CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

Nowadays we live in a world where the technology is rapidly growing. Complex system as internet have an enormous impact on our daily life. Almost all people have access to the internet and make use of it for numerous things such as paying bills, phone someone or shopping. Most people buy a lot on the internet such as books, clothes and multimedia. However, groceries are not very comparison with the other products. The online grocery shopping rate is below then offline grocery store, So, we have been preparing this platform for inline grocery store which will be very friendly to our customers, as it will be very helpful for the customers as it has many advantages such as no waiting in the lines, no parking space needed, no old people only if they do online shopping, they can avoid these negative association due to these problems it could be successful the online grocery shopping. The grocery house is a store where the users can buy grocery items for their house holds according to their needs such as fruits, vegetables, dairy etc. within their own budget. Although due to confusion layout, dead stock, out of stock, non- sale items, out of stock on advertised items. To overcome this this problem, I have decided to develop a shopping platform where the customer can make a list of his items according to his/her budget. Building Sales and profits-The major objective of a store companies is to sell products and earn the highest profits. Therefore, the main thing is to increase the customer traffic, building the customer loyalty and improve the department ratings.

1.2 OUTCOME

To make a list of grocery items for an, grocery store with name, price and quantity; If the list already contains an item then only, we will update the price and quantity and it should not append the item again. We will ask the user their budget initially and deduct the budget after adding a new item in the list. If the user’s budget becomes ‘0’, then no more items could be bought and if some money gets left and the user adds an item greater than their budget left, then it would inform “over price” or” No budget left”. After the list is made any money left in the budget it should show an item within the budget from the list made that can be bought. However, the online several studies determine that online consumer expenditure on grocery sales is only a couple of percent of its total group sale. Through the internet customer have to use a new technology with no interactions between humans. Since there are only less online grocery store so it will be profitable to set up an online super market. There are some significant differences between offline and online shopping process which are very important for managers to fully understand. The customer preferences are important, because the preferences for online supermarket are probably different from offline supermarket. The complete research will be very helpful in setting up an online supermarket since it includes all the aspects that need extra attention.

1.3 OBJECTIVE

- The major objective of a store companies is to sell products and earn the highest profits.
- Building sales.
- Building profits.
- To ensure continuous supply of item.
- To maintain the availability of items whenever required in enough quantity.
- To ensure the good quality of the products.
- To sell the products in a reasonable price.

CHAPTER 2

REQUIREMENT SPECIFICATION

2.1 SOFTWARE REQUIREMENTS

- Windows operating system, LINUX or IOS.
- Efficiency manage inventory.
- Easy to learn and implement.
- Smart notification.
- My SQL.

2.2 HARDWARE REQUIREMRNTS

- A PC with windows operating system.
- RAM: 4GB RAM or more.
- Processor: Intel atom or Intel Celeron.
- Basic GPU.
- Secondary memory of 2GB or more.
- Hard Drive: 64GB minimum.

CHAPTER 3

PYTHON FUNDAMENTALS

Python is a scripting programming language known for both its simplicity and wide applications. The software development companies prefer Python language because of its various features and fewer programming codes. Nearly 14% of the programmers use it on the appear systems like UNIX, Linux, Windows and Mac OS.

The programmers of full companies use Python as it has created a mark for itself in the software improvement with characteristic features like-

- Interactive mode: involves running code directly on shell.
- Interpreted: It uses interpreter which turns code into a language

  understandable by computer's processor.
- Modular: File consisting of python code which can define functions, classes

  & variables
- Dynamic: doesn't know about the type of variable until the code is run.
- Object-oriented: Allows developing applications using oops concepts
- Portable: one code can run on different machine there is no need to write

  different code for different machines.
- High level: It has to run in a given system using another program instead

  of local processor.
- Extensible in C++ & C: can write python code in other languages.

3.1 DATABASE FUNDAMENTALS

A database is a collection of related data and a set database management system (DBMS) is a collection of programs that enables users to create and maintain a database.

Database management systems provide respective functions in addition to simple file management:

- allow concurrency: ability to allow multiple users
- Control security: To reduce the risk
- maintain data integrity: Accuracy and consistency of data

- Provide for backup and recovery: creating and storing copies of data against loses.

- Control redundancy: to speed up database access
- allow data independence: refers to changes made in data organization
- Provide non-procedural query language: To produce the desired result without the step by step process.
- Many conflicting individuals are involved with a database management system over its life: systems analysts, database designers, database administrators, application developers, users.

3.2 ADVANTAGES

- Reducing Data Redundancy**:** The file-based data regulation systems contained multiple files that were stored in many different locations in a system or even across multiple systems. Because of this, there were sometimes multiple copies of the same file which lead the data decrease. This is prevented in a database as there is a single database and any change in it reflected immediately. Because of this, there is no chance of encountering clone data.

- Sharing of Data**:** In a database, the users can be database can share the data among themselves. There are different levels of approval to access the data.
- Data Integrity**:** Data integrity means that the data is correct and of a piece in the database.
- Data Security**:** Only authorized users should be allowed to access the database and their identity should be authenticated using a username and password. Unauthorized users should not be allowed to access the database under any circumstances as it opposes the integrity constraints.
- Backup and Recovery**:** Database Management System automatically takes care of the backup and recovery. Moreover, it also restores the database after impact or system failure to its previous condition.
- Data Consistency**:** All data appears consistently across the database and the data is same for all the users viewing the database. Moreover, any changes made to the database are immediately reflected to all the users and there is no data inconsistency.

3.3 DISADVANTAGES

- Time-Consuming Design**:** Converting from paper files to an electronic database system can be composite, crucial and time-consuming. Different steps must be followed when switching over to a database. The project leader must determine the direction of the database and collect all the information that needs to be organized.
- Potential Technical Problems: The database may encounter errors that damage virtually all information within it. Small miscalculation can lead to the need for a major maintain of the system. Regular backups are critical to limit the potential for devastation.
- Cost of Conversion**:** Converting to a database can be very costly. The business owner will incur several costs.

CHAPTER 4

ER DIAGRAM

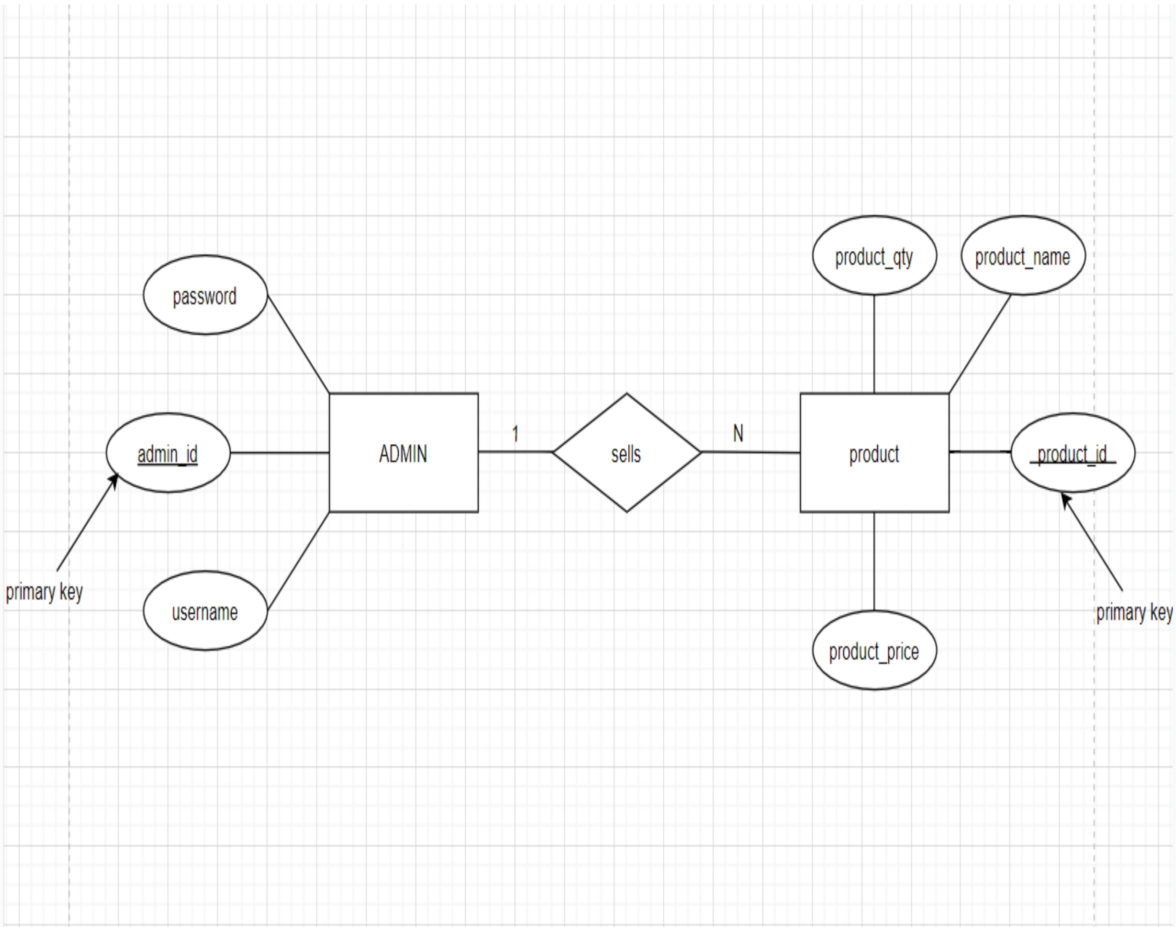


Fig 4:1 ER Diagram

CHAPTER 5

IMPLEMENTATION

5.1 PARTIAL CODE

```
 # This loop will go on until the budget is integer or float

while True:

    try:

        bg = float(input("Enter your budget : "))

        # if budget is integer or float it will be stored

  # temporarily in variable 's'

        s = bg

    except ValueError:

        print("PRINT NUMBER AS A AMOUNT")

        continue

    else:

        break

  # dictionary to store product("name"), quantity("quant"),

# price("price") with empty list as their values

a ={"name":[], "quant":[], "price":[]}
```

```
 # converting dictionary to list for further updation
b = list(a.values())
# variable na value of "name" from dictionary 'a'
na = b[0]
 # variable qu value of "quant" from dictionary 'a'
qu = b[1]
  # variable pr value of "price" from dictionary 'a'
pr = b[2]
  # This loop terminates when user select 2.EXIT option when asked
# in try it will ask user for an option as an integer (1 or 2)
# if correct then proceed else continue asking options
while True:
    try:
        ch = int(input("1.ADD\n2.EXIT\nEnter your choice : "))
    except ValueError:
        print("\nERROR: Choose only digits from the given option")
        continue
    else:
  # check the budget is greater than zero and option selected
# by user is 1 i.e. to add an item
if ch == 1 and s>0:
     # input products name
   pn = input("Enter product name : ")
   # input quantity of product
   q = input("Enter quantity : ")
   # input price of the product
   p = float(input("Enter price of the product : "))
  if p>s:
      # checks if price is less than budget
  print("\nCAN, T BUT THE PRODUCT")
      continue
   else:
      # checks if product name already in list
      if pn in na:
         # find the index of that product
         ind = na.index(pn)
         # remove quantity from "quant"  index of the product
         qu.remove(qu[ind])
```

```
        # remove price from "price" index of the product

        pr.remove(pr[ind])

         # insert new value given by user earlier

        qu.insert(ind, q)

        # insert new value given by user earlier

        pr.insert(ind, p)

        # subtracting the price from the budget and assign

        # it to 's' sum(pr) is because pr = [100, 200] if

        # budget is 500 then s = bg-sum(pr) = 200

        # after updating for same product at index 0 let

        # pr = [200, 200] so s = 100

        s = bg-sum(pr)

        print("\namount left", s)

    else:

        # append value of in "name", "quantity", "price"

        na.append(pn)

         # as na = b[0] it will append all the value in the

        # list eg: "name":["rice"]

        qu.append(q)
```

```
                 # same for quantity and price

                  pr.append(p)

                  # after appending new value the sum in price

                  # as to be calculated

                  s = bg-sum(pr)

                  print("\namount left", s)

       # if budget goes zero print "NO BUDGET"

       elif s<= 0:

           print("\nNO BUDGET")

else:

    break

  # will print amount left in variable 's'

print("\nAmount left : Rs.", s)

if s in pr:

    print("\nAmount left can buy you a", na[pr.index(s)])

  print("\n\n\nGROCERY LIST")

for i in range(len(na)):     print(na[i], qu[i], pr[i])
```

CHAPTER 6

RESULTS

6.1 Opening page

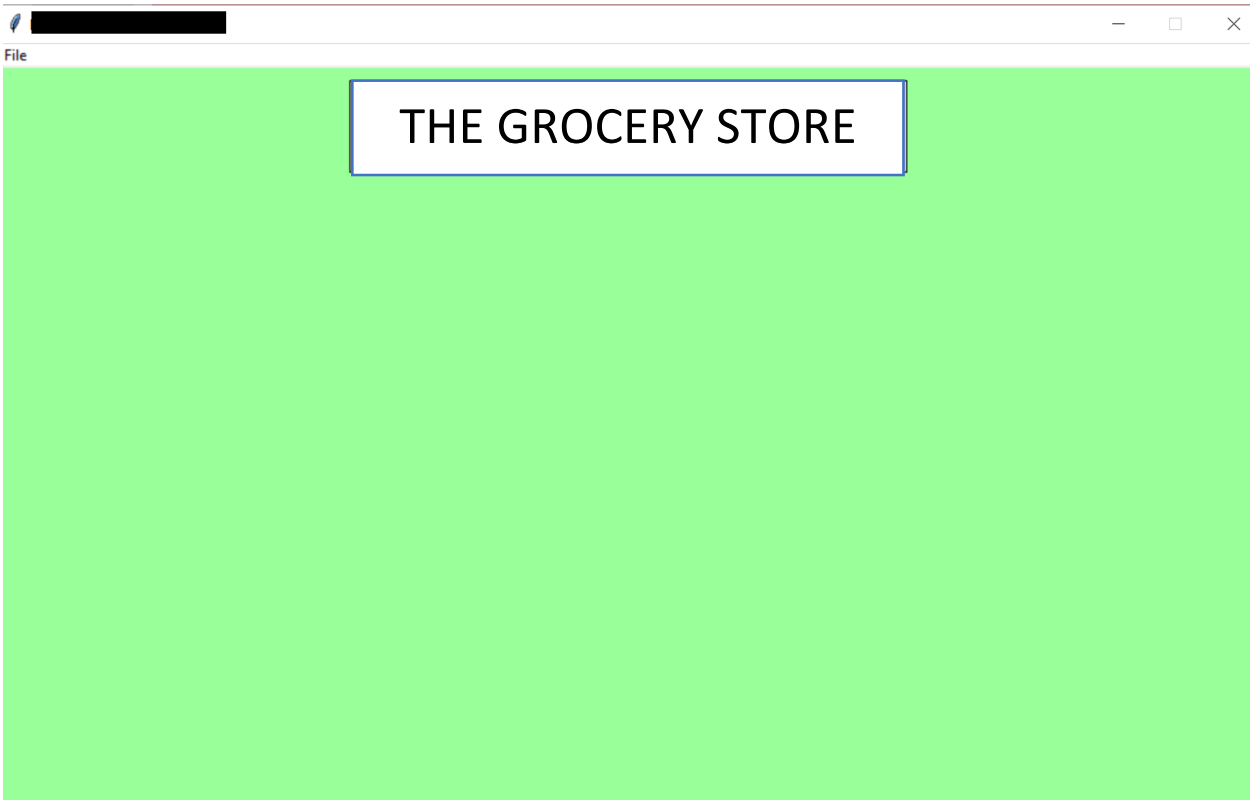
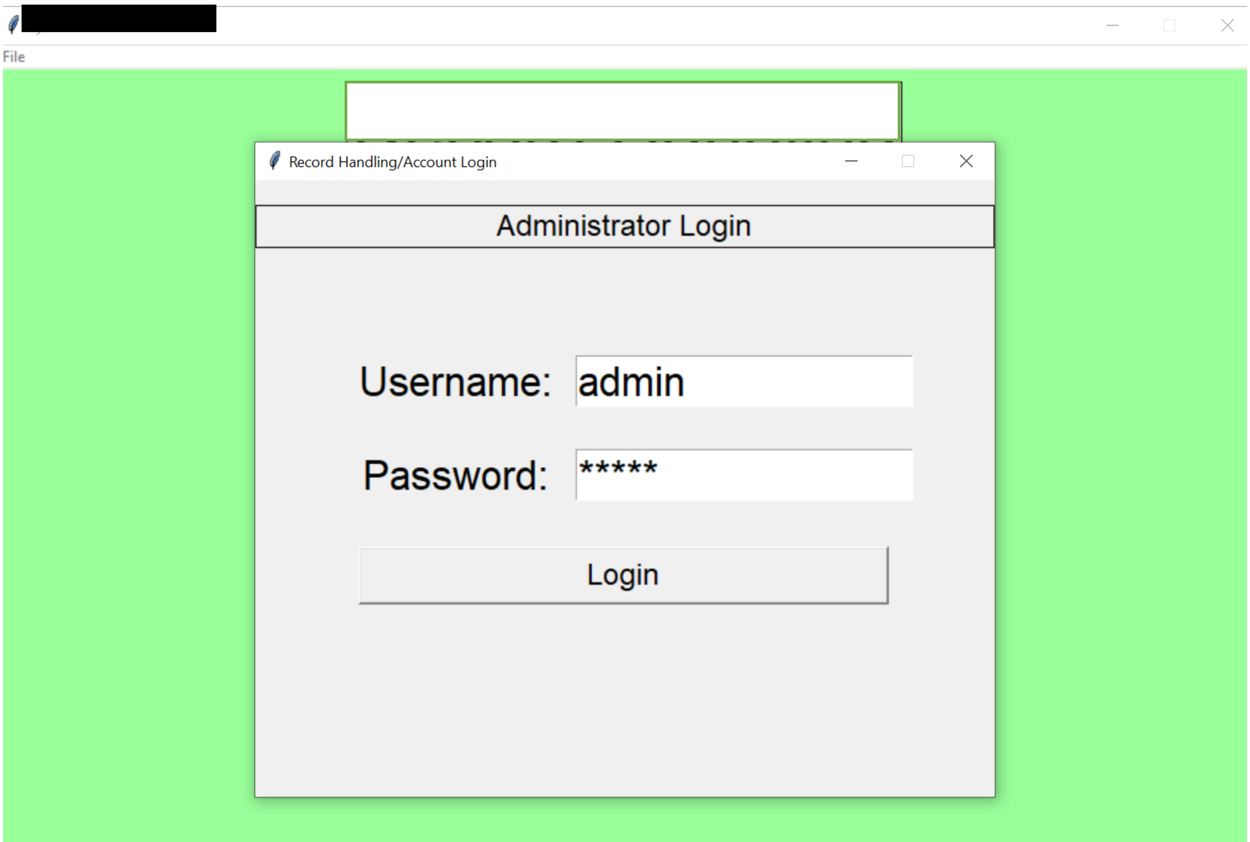


Fig 6.1: Opening Screen

6.2 Administrator login will be the second face.



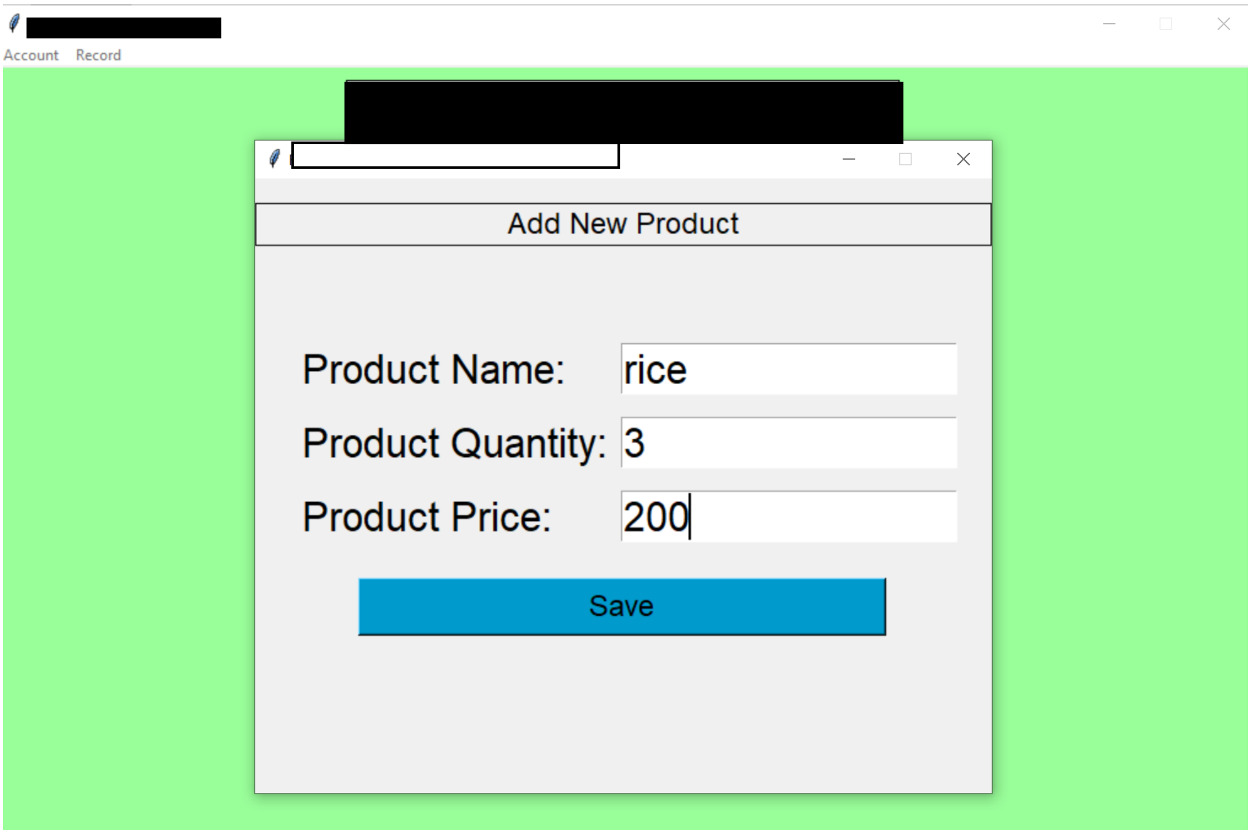
[redacted]

Username: admin

Password: *****

Fig 6.2: Login

6.3 Add new product of the list like product name, product quantity, and product price.



[redacted]

[redacted]

[redacted]

Add New Product

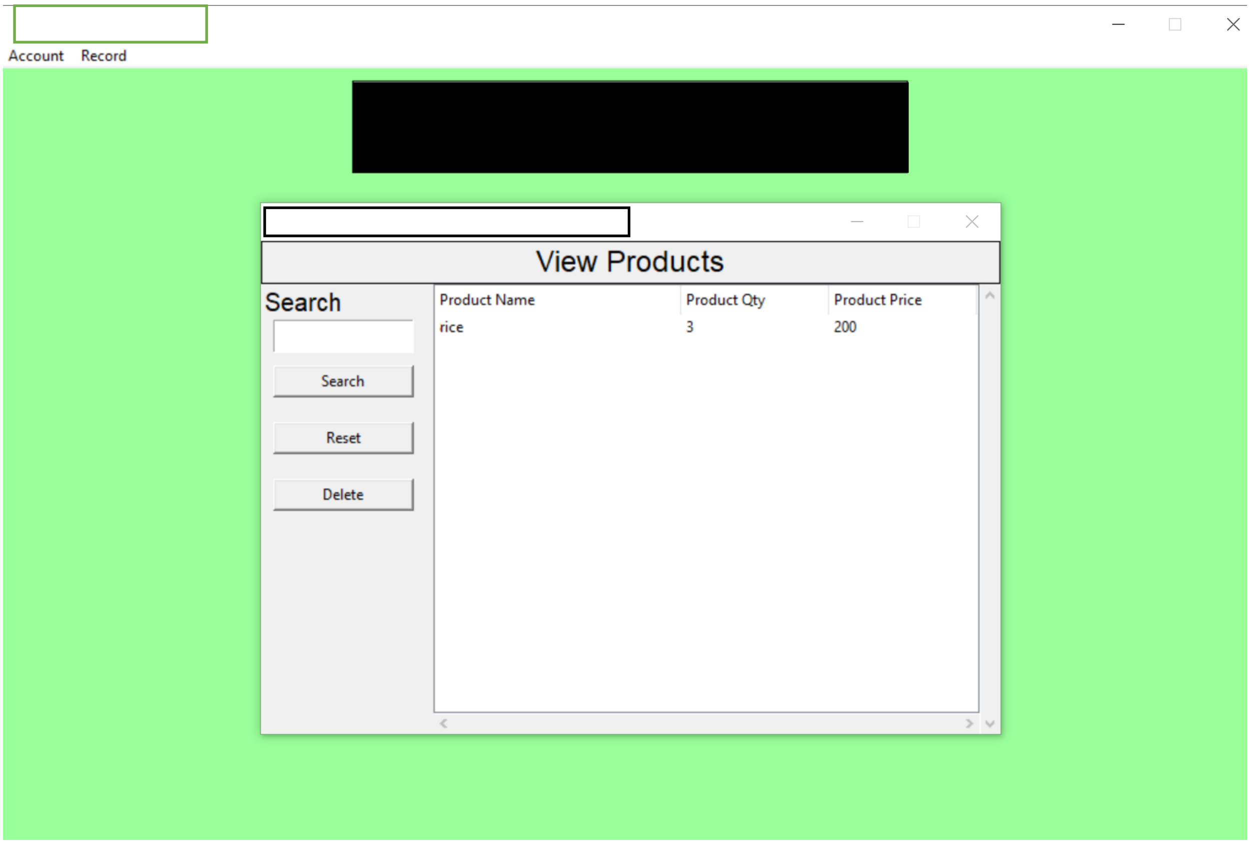
Product Name: rice

Product Quantity: 3

Product Price: 200

Fig 6.3: Product

6.4 The list of product which can be insert in the list and there are some operation like search, reset, delete.



[redacted]

View Products

Product Name	Product Qty	Product Price
rice	3	200

Fig6.4: Product list

CHAPTER 7

CONCLUSION

The online grocery shopping rate is below then offline grocery store, So we have been preparing this platform for inline grocery store which will be very friendly to our customers, as it will be very helpful for the customers as it has many advantages such as no waiting in the lines, no parking space needed, no old people only if they do online shopping they can avoid these negative association due to these problems it could be successful the online grocery shopping.

The grocery house is a store where the users can buy grocery items for their house holds according to their needs such as fruits, vegetables, dairy etc. Within their own budget. Although due to confusion layout, dead stock, out of stock.

To overcome this this problem, I have decided to develop a shopping platform where the customer can make a list of his items according to his/her budget. Building Sales and profits-The major objective of a store companies is to sell products and earn the highest profits.

REFERENCE

1. www.geeksforgeeks.org/python-maintaining-grocery-list/

2. https://www.tutorialspoint.com/python/index.htm

3.https://www.geeksforgeeks.org/sql-tutorial/

4.https://www.tutorialspoint.com/python/python gui programming.htm

5.https://opensourceforu.com/2019/04/database-proagramming-python/