.

USR-53 As a driver, I need to deliver orders to an address.

USR-54 As a driver, I need to notify the customer that I'm on my way to the merchant store.

USR-55 As a driver, I need to notify the customer that I'm working on the order.

USR-56 As a driver, I need to notify the customer that I'm on the way to deliver the order.

USR-57 As a driver, I need to be able to reach out to a customer in case an item is not available.

USR-58 As a driver, I need to see a history of all the orders I delivered.

USR-59 As a driver, I need to be able to take a picture, when the order is delivered.

USR-60 As a driver, I want to know if I am in an available area to receive orders. *

USR-61 As a driver, I want to know the route to deliver the customer order. *

USR-62 As a driver, I want to be able to see the items in the order I accepted.

USR-63 As a driver, I want to know the route to the merchant store from an order that i accepted. *

USR-64 As a driver, I want to be able to reach out for technical support.

USR-65 As a driver, I want to know how much profit I made from an order. *

USR-66 As a driver, I want to be able to reset my password.

USR-67 As a driver, I want to be mark an order as delivered.

USR-68 As a driver, I want to know where I'm located inside a map.

5.5 Operating Environment

OE-1 The PSSS is going to run on AWS infrastructure.

OE-2 Image assets are going to be stored in AWS S3.

OE-3 New servers are going to be spin up on-demand using EC2 VMS.

OE-4 The PSSS can only be accessed by PSS, PSMS, PSDS.

OE-5 Network requests coming from outside of the US are going to be ignored.

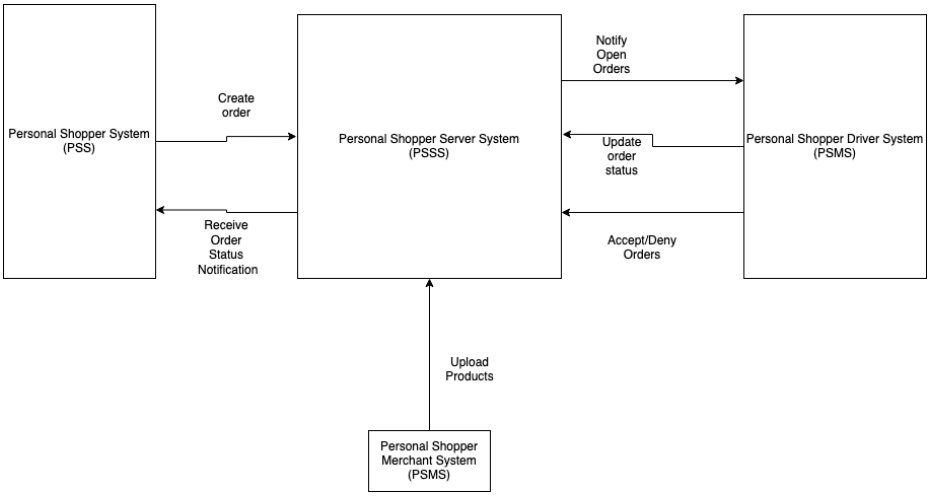

Figure 2: Context Diagram

5.6 System Requirements

5.6.1 General

System refers to all the user facing systems (PSS, PSMS, PSDS)

SY-1 The system shall enable users to login into the platform.
Ref: USR-1, USR-14, USR-47

SY-1.1 The system shall request the user to input the following required fields:

- Email
- Password

SY-1.2 The system shall display an error message if one of the fields to be entered is empty.

SY-1.3 The system shall display an error message if the format for one of the fields is incorrect.

SY-1.4 The system shall display an error message if the email and the password does not match the previously stored credential.

SY-2 The system shall enable users to log out.
Ref: USR-3, USR-16, USR-49

SY-3 The system shall enable users to update their passwords.
Ref: USR-2, USR-15, USR-48

SY-4 The system shall display a phone number to reach out to customer support.
Ref: USR-13, USR-37, USR-64

5.6.2 Personal Shopper System

SY-5 PSS shall enable a customer to sign up:
Ref: USR-41

SY-5.1 PSS shall request that customer fill their information by writing inside an input text requesting the following fields:

- Name
- Last name
- Phone number
- Email
- Password
- Confirm Password

SY-5.2 PSS shall display an error message if one of the fields is empty.

SY-5.3 PSS shall display an error message if the format is incorrect.

SY-5.4 PSS shall display an error message if the passwords do not match.

SY-6 PSS shall enable customers to sign up into the platform by using an authorized public OAuth implementation:
Ref: USR-41

- Google Sign-in [2]
- Sign in with Apple [3]

SY-7 PSS shall display a message that suggest the user to sign up in response to a failed login in case the email address that they insert does not exist in the platform.

SY-8 The PSS shall enable customers to login into the platform by using authorized public OAuth implementation:
Ref: USR-14

- Google Sign-in [2]
- Sign in with Apple [3]

SY-9 PSS shall display a list of all the available merchants.

SY-9.1 PSS shall use the device GPS to locate registered merchants that are in a 2 mile radius from the customer position.
Ref: USR-17

SY-9.2 If no merchants are found, it displays the message "No merchants are in your area".

SY-9.3 PSS shall display the merchants as cards. [5]

SY-9.4 PSS shall display merchant information:
Ref: USR-19, USR-36

- Name
- Address
- Image
- Phone
- Distance between the customer and the merchant, in meters.

SY-10 PSS shall display what is the customer current location inside a map.
Ref: USR-20

SY-11 When the customer is inside a merchant page, PSS shall enable the customer to search a product.
Ref: USR-18

SY-11.1 PSS shall show a list of products, that matches the search, and display the following fields.

- Product name
- Image
- Short description
- Price

SY-12 PSS shall enable customers to add a product to the cart.
Ref: USR-42

SY-13 PSS shall enable customers to add their payment information into the platform.
Ref: USR-21

SY-14 When the cart has one or more items, PSS shall permit the user to create an order.
Ref: USR-43

SY-15 PSS shall permit customers to cancel an order.
Ref: USR-31

SY-16 PSS shall permit customers to confirm that they received an order.
Ref: USR-45

SY-17 If the customer forgets its password, PSS shall request to user to enter their email address and shall send an email with a temporary password.
Ref: USR-44

SY-18 If there is an issue with an order, PSS shall enable customers to open an ticket about the order.
Ref: USR-46

SY-19 PSS shall enable user to send a message to the driver.
Ref: USR-35

SY-20 PSS shall display drivers information:
Ref: USR-28

- Phone Number
- Name
- Profile Picture

SY-21 PSS shall display the ETA from a pending order.
Ref: USR-25

SY-21.1 PSS shall display the ETA in terms of minutes.

SY-22 PSS shall display merchants information.
Ref: USR-19, USR-36

- Name
- Phone
- Email
- Address

SY-23 PSS shall enable user to sort products by price.
Ref: USR-39

SY-24 PSS shall enable user to sort merchants by distance.
Ref: USR-40

5.6.3 Personal Shopper Merchant

SY-25 PSMS shall enable merchants to add a product, by requesting the following required fields.
Ref: USR-4

- Name
- Description
- Picture
- Weight
- Price
- Tags

SY-26 PSMS shall enable merchants to update a selected product.
Ref: USR-6

SY-27 PSMS shall enable merchants to delete a selected product.
Ref: USR-7

SY-24.1 PSMS shall display a message dialog that requests a confirmation from the merchant.

SY-28 PSMS shall enable merchants to disable a selected product.
Ref: USR-5

SY-29 PSMS shall enable merchants to group products into categories.
Ref USR-9

SY-30 PSMS shall enable merchants to group products into collections.
Ref USR-12

SY-31 PSMS shall enable merchant to enable products that were previously disabled.
Ref USR-8

SY-32 If the merchant forgets its password, PSMS shall request to user to enter its email address and PSMS shall send an email with a temporary password.
Ref USR-10

5.6.4 Personal Shopper Driver

SY-33 If the driver forgets its password, PSDS shall request to user to enter its email address and shall send an email with a temporary password.
Ref USR-66

SY-34 If the driver receives an order requests, PSDS shall request the driver to accept the order.
Ref USR-51

SY-35 If the driver receives an order requests, PSDS shall request the driver to decline the order.
Ref USR-52

SY-36 If the driver fulfills the order, PSDS shall enable the driver to mark the order as delivered.
Ref USR-67

SY-37 If the driver mark an order as delivered, PSDS shall request the driver to take a picture of the delivery.
Ref USR-59

SY-38 PSDS shall enable the driver to send a signal to PSSS that notifies the customer that the driver it's on its way to the merchant.
Ref USR-54

SY-39 PSDS shall enable the driver to send a signal to PSSS that notifies the customer that the driver it's at the merchant and it's working in the order.
Ref USR-55

SY-40 PSDS shall enable the driver to send a signal to PSSS that notifies the customer that the driver it's on its way to the deliver the order to the customer.
Ref USR-56

SY-41 If the driver requires to reach out to the customer, PSDS shall display the phone number of the customer.
Ref USR-57

SY-42 PSDS shall display a list of all the orders that a driver delivered.
Ref USR-58

SY-43 PSDS shall display what is the driver's current location inside a map.
Ref USR-68

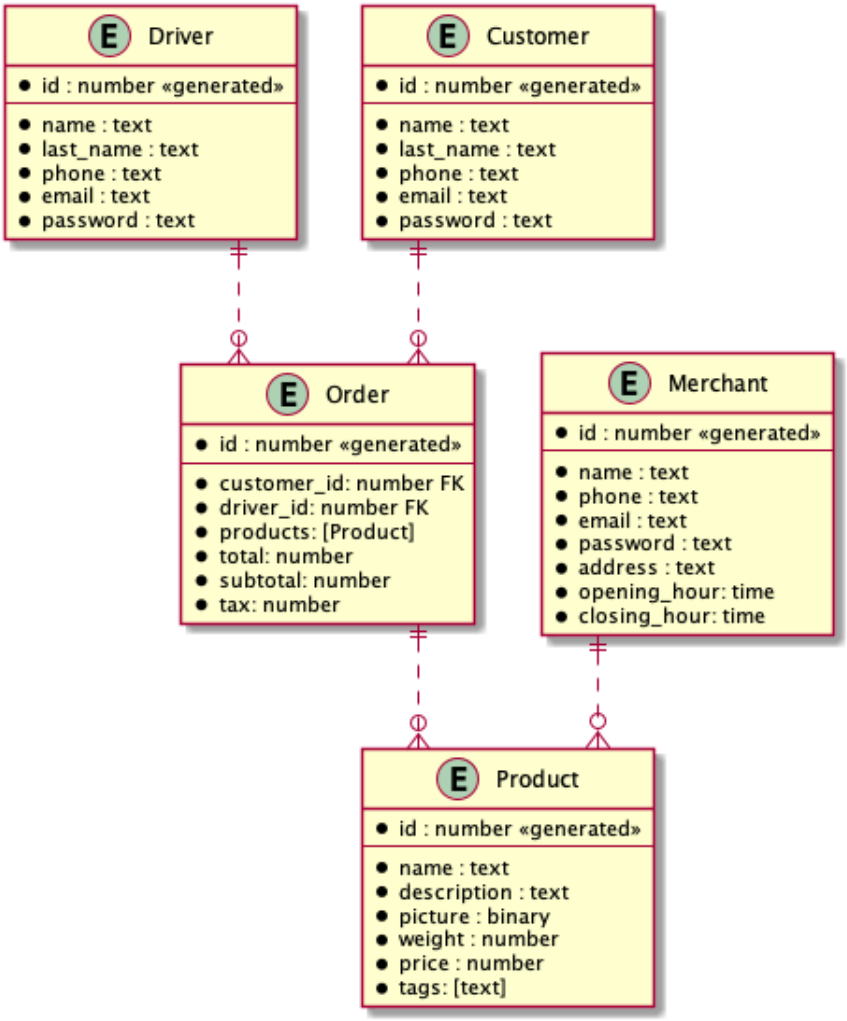

Figure 3: Data Model Diagram

5.7 Security Requirements

SEC-1 The PSS shall require an email and password access to log in.

SEC-2 PSS password must be no shorter than 8 characters, it requires at least 1 special character, and at least 2 numbers. *Special character are any character under UTF-8, that is not an alphanumeric character.*

SEC-3 The PSS shall not store passwords in plain text, it shall salt the passwords and encrypt them using SHA-2 algorithm.

SEC-4 The PSS shall display an error message in the case of a failure in a transaction in order to protect customer data.

SEC-5 The PSS shall allow customers to file any reports with their orders by sending an email to the support email address with the order number and issue details.

SEC-6 The PSS shall ask the customer whether or not their order has arrived if it is 1 hour after the driver gets to the store.

SEC-7 The PSS shall allow users to delete their own account, this is also shared with the PSMS and the PSDS.

SEC-8 The PSS shall allow users to disable their own accounts for a certain period of time up to 1 year.

- After 1 year, since the account is disabled, PSS will delete the account.

SEC-9 The PSS shall not store any credit card information, it will use an authorized, PCI Compliant, provider to store the information. *PayPal*

SEC-10 The PSS shall ignore all network requests that come outside of the US.

SEC-11 The PSSS shall log all information regarding the server status into a log file.

SEC-11.1 PSSS shall log the utilized RAM every 5 minutes.

SEC-11.2 PSSS shall log the utilized CPU every 5 minutes.

SEC-11.3 PSSS shall log all search queries.

SEC-12 PSSS shall considered that a request is coming from a secure environment if the communication protocol is HTTPS and a valid JWT [7] is sent inside the **Authorization** header.

SEC-13 A security audit shall be performed in all the dependencies that are add it to the project.

SEC-14 A system security audit should be performed each year.

SEC-15 The PSDS shall have a 2 factor authentication procedure as an option.

SEC-16 PSMS requires 2FA procedure for authentication.

SEC-17 The PSDS shall only allow the driver to log in from one device at a time.

5.8 Quality Requirements

5.8.1 Availability

AVL-1 The PSSS shall be available at least 90% during the night on regular days (days that are not holidays), from 01:00AM to 8:59 AM and from 7:01 PM to 11:59 PM.

AVL-2 The PSSS shall be available at least 99.9% during holidays, from 01:00 AM to 23:59 PM.

AVL-3 The PSSS shall be available at least 99% during the day on regular days (days that are not holidays), from 9:00 AM to 7:00 PM.

5.8.2 Installability

INS-1 The PSS (Mobile) should be installed from Google Play Store.

INS-2 The PSDS (Mobile) should be installed from Google Play Store.

5.8.3 Interoperability

IOP-1 HTTPS be the only data access protocol supported by the PSSS.

5.8.4 Performance

	PSS	**PSDS**
Loading view	Less than 3 seconds	Less than 2 seconds
Search product	Less than 2 seconds	-
Customer pay for the order	Less than 5 seconds	-
Log In	Less than 5 seconds	Less than 4 seconds
Log Out	Less than 3 seconds	Less than 2 seconds
Accept an Order	-	Less than 3 seconds
Decline an Order	-	Less than 3 seconds

5.8.5 Reliability

REL-1 No more than 5 orders out of 1,000 can be lost due to software errors.

REL-2 The PSSS shall not be down for more than 3600 consecutive seconds. *1 hour.*

5.8.6 Robustness

For the context of PSDS, PSMS, PSS offline means: *The software is running but it loses internet connection with the application server.*

ROB-1 If PSDS goes offline, it will try to reconnect to PSSS in intervals of 10 seconds.

ROB-2 If a message is sent from PSDS to PSSS and PSDS is offline, PSDS is going to store the message in a queue and send the message when it reconnects.

ROB-3 If a message is sent from PSS to PSSS and PSS is offline, PSS is going to store the message in a queue and send the message when it reconnects.

ROB-4 PSDS, PSS, PSMS shall have empty input fields by default and display an error message in case a required input field is empty.

5.8.7 Usability

Supported OAuth accounts:

- Google Sign-In
- Sign in with Apple

For the context of usability a workflow is: *Any combination of views, buttons, actions and inputs that helps a user to achieve a goal.*

USE-1 A first time user of PSS shall be able to create an order, with a cart already filled, in no longer than 3 seconds.

USE-2 A first time user of PSS shall be able to sign up in no longer than 2 minutes.

USE-3 A first time user of PSS shall be able to sign up in no longer than 20 seconds, if the user has a supported OAuth account.

USE-4 A first time user of PSS shall be able to sign in no longer than 5 seconds.

USE-5 A first time user of PSS shall be able to sign in no longer than 3 seconds, if the user has a supported OAuth account.

USE-6 A first time user of PSS shall be able to search for an item and add it to the cart in no longer than 20 seconds.

USE-7 A first time user of PSDS shall be able to sign in no longer than 5 seconds.

USE-8 A first time user of PSDS shall be able to accept an order in no longer than 10 seconds.

USE-9 A first time user of PSDS shall be able to deny an order in no longer than 10 seconds.

USE-10 A first time user of PSDS shall be able to log out in no longer than 5 seconds.

USE-11 A first time user of PSDS shall be able to mark that it's on its way to the merchant store, in no longer than 3 seconds.

USE-12 A first time user of PSDS shall be able to mark that it's on its way to the customer, in no longer than 3 seconds.

USE-13 A first time user of PSMS shall be able to add a new product in no longer than 2 minutes.

USE-14 A first time user of PSMS shall be able to remove a product in no longer than 15 seconds.

USE-15 A first time user of PSMS shall be able to update a product in no longer than 1 minute.

USE-16 A first time user of PSMS shall be able to login in no longer than 5 seconds.

USE-17 A first time user of PSMS shall be able to log out in no longer than 5 seconds.

USE-18 A single workflow would be consider user-friendly as long as the user makes 4 mistakes or less out of 100 times repeating the same workflow. *A mistake is any action that the user performs that he didn't intent to (E.g. Button miss clicks, invalid characters on input fields, wrong text format).*

USE-19 For the systems PSMS, PSS, PSDS, 98% of first time users should be able to use the applications without needing any help.

USE-20 A user of PSS should be able to create an order from an already filled cart in no more than 4 interactions.

USE-21 A user of PSS should be able to create an order from an empty cart in no more than 8 interactions. For an order of a single item.

USE-22 A user of PSDS should be able to complete a delivery in no more than 2 interactions.

USE-23 A user of PSDS should be able to signal a customer that it's on its way to the store in no more than 2 interactions.

USE-24 A user of PSDS should be able to signal a customer that it's on its way to the complete the order in no more than 2 interactions.

USE-25 A user of PSMS should be able to add a product in no more than 2 interactions.

USE-26 A user of PSMS should be able to update a product in no more than 2 interactions.

USE-27 A user of PSS should be able to edit their own profile in no more than 2 interactions.

5.8.8 Efficiency

EFF-1 The PSSS should used less than 70% of the available CPU under regular conditions. *Regular Condition: 90% of uptime*

EFF-2 The PSSS should used less than 500MB of RAM under regular conditions.

5.8.9 Modifiability

MOD-1 Classes must follow the SOLID principles. [6]

MOD-2 Functions calls required at least 80% of testing code coverage.

MOD-3 Public functions must be above protected functions.

MOD-4 Protected functions must be above private functions.

MOD-5 All merges to master must come from the CI server.

MOD-6 All test must pass before pushing changes to the remote repository.

MOD-7 All changes push to a release branch must be peer-reviewed.

MOD-8 Public and protected functions must include a comment that explains the functionality.

MOD-9 Variable must be descriptive in nature, abbreviations must be avoided.

MOD-10 Constants must be all uppercase, if the constant have more than one word they would be separated by an underscore.

MOD-11 Functions and variables must begin with a lower case.

MOD-12 Classes will be capitalize.

MOD-13 Variables and functions must be written in camel case.

MOD-14 Composition must be preferred over inheritance.

MOD-15 Lines would not exceed more than 80 characters, including white spaces.

MOD-16 Code format should be applied before committing.

5.8.10 Portability

POR-1 Modifying the Android version to the latest version shall require changing no more than 5% of the source code.

POR-2 PSMS, PSS (Web) shall support:

- Chrome 89, 88, 87
- Latest version of Safari
- Firefox 87, 86, 85
- Microsoft Edge 89, 88, 87

POR-3 PSDS, PSS (Mobile)

- Android 12, 11, 10

5.8.11 Reusability

REU-1 Web components must be reused between PSS (Web) and PSMS.

REU-2 Authentication mechanism must be shared between PSS (Mobile) and PSDS

REU-3 At least 45% of application architecture shall be reused between PSS (Mobile) and PSDS.

REU-4 2FA (2-Factor Authentication) mechanism shall be shared across all the systems.

REU-5 JWT token signing and token rotation mechanism shall be shared across all the systems.

5.8.12 Scalability

SCA-1 The system should scale up the system horizontally, if the system gets 95% of CPU or RAM utilized, which ever comes first. The system will spin up 2 servers with the same operating system and environment but with 80% less memory than the original server.

SCA-2 The system will scale down to 1 (original server) if one of the previous spin up servers, has less than 95% of CPU usage during 3600 consecutive seconds.

5.8.13 Verifiability

VER-1 The staging environment configuration shall be identical to the production configuration environment to avoid irreproducible testing failures.

VER-2 A tester shall be able to configure if the application is running in testing, develop or production mode.

6 Elicitation

6.1 Context

For the Personalized Shopping Assistant project, we set up meetings every other week with our customers, because of COVID we had to have this meeting in a remote setting using Google Meet.

6.2 Methods

6.2.1 Brainstorming

For the first meeting, our goal as a team was to figure out what is the main goal of the project, identify the stakeholders and get an idea of the scope of the project. For this project, we use user personas [8] to identify the stakeholders, because the customer does not have end-users at hand so we have to create imaginary roles as reference.

Next, we wanted to understand the customer's background, so we know how to translate ideas using the customer's language as a reference. This background helps us to create analogies that the client can understand.

To achieve this we set up the meeting in a way that each of the people involved in the project introduce themselves and tell us what kind of experience they have.

Once we establish this, we have a baseline of the technical vocabulary that we can use during the project.

We noticed the following by observing the group dynamic:

- Samantha and Timothy focus mainly on the business side, they are the ones that guide the conversation and have a better understanding on how the application should behave if a particular edge case appears.
- Alex focuses more on user privacy, security, and technical implementation.
- Kristin more towards user experience, on how to make the application more approachable towards the users.

We discover this by enabling them to talk freely about what is the the goal of the project, and each of them tends to participate when a particular subject is discussed.

6.2.2 Interviews

From the second meeting onwards, we take the input from the previous meeting and we create a set of questions that we send to the client 2-3 days before the next meeting so they have time to discuss.

We use this to get some clarification about requirements that were too vague or to confirm how a particular edge case should be handled.

These questions help us to stay on the scope during the meeting, once we are in the meeting, we review the questions one by one and ask more information for clarification, and depending on the use case we move the conversations towards how to validate them or what quality attributes they would expect from the use case.

If the client does not have the answer to a particular question we skip that question and go to the next one. Once we finish with the questions, we backtrack to the questions that they did not have an answer and we start brainstorming about possible solutions.

While we are doing the brainstorming session we validate right away, how can we verify those requirements and also we evaluate if those are feasible or not, sometimes in the conversation, some edge cases arise and we think about possible solutions for those edge cases.

When the meeting finalizes, we take notes of all the answers and start analyzing them. We normally take notes of our meetings with our customers and make changes accordingly, note by note, from what we hear in response to them.

7 Validation

At this point, we have a list of all requirements that the client would like to see in the project. We do a review of the list and we start discarding the requirements that are not feasible, because of technical, geographical, or monetary reasons.

Since this project is a CRUD application [10] and do not require anything special from the hardware besides GPS and internet connection, most of the requirements are feasible, except for some of the performance requirements, where the client requested 15ms on page load.

Latency is an issue in distributed systems, making the requirements infeasible because of physics laws.

Next, we ask them to put each system and requirement into the following categories:

- **MVP** [9]
- **Future Release**

For the **MVP** we ask them which features and systems are required for their release date, we ask them to be as objective as possible. Anything that is not in the **MVP** is going to **Future Release**.

We focus the conversation on the MVP, now that we know which features and systems are important we need to prioritize them, we send the client a document with all the features, grouped by system, and ask them to rank them between 1 and 5.. *1 being the highest priority.*

We use **dialog maps** to validate the workflow for each of the systems, the dialog maps are available in the Appendix.

- For the **Personal Shopper System** we used the Figure 4.
- For the **Personal Shopper Driver System** we used the Figure 5.
- For the **Personal Shopper Merchant System** we used the Figure 6.

In our last meeting, we review each of the requirements, one by one and they assure us that we capture all the requirements that they expected.

We also display them a 5 minute video where they approved that we capture their requirements.

References

[1] 2020, April 16, PCI FAQs. https://www.pcicomplianceguide.org

[2] Google Developers, 2021, April 16, Integrating Google Sign-In into your web app https://developers.google.com/identity/sign-in/web/sign-in

[3] Apple, 2020, September 16, What is Sign in with Apple? https://support.apple.com/en-us/HT210318

[4] Fruhlinger, Josh, 2019, September 10. 2fa explained: How to enable it and how it works https://material.io/components/cards

[5] Google, 2021, April 16 Cards https://www.csoonline.com/article/3239144/2fa-explained-how-to-enable-it-and-how-it-works.html

[6] Samuel Oloruntoba, 2020, December 17, SOLID: The First 5 Principles of Object Oriented Design. https://www.digitalocean.com/community/conceptual_articles/s-o-l-i-d-the-first-five-principles-of-object-oriented-design

[7] Auth0, 2020, April 16, Introduction to JSON Web Tokens. https://jwt.io/introduction

[8] Patrick Faller 2019, Dec 17, Putting Personas to Work in UX Design: What They Are and Why They're Important https://xd.adobe.com/ideas/process/user-research/putting-personas-to-work-in-ux-design/

[9] ProductPlan 2021, Aprl 30, Minimum Viable Product (MVP) https://www.productplan.com/glossary/minimum-viable-product/

[10] Brenden Thornton 2019, Aug 30, What's a CRUD App? https://medium.com/@thorntonbrenden/whats-a-crud-app-e5a29cce03b5

[11] Product Plan 2021, May 12, What is Product-Market Fit? https://www.productplan.com/glossary/product-market-fit/

[12] Andrew Bloomenthal 2021, Feb 5, Gross Margin Definition https://www.investopedia.com/terms/g/grossmargin.asp

[13] Onix-Systems 2019, Jan 8, Matters of the HEART: How to Measure UI and UX Design https://onix-systems.medium.com/matters-of-the-heart-how-to-measure-ui-and-ux-design-b7f29b77a711

[14] Google 2021, May 14, Guidelines - Material Design https://material.io/design/guidelines-overview

8 Appendix

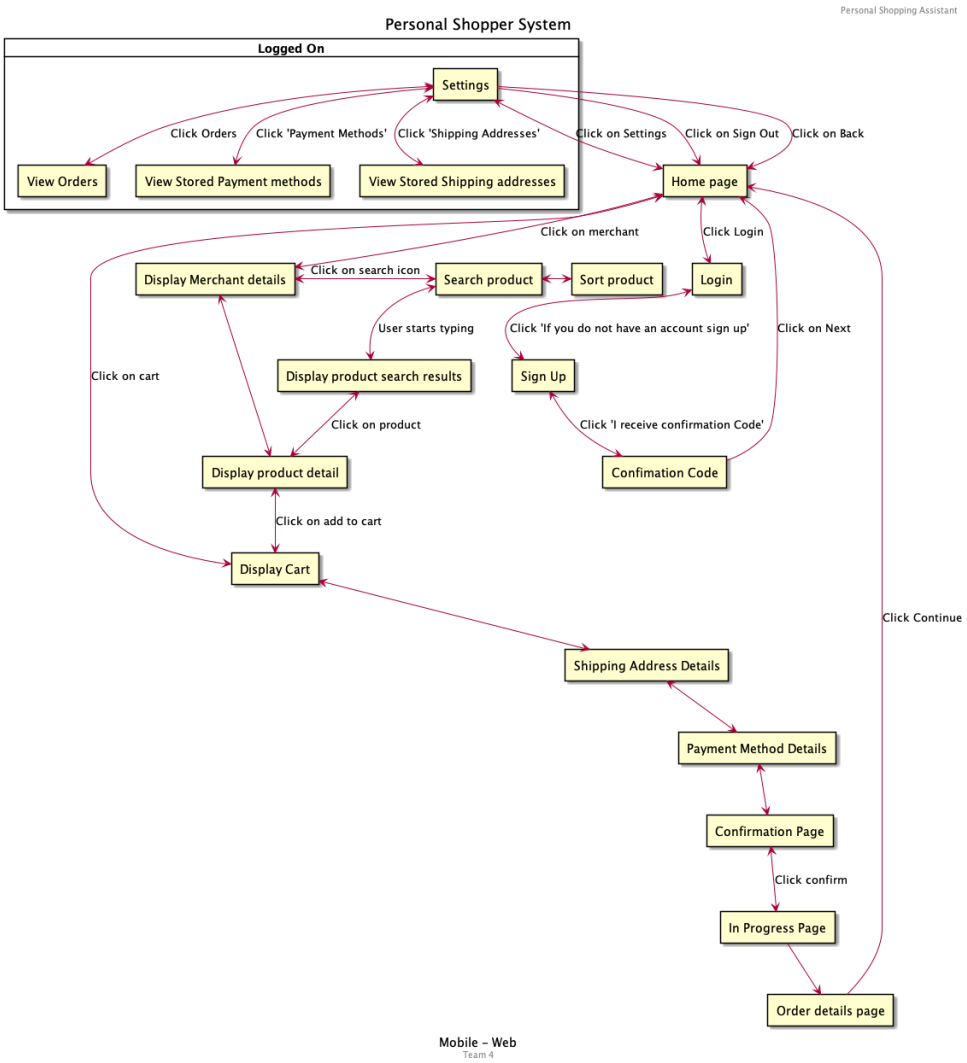

Figure 4: PSS Dialog Map

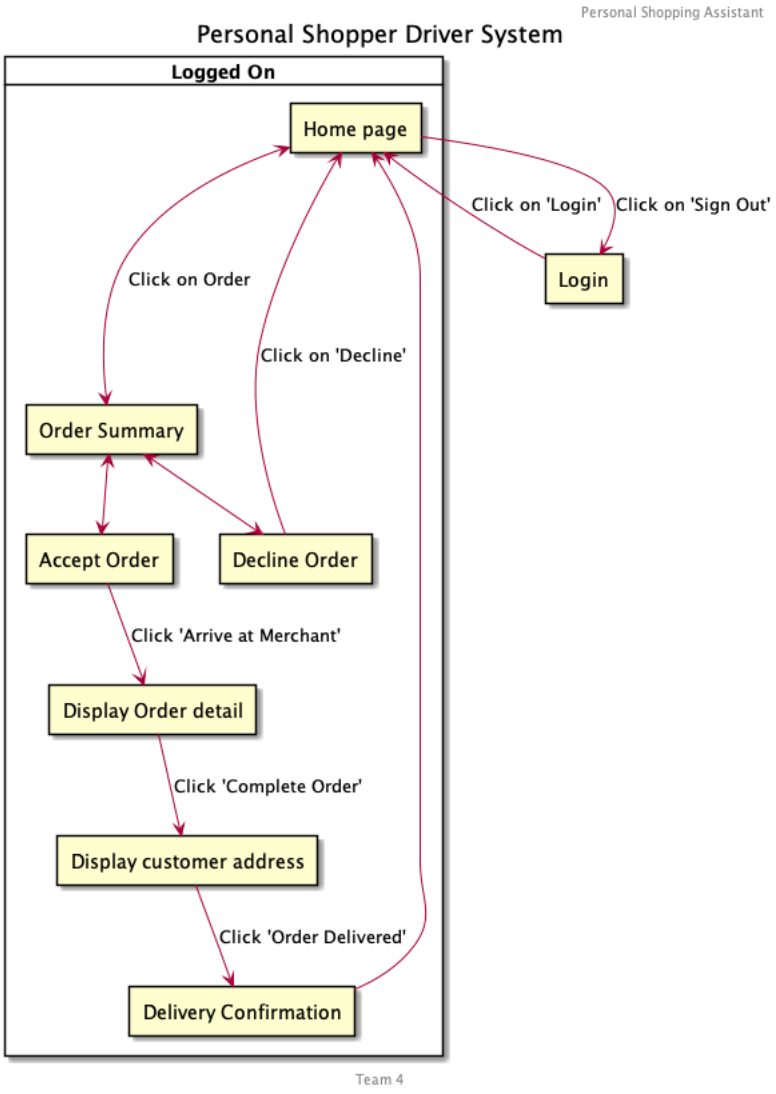

Figure 5: PSDS Dialog Map

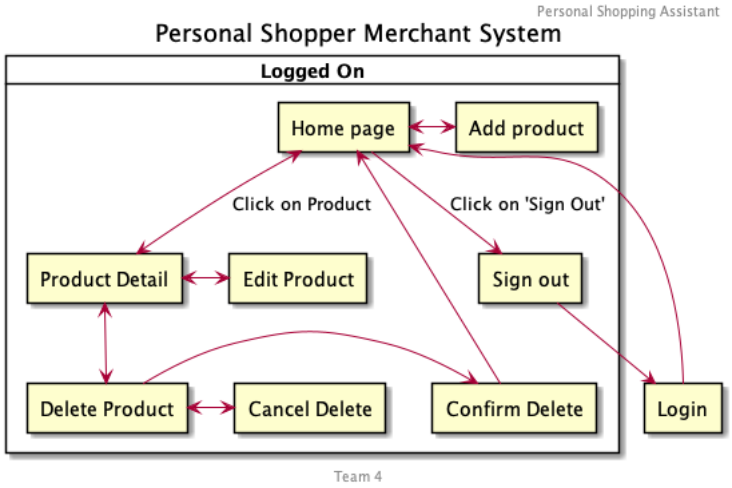

Figure 6: PSMS Dialog Map

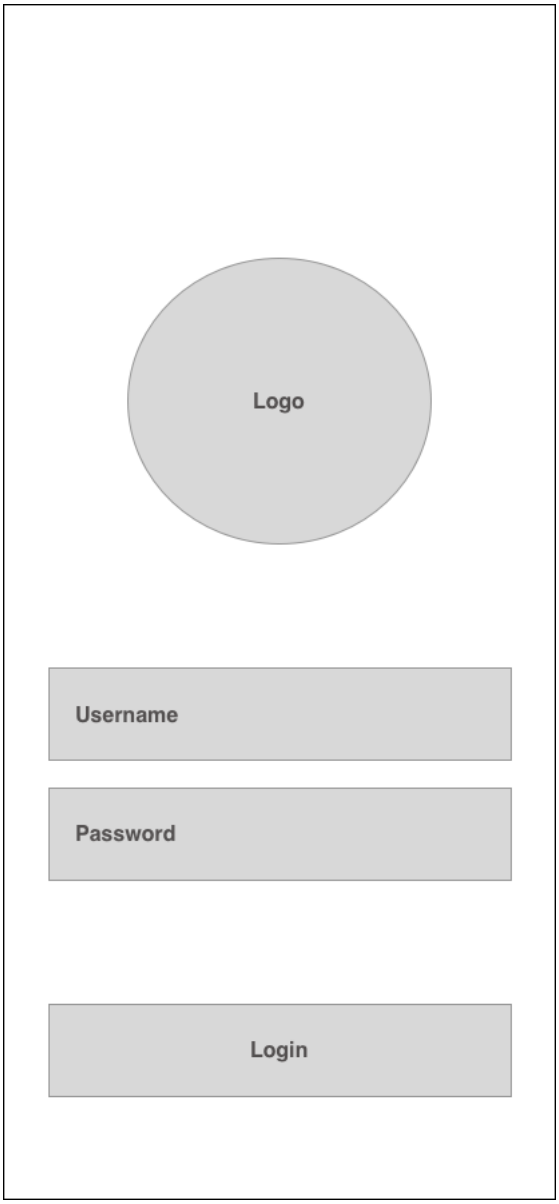

Figure 7: Prototype: PSS - Login

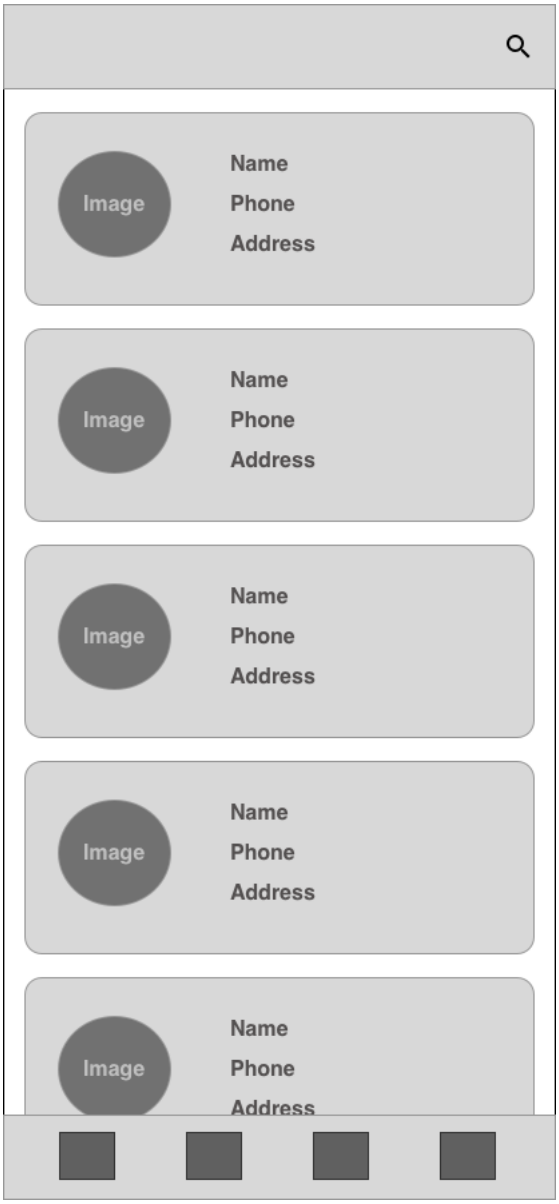

Figure 8: Prototype: PSS - Home

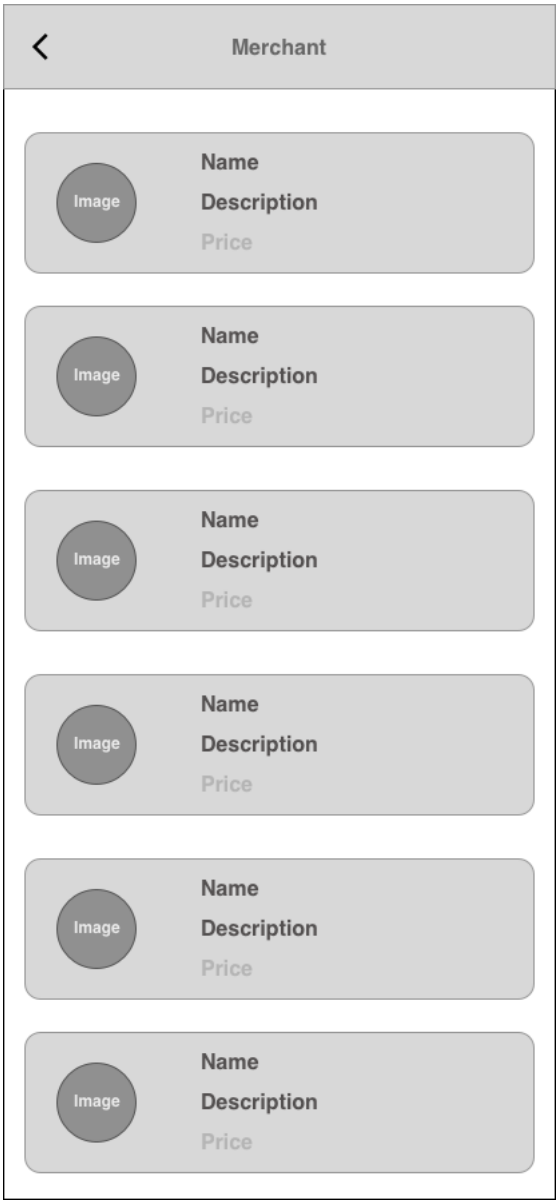

Figure 9: Prototype: PSS - Merchant Detail

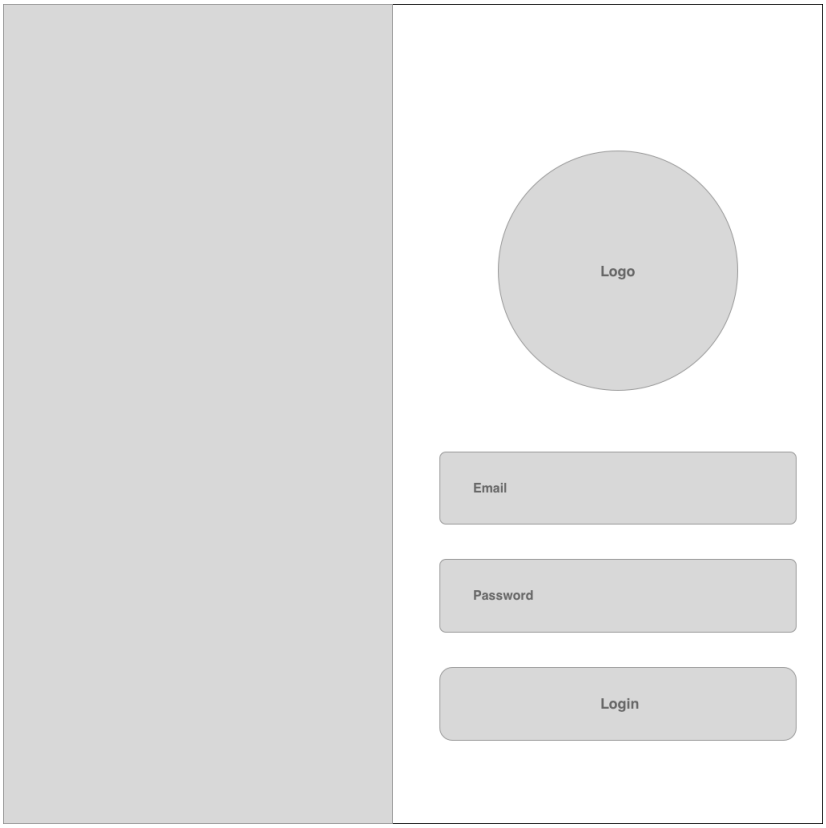

Figure 10: Prototype: PSMS - Login

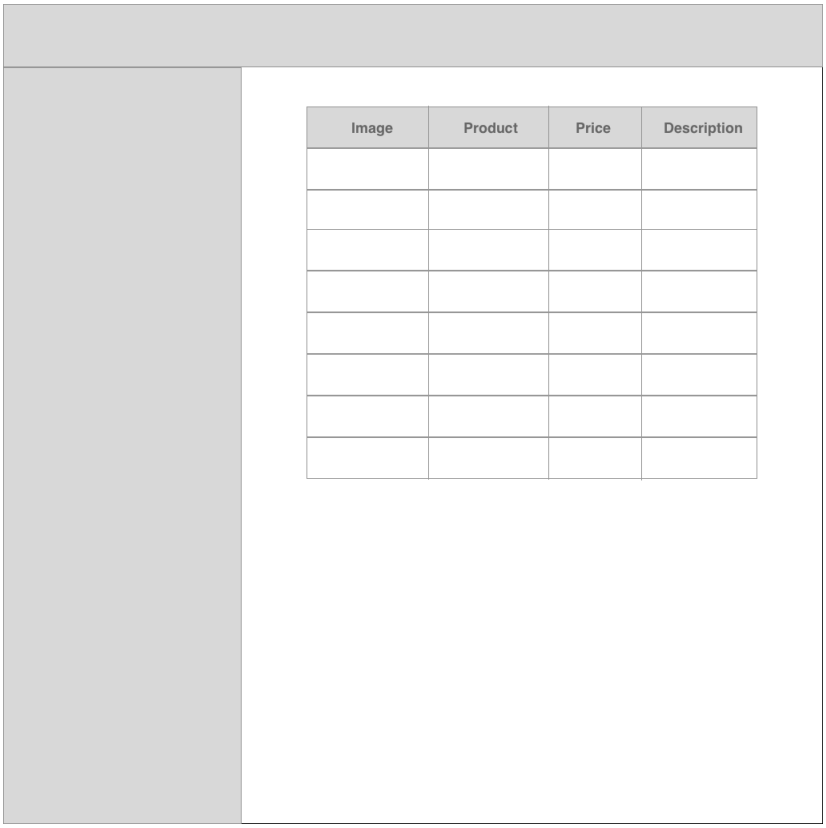

Figure 11: Prototype: PSMS - home

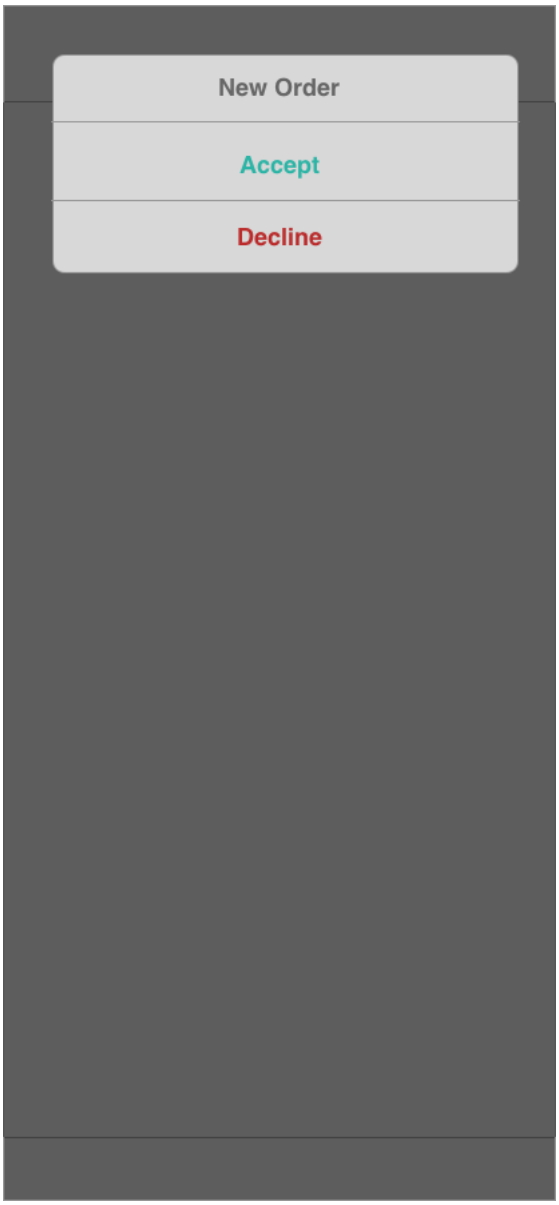

Figure 12: Prototype: PSDS - Accept/Decline Order

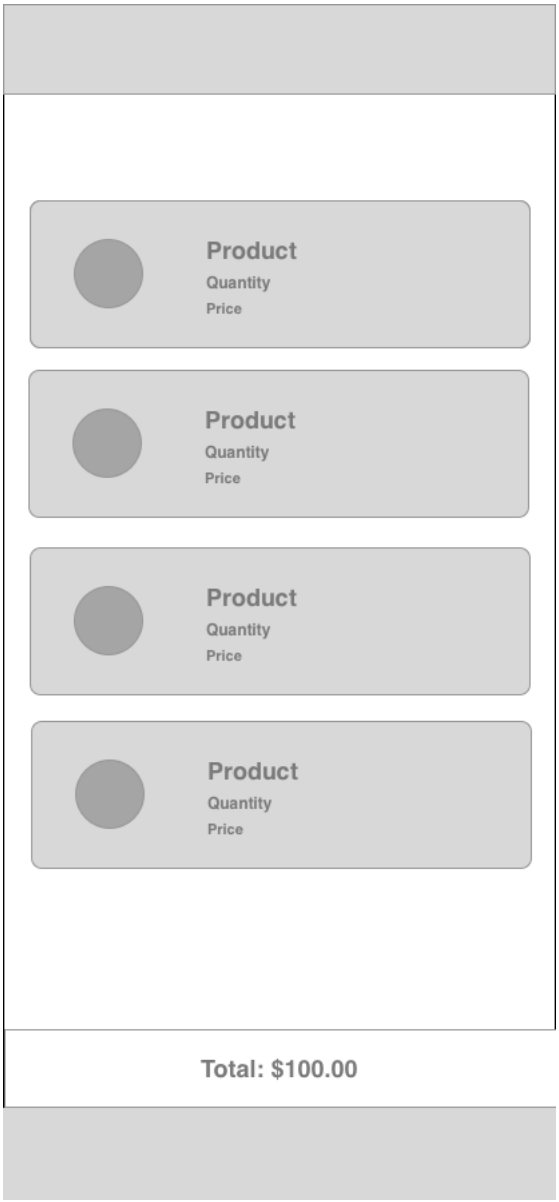

Figure 13: Prototype: PSDS - Order Detail