**Online Supermarket Management System**

**Hermain Qadir – 19K-1517 – Section H**

**Saman Khan – 19K-0354 – Section H**

**Anusha Saad - 19K-0281 – Section H**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**0. Acknowledgment**

We took help from quite a few sources on YouTube etc., but two entities had been specifically helpful. They are as follows:

- Ahmed Raza (19K-0134 – Section H)

- Stack Overflow.

Both helped when my mind went totally blank which happened quite a lot.

**1. Introduction**

Considering the current situation of pandemic and increasingly fast-paced lives, it becomes tedious to go out to shop for grocery and other miscellaneous items needed at home. The goal of our project was to automate the entire idea of supermarkets and provide the ease of shopping from home.

Our management system provides an easy way for people to shop from home for everything ranging from grocery to other household items. It helps the admin to get notified whenever a certain item is about to run out. It also has a customer panel which is easy-to-use. Customer can order items from that panel all at once by adding them to cart. The system comes with additionally showing a running cumulative bill, so the customer is keeping track of the budget spent.

The project has various features like admin having the option to change stock count or cost of an item, checking notifications or details about a certain item. On the other hand, customer can use promo codes to avail discount and choose items from any of the four categories available.

**2. Tools and technologies used**

- Visual Studio 2019 IDE

- <iostream> library

- <fstream> library

- <string> library

- <cstring> library

- C++/CLR **.**NET Framework

- <msclr\marshal\_cppstd.h> library

**3. Class diagram**

|  |
| --- |
| Customer |
| -cust\_name : string  -CNIC : string  -address : string  -mobno : string  -email : string |
| + writedetails : void  + viewhistory : void |

|  |
| --- |
| Item |
| -ID : char  -stock : int  -cost : int  -bill : static int |
| + UpdateBill (int) :void  + decrementstock :void  + updatestock (int) : void  + updatecost (int) :void  + getID () : char\*  +getstock () :int  +getcost() :int |

|  |
| --- |
| Apple |
|  |
| + writeuser :void |

|  |
| --- |
| Banana |
|  |
| + writeuser :void |

|  |
| --- |
| Carrot |
|  |
| + writeuser :void |

|  |
| --- |
| Potato |
|  |
| + writeuser :void |

|  |
| --- |
| Toothpaste |
|  |
| + writeuser :void |

|  |
| --- |
| Shampoo |
|  |
| + writeuser :void |

|  |
| --- |
| Rice |
|  |
| + writeuser :void |

|  |
| --- |
| Flour |
|  |
| + writeuser :void |

|  |
| --- |
| Toothbrush |
|  |
| + writeuser :void |

|  |
| --- |
| Oil |
|  |
| + writeuser :void |

|  |
| --- |
| Egg |
|  |
| + writeuser :void |

|  |
| --- |
| Milk |
|  |
| + writeuser :void |

|  |
| --- |
| Chicken |
|  |
| + writeuser :void |

|  |
| --- |
| Cheese |
|  |
| + writeuser :void |

|  |
| --- |
| Lentils |
|  |
| + writeuser :void |

|  |
| --- |
| Butter |
|  |
| + writeuser :void |

|  |
| --- |
| Soap |
|  |
| + writeuser :void |

Customer class served the purpose of storing a customer’s details and inputting those into a file in order to manage customer history. Item class served as parent class to all the items that are sold in the supermarket. It holds ID, stock count, and price and those variables are common to all the items. The other seventeen classes each by the name of specific item are publicly inherited from the item class. Each item is a separate class in order to maintain files of every item specifically and has its own writeuser method.

**4. Link to source**

[Project Link](https://drive.google.com/file/d/1Zi5TIRE9ej9YO7S_XXPk912C0rgWMqmX/view?usp=sharing)

**5. Future work**

The project has room for a lot of improvisation. The admin panel could have more options, like adding an item or checking customer history. Moreover, we can work more upon aspects like generating a receipt for customer when shopping ends because currently the customer can only see one’s total bill. Customer could also have been given option to cancel anything that may have been mistakenly added to cart. Lastly, sales, offers etc. could also have been added.