# Scope

On March 11th 2020, the COVID-19 virus made a devastating impact on a global scale. Overnight, the virus became a massive concern on an international scale prompting all countries to close their borders and implement a strict quarantine in order to inhibit the spread of this deadly disease. As more aspects of each community shut down, the general population was faced with a set of problems unique to this era; namely the concerns of living under strict and isolated conditions.

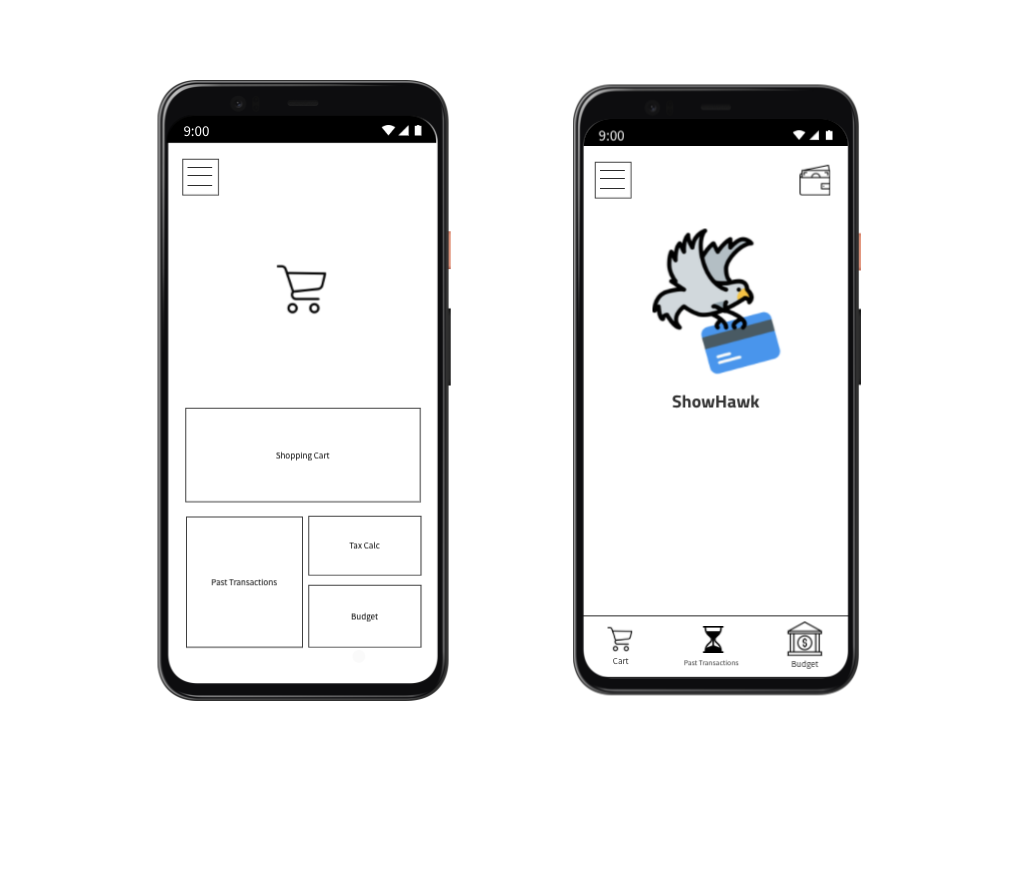
# The Problem

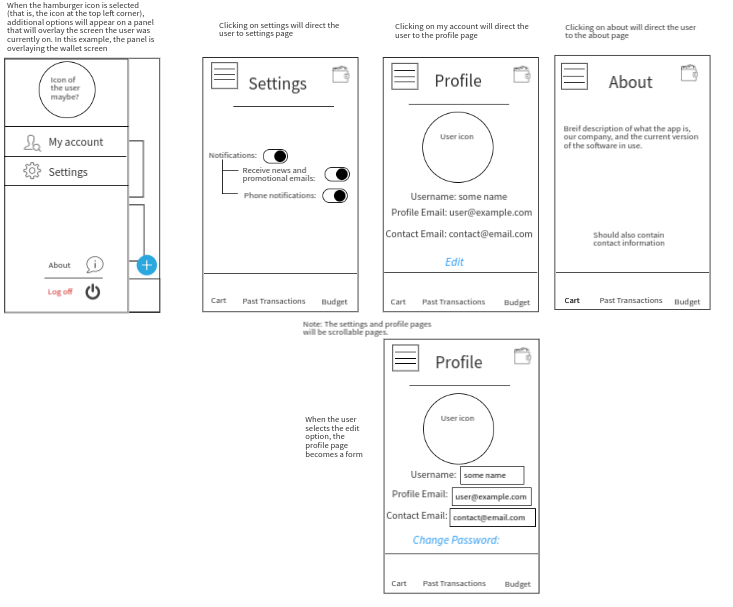
In an effort to curtail the virus, the Canadian Federal government imposed a lockdown on non-essential services, and implemented social distancing orders for the entire country. This caused certain businesses to impose limits on the number of customers allowed in their physical stores, greater protective measures for staff including better PPE (Personal Protective Equipment), and some businesses stopped accepting physical cash altogether.

However, the handling of credit cards and cardholder apps still require physical contact with a terminal (Inserting card/keying in a pin/tapping) as well as the handling of paper reciepts (after a transaction has completed). For this reason, Team JauntyPython has decided to create a new application in order to implement a fully contactless method of payment. This will therefore decrease the amount of contact required for any given purchase in a physical store. In addition the fully digital nature of this new point of sale system will provide a stepping stone to expand in new creative and innovative directions.

# Introducing ShopHawk

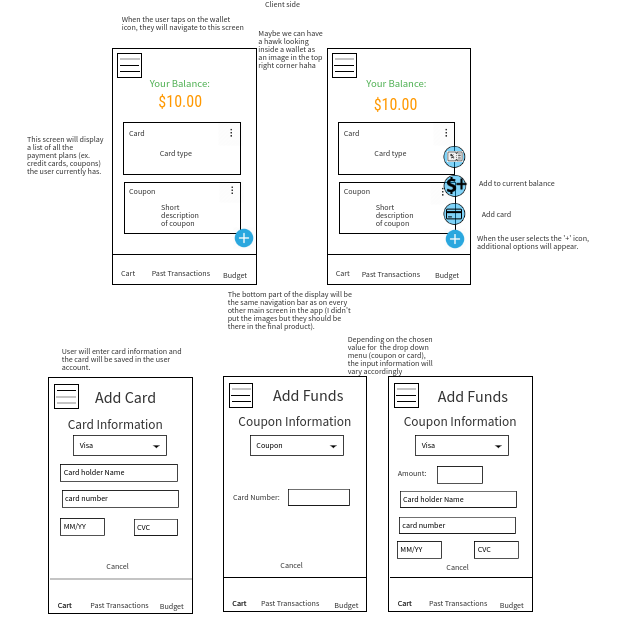
ShopHawk is Team JauntyPython’s fully contactless solution to shopping during a global pandemic. It will not only be a source for contactless payment, but a digital receipt keeper, and budgeting companion. On the store-side application, ShopHawk can be a helpful tool in inventory management, customer service, and accounting.





## Basic Operations (Use cases)

The following will be a brief introduction to each of the app’s functionalities. This will describe how it benefits a specific stakeholder in the business to help them achieve their goals. Note that the clients may also make use of these features.



#### **Transaction (Sales system)**

Transactions between customer and store take place between a server and two

Terminals. The customer’s terminal (most likely smartphone) will be scanned by the Store’s terminal (likely a desktop computer) which will identify the Unique ID of the Customer and start the Transaction process.

Stakeholder: Sales Representative

1. Cashier scans/inputs customer ID. Alternatively, they could also input some other information stored on a user’s account such as an email address.

Loop

1. Enter item via scanning the item or searching for the item by name.
2. Prompt for another item

End loop (if all items have been entered)

1. Cashier verifies the list of entered items.
2. Cashier presses a button to send the pending E-receipt to the Customer
3. Customer looks over the receipt and chooses to pay or reject the order. Customers will be notified of their remaining budget if they were to proceed. (The notification itself will vary depending on the budget and how much is left)
4. If the customer accepts, the order is paid through the Customer’s choice of E-Wallet or Credit Card (which was chosen when they have accepted the order).
5. Application files away the E-receipts of both Customer and Store for later review.

#### **Receipt Sorting (Accounting system)**

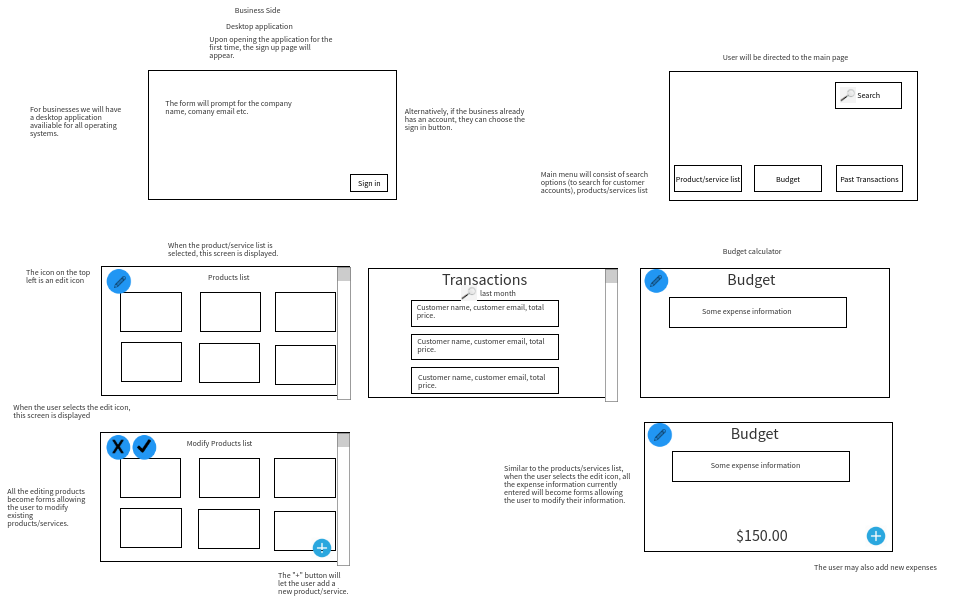
The Receipts generated by the transaction process are stored digitally on a

Server so as not to crowd the physical devices with data. The digital nature of the receipts will also aid in sorting receipts by category which is useful for tax purposes.

Stakeholder: Finance

1. User selects the history icon on the bottom of the screen (the navigation bar)
2. They may select an option to filter receipts by date
3. Optionally they may do a more fine-grained search by filtering by specific categories (that is, area of the business)
4. User confirms their settings for the filter
5. they will be presented with lists of receipts grouped by their category along with the running total of the receipts at the bottom of the list.
6. This report may be saved for future reference

#### **Budgeting (Accounting system)**

****

Managing funds and budgets is difficult if the amount of money spent and allotted for spending is unavailable. Therefore ShopHawk will allow the user to set and implement visual representations of their spending budget. Upon a purchase, the user will be notified of their updated budget allowing them to keep track of their finances with ease. While there are many financial management apps for personal use, we believe that adding this functionality to an online payment application will be a large draw to the platform.

Stakeholder: Finance

1. User selects the budget icon (‘$’ icon) on the bottom navigation bar.
2. User is now presented with the current budget information

Loop

1. Users may add a new budget with the “+” icon.
2. The new budget may be given a specific category to watch for finances in a specific area.
3. Users can set their new budget as well as a reset time (that is, when the budget should be replenished).
4. Changes will apply if the “Save” button is selected.
5. If the user would like another budget for another category, They may add (continue looping).

End loop

1. If necessary, users can optionally modify the budget by selecting the “Set Budget” button.
2. Selecting the option will allow the user to edit any detail of the budget information they would like to be modified.
3. User confirms modified changes by selecting the “Save” button.

#### 

#### **Rewards System (Marketing system)**

While enticing the customer to use the platform is important, there should also be some functionalities for the Business implementing them. The Rewards System functionality allows the Business to create a Points based or stamp based Rewards system for their business. As all the customer information is already on their hand held terminal, there should be very little left to do other than to press a button to opt into the Business’ reward program. All rewards stamps and points will be stored on the app, therefore removing the need to print or present a physical card. Adding points to the card is also automatic upon purchase.

Redemption could be set to automatically trigger upon the acquisition of a certain amount of points/stamps, or as a visible button on the E-receipt for use at the User’s discretion.

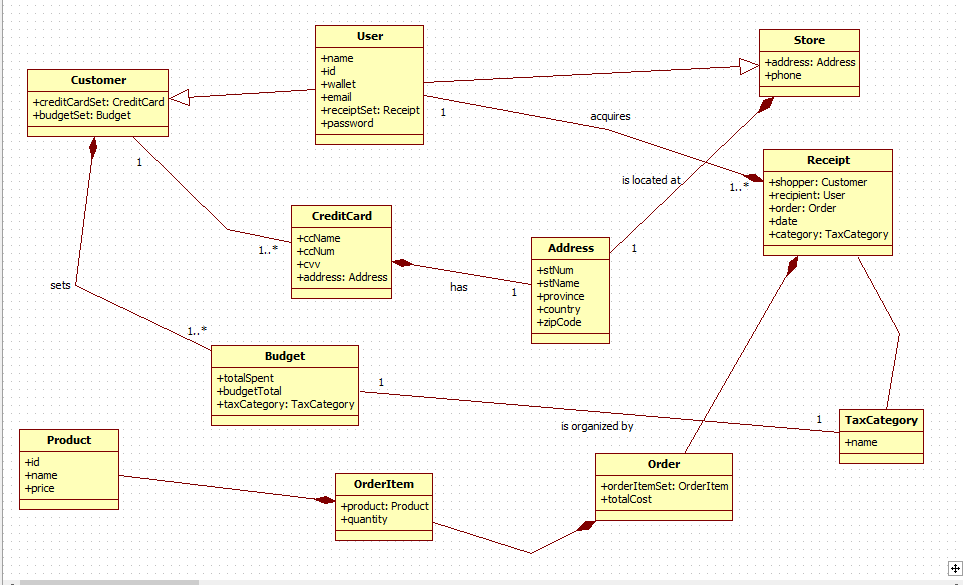
# Future Developments

While ShopHawk is starting as an answer to a very new and modern problem, there is room for it to grow past the current crisis. In the future we would like to create a system that will allow not only transactions between businesses and their clients, but for individual use as well. We would like to create another avenue for a person to pay their rent, pay back a loan from a friend, or for parents to give their children an allowance. The added benefit of a generated E-Receipt will make keeping track of these personal finances a more convenient task.

# Classes

#### Anatomy of an E-Receipt

The most integral item in this Application is the composition of an E-Receipt. Each Receipt is composed of Several Classes which make up the information that is presented in the final product.



##### User

##### Customer

##### Store

##### Product

##### Order

##### OrderItem

##### Receipt

##### Budget

# How to use ShopHawk

This is a prototype of our application that we will use to prototype:

<https://www.figma.com/file/OePx6UT6TXVOvadxM6wNxm/Hackathon---Components---Design?node-id=0%3A1>