# Report

# POS(POINT OF SALE) SYSTEM

## 1.Overview of the Software:

We have worked together to make a point of sale system for a grocery store and designed it according to our knowledge of programming and databases. For this point of sale(POS system), we have used IntelliJ, SQLite and java scene builder to develop Java GUIs using JavaFX. For this project, this first thing that we did was making a SQLite database of products and employees. Along with that, we started with mockups and UML diagrams for our POS system. The best thing about this project is everything is well maintained. We started with UML and mockups which acts like a starting point for the development process. The employee database is encrypted for security reasons. This software is useful because it saves time and simplifies existing business processes. For instance, suppose if you go to a grocery store and the employee does everything by hand, it would be hard for employees to handle all the customers since grocery stores are always busy. In this way, POS usage and functionality are helpful.

In this POS system, we are primarily focusing on two main things, one is sale of products in a grocery store and then return items that customers do not want anymore. In this software, an employee just has to login with their credentials and then start scanning items that are purchased by the customer, where totals and tax are calculated.

## 2.Target industry:

This kind of software is useful for every industry that does things related to sales. This software is targeted to a grocery store, as the products stored are food related. This kind of system reduces manual work and simplifies the work which is done by the human.

## 3.Description of UML Diagrams:

We have made four UML diagrams for this software.

The first one is a class diagram which shows how the classes interact. Application is the starting class, which Main inherits from. Any other forms associate with Main. All the events are handled by the controller which uses system logic for scanning, tax and database access.

The second UML is a sequence diagram which shows how system logic and employee base interact with each other and product code base. If login is found, it proceeds further and if it is not found, the POS system does not open. The same thing is with the system logic and product code base. If product code is found, the sales proceed and if not, then it displays a message.

The third UML diagram is a state diagram for returns. This diagram shows the return process. Which involves entering items to be returned, calculating amount owed, and paying the customer. The fourth UML diagram is a state chart diagram for sale action. Which involves entering items to purchase, tabulating totals and accepting payment.

# 4. Project Retrospective

- ## Easy thing to implement:

  Now, we are going to explain more about our experience while doing this project. The easiest thing to implement was the basic functionality. This project was simple. We needed two basic things to be done. One was sale and the second one was return. So, we focused on these two different things separately. After we got the mockups and UMLs done, things fell into place easily.

- ## Most difficult part:

  We would say that the most difficult thing in this project was the database functionality, due to needing to learn SQL queries.

- ## What would we do if we go back:

  If we would go back, we would add more functions to our POS system. The functions like coupons, discounts and some things related to inventory control so that our system keeps tracks of how much stock is being purchased by the customers. This would make it easier for the store to keep track of the items in stock and more popular products. Then the next thing that we might want to do is keep a graph on our POS system that records how our sale is going. This would show what people prefer to buy more. This would show more popular to the least popular product sale. We believe that these would be cool things to implement.

- ## Learning Outcomes:

  With this project, we have learnt many cool things. All the new things that we have learnt in COIS 2240, we got to implement them here starting from UMLs, java scene builder and SQLite. Implementation of all these things in this project was wonderful.