TeleMedicine

Subhan Hussain

9315

Sahar Saleem

8684

Under the Supervision of

Usman Khan

2021

College of Computing & Information Sciences
PAF-KIET Main Campus, Karachi Pakistan

TeleMedicine

Subhan Hussain

9315

Sahar Saleem

8684

Under the Supervision of

Usman Khan

2021

College of Computing & Information Sciences
PAF-KIET Campus, Karachi Pakistan

College of Computing & Information Sciences

Main Campus, Karachi, Pakistan

CERTIFICATE

This project "**TeleMedicine**" presented by **Subhan Hussain (9315) Sahar Saleem (8684)** under the direction of their project advisor and approved by the project examination committee, has been presented to and accepted by the CoCIS, in partial fulfilment of the requirements for the bachelor's degree of Computer Science.

Mr.Moneeb Ahmed Usmani
(Project Advisor)

Usman Khan
(FYP Committee Head)

Dr.Khalid Khan
(Director CoCIS)

TABLE OF CONTENT

Abstract

Our project **TeleMedicine** design for medical stores. It is very fast , reliable and easy to use. It provides all the information to the admin about the inventory. It will mange and control all the stock. our project convert the manual chemist inventory in to automated inventory system. It manage all the records in a very organized form Telemedicine is a web application to maintain all type of data and our android application will capable to sell medicine online. A user can order online . This idea will provide medicine in lockdown situation because it's a home-base delivery application .

Acknowledgements

First of all, we would like to acknowledge Our Lord the Almighty Allah for everything He blessed us with and without Whom no endeavor including this one could have been successful than we would like to acknowledge our parents, all the teachers, institutes and mentors that have always supported us and helped us reach this level. Our most humble regards to the Director CoCIS, Khalid Khan, our Program Managers, Kashif Bashir and the outstanding FYP committee members, Nadeem Qamar and Abdul Wahab that have time and again guided us, tested us and clarified many problematic aspects of our work. Adding to the long list we are grateful to have had Moneeb Ahmed Usmani as our advisor.

Subhan Hussain
Sahar Saleem
2020-2021

1. Introduction

Our project is basically design to reduce the workload of the chemist. our project convert the manual chemist inventory in to automated inventory system It manage all the records in a very organized form.

1.1.Motivations

This project could be the wishful fulfillment for any Medical Store categorized for ease. While designing KIETBOOK, main concerns were to be addressed, platform independency. A further concern was to build the system in a generic way. It has to be possible for future developers to expand the system. Therefore, the system will have a modular structure.

1.2.Objectives and Scope

The principle objective of the Medical Shop Management System is to deal with the subtleties of Medical Shop. Our main objective is to manage the data of medical shop,stock and sells . Our project is made to reduce the workload manually and use digital or online technology to manage the medical store It is very use friendly , fast and reliable.

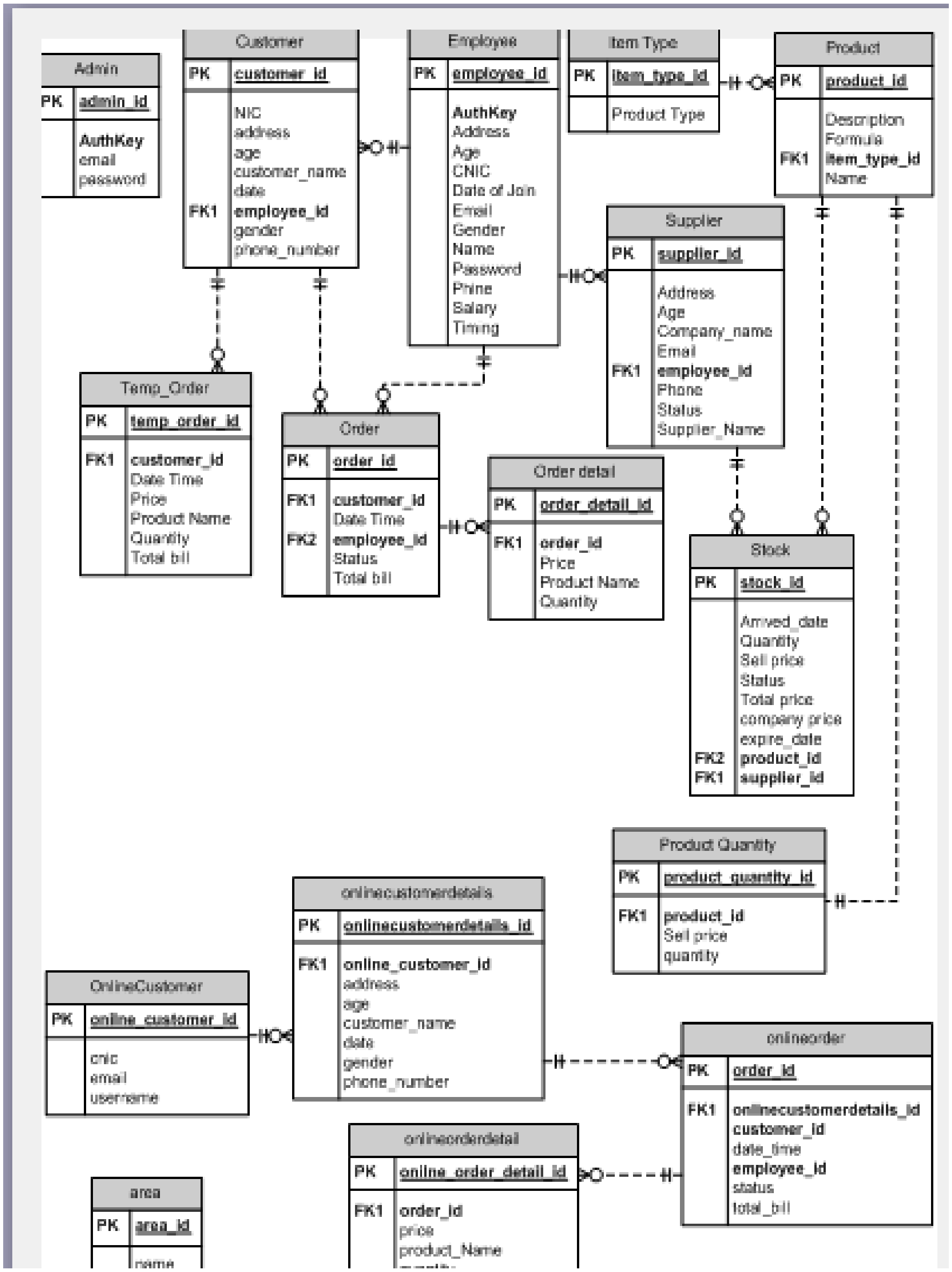
2. Methodology

Our approach was based on the waterfall methodology, Because the waterfall model is a breakdown of project activities into linear sequential phases, where each phase depends on the deliverables of the previous one and corresponds to a specialization of tasks. The approach is typical for certain areas of engineering design.

3. Design and Implementation

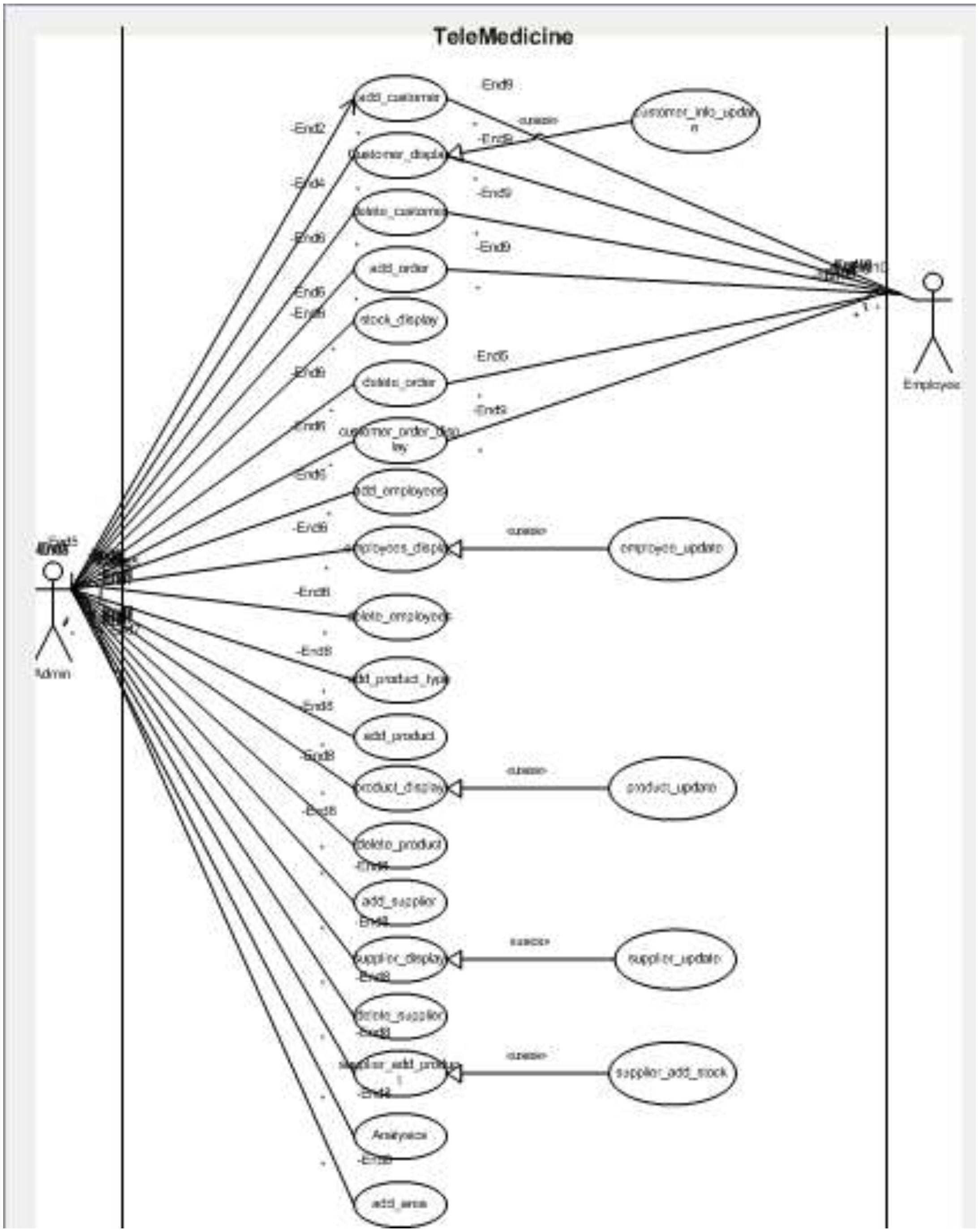
3.1.Design

3.1.1 ERD



This is the erd diagram of our project which shows all the graphical representation of database and tables or events

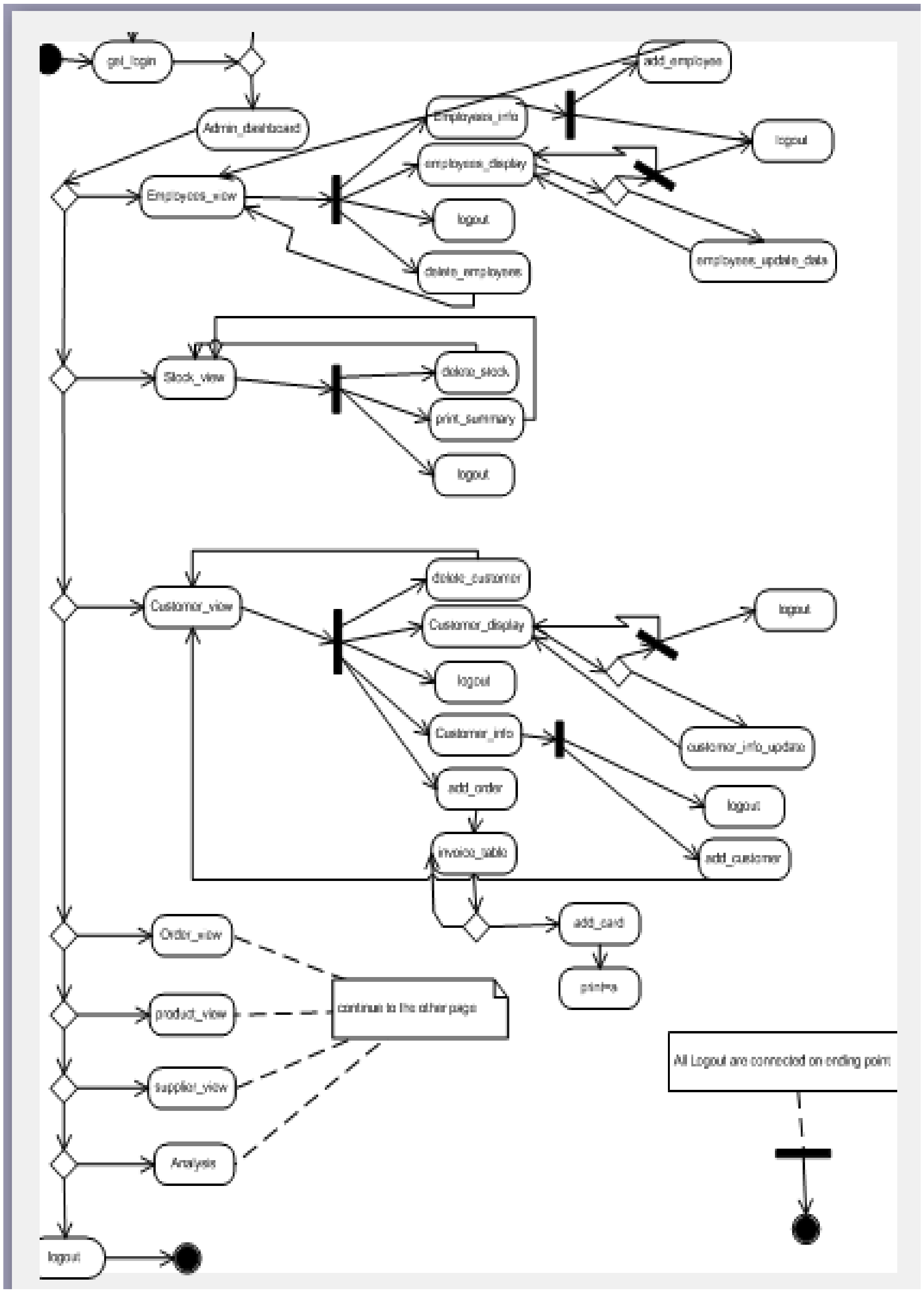
3.1.2 Use Cases



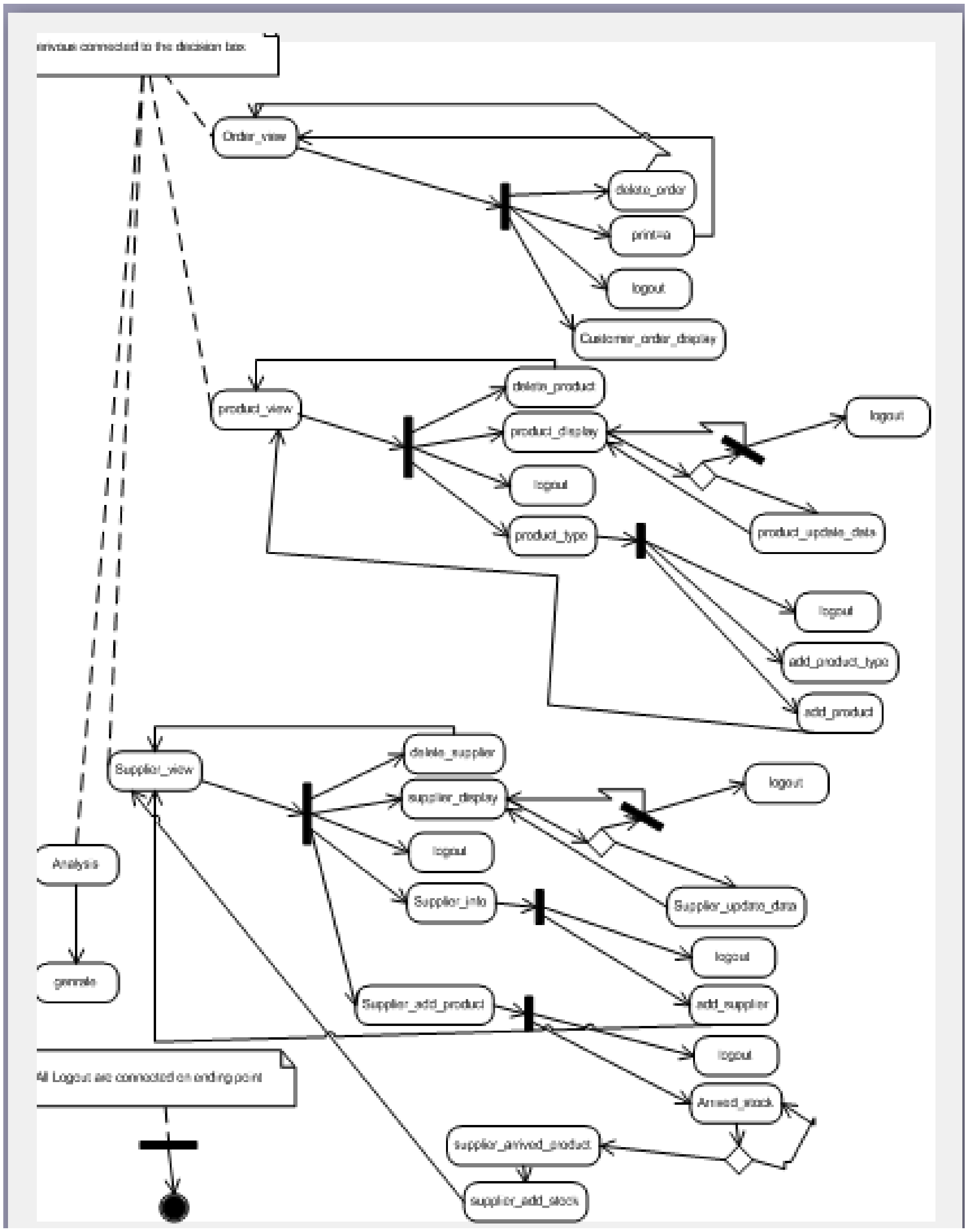
This is the use case diagram it will show how user will perform its tasks Each utilization case is addressed as a grouping of straightforward advances, starting with a client's objective and finishing when that objective is satisfied.

3.1.3 Activity Diagram

3.1.3.1 Admin P1

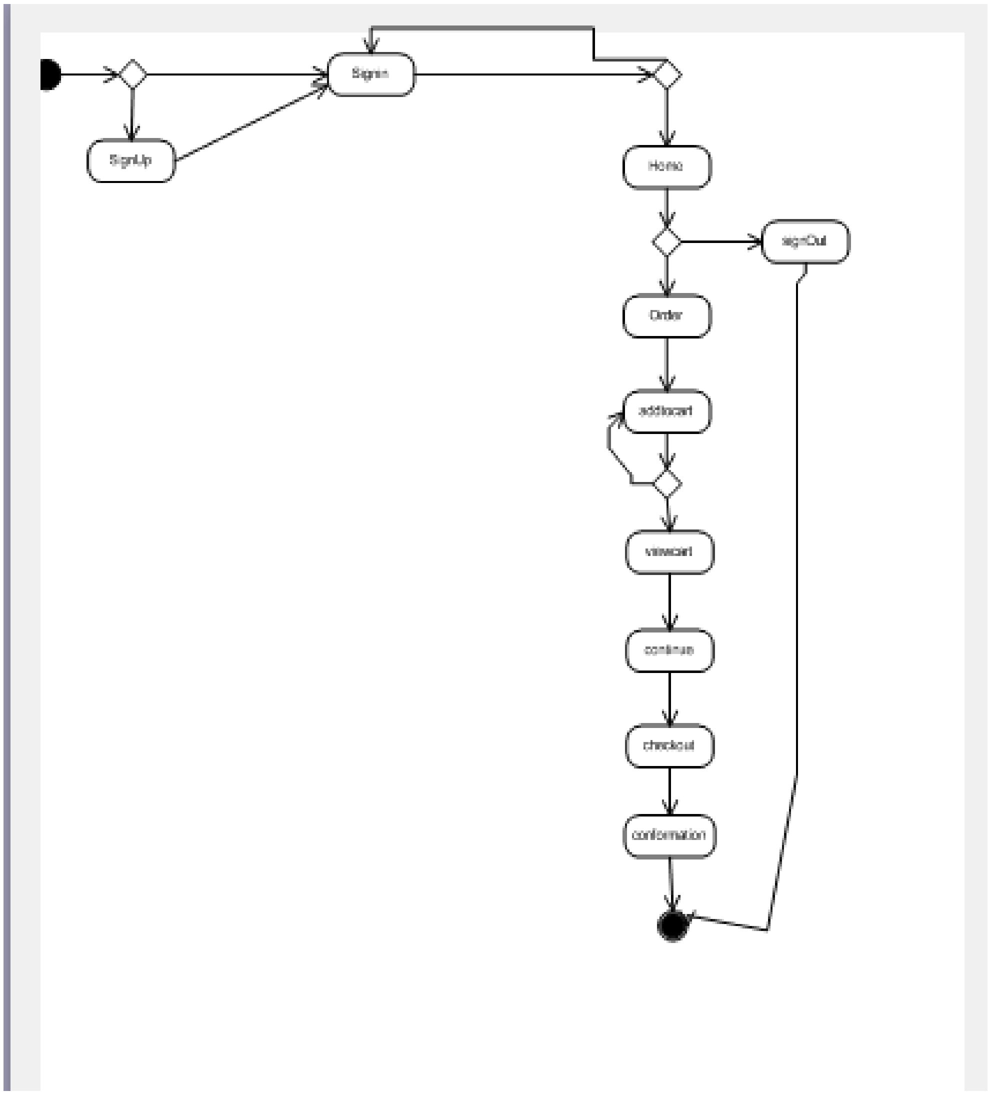


3.1.3.2 Admin P2



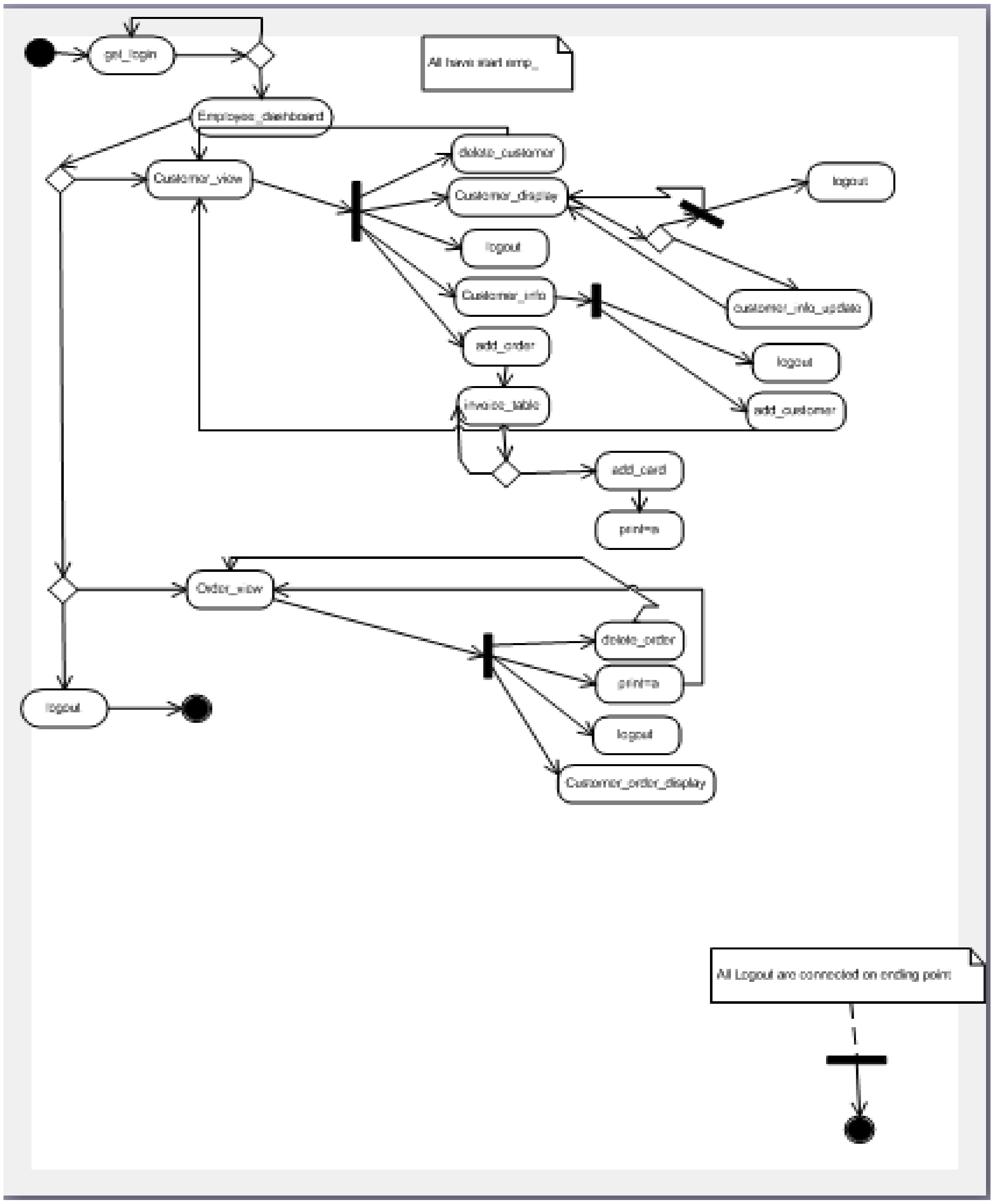
Movement chart outwardly present a progression of activities or stream of control Movement chart are regularly utilized in business prosses modeling.they can likewise portray the means in a utilization case outline.

3.1.3.3 Customer(android)



Activity diagram visually present a series of actions or flow of control Activity diagram are often used in business prosses modeling.they can also describe the steps in a use case diagram

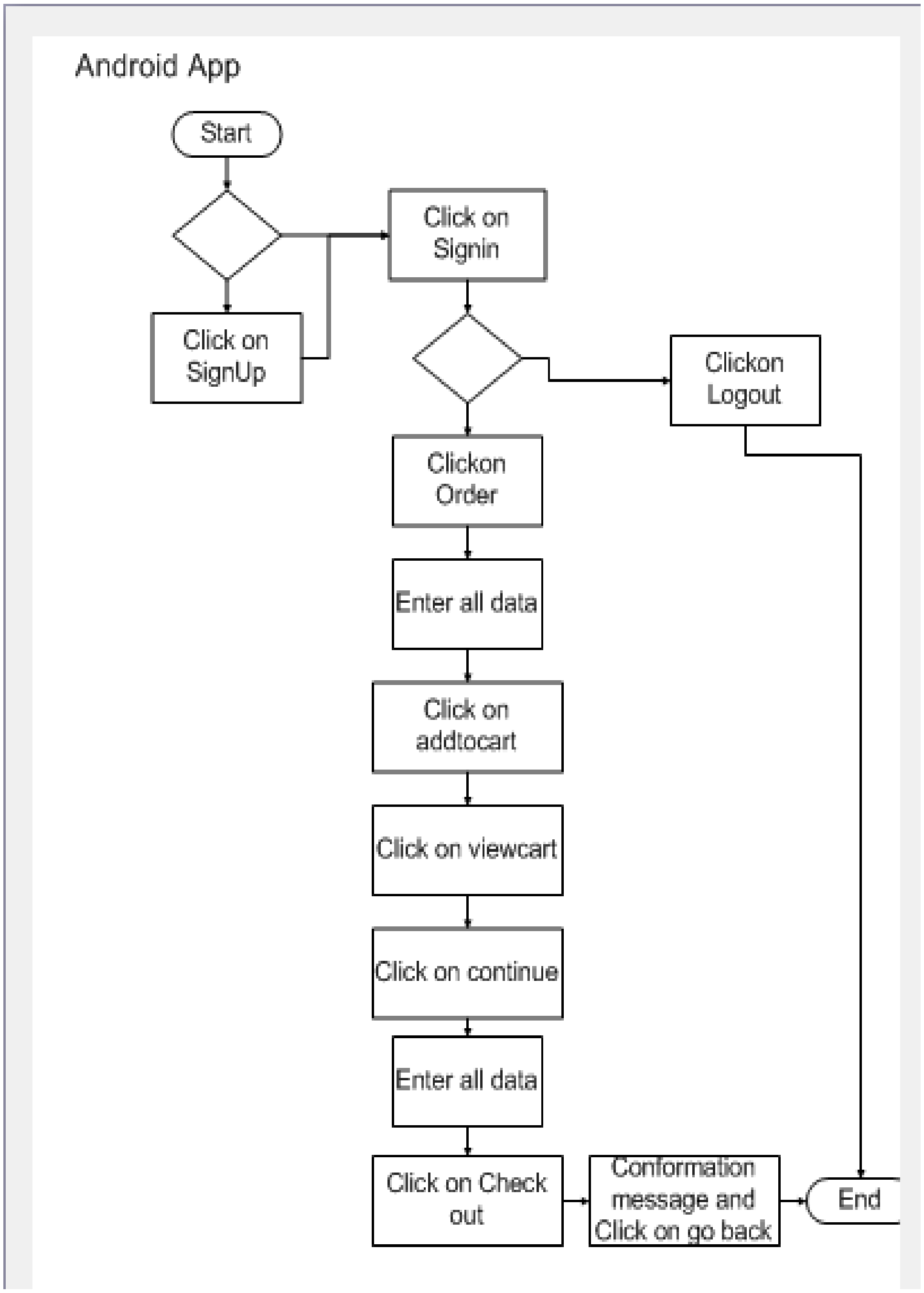
3.1.3.4 Employee



Activity diagram visually visually present a series of actions or flow of control Activity diagram are often used in business prosses modeling.they can also describe the steps in a use case diagram

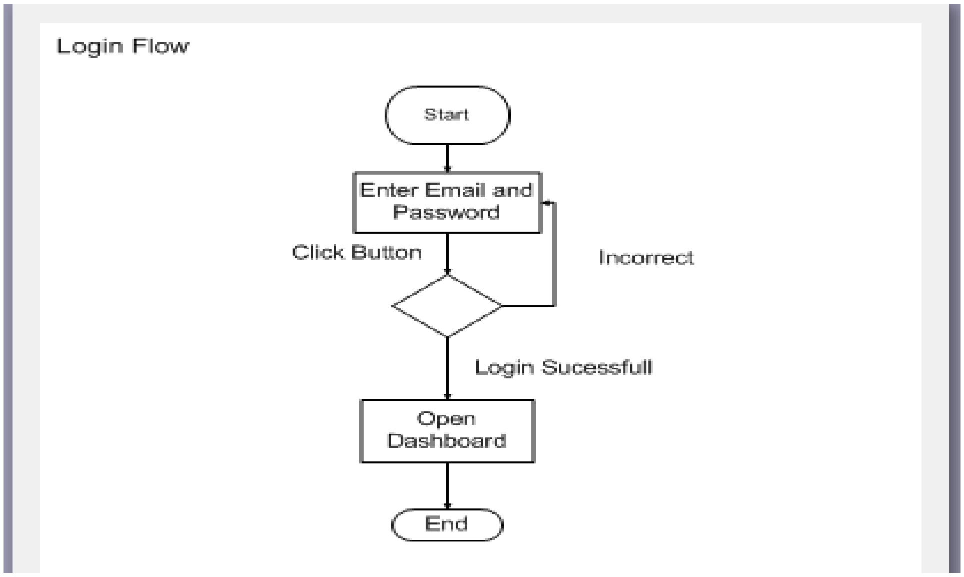
3.1.4 Flow Diagrams

3.1.4.1 Android App

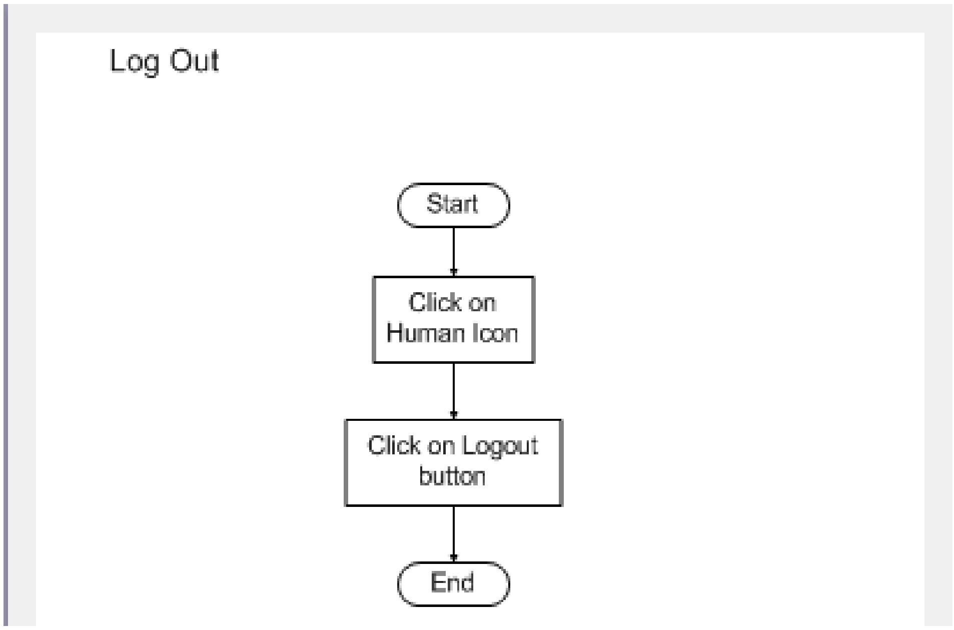


This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.2 Login

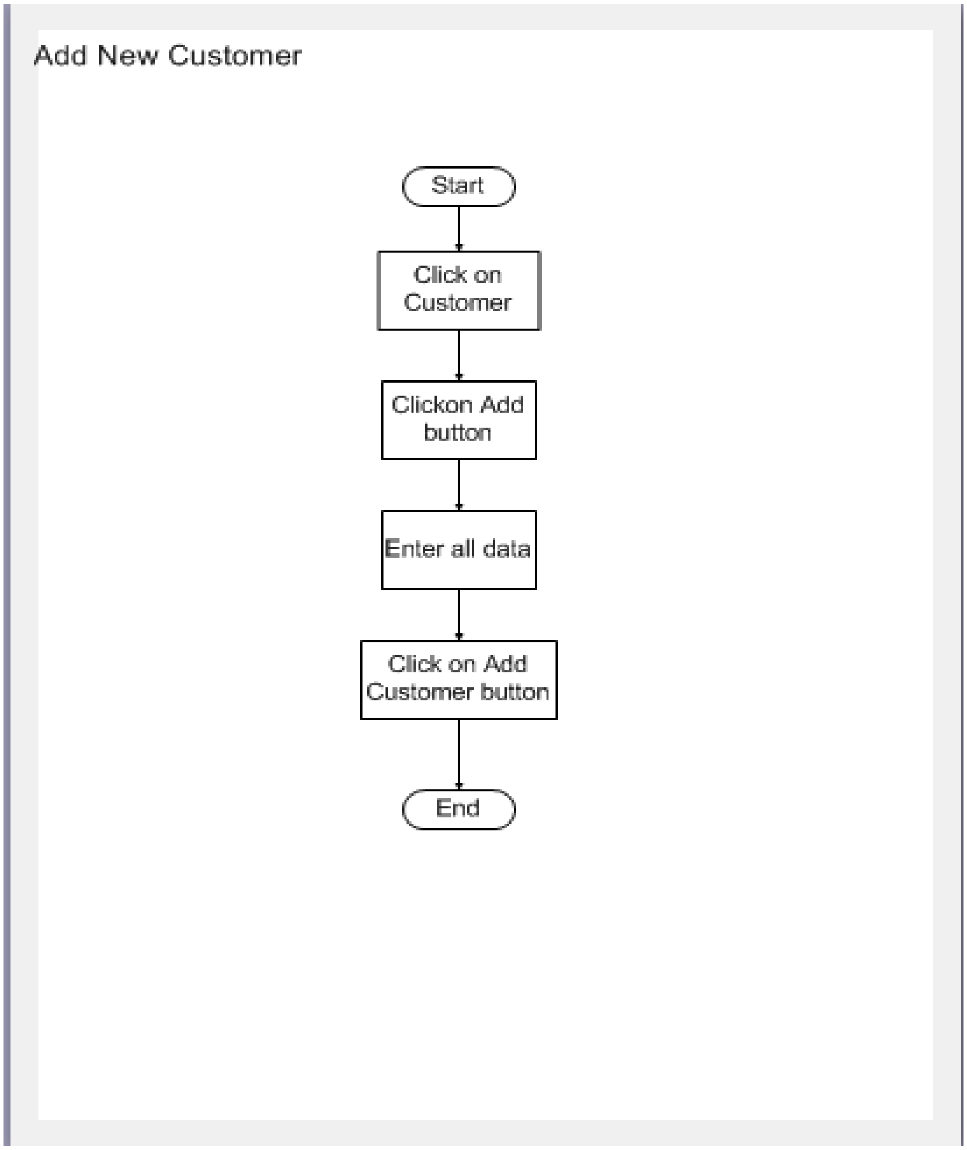


3.1.4.3 Logout



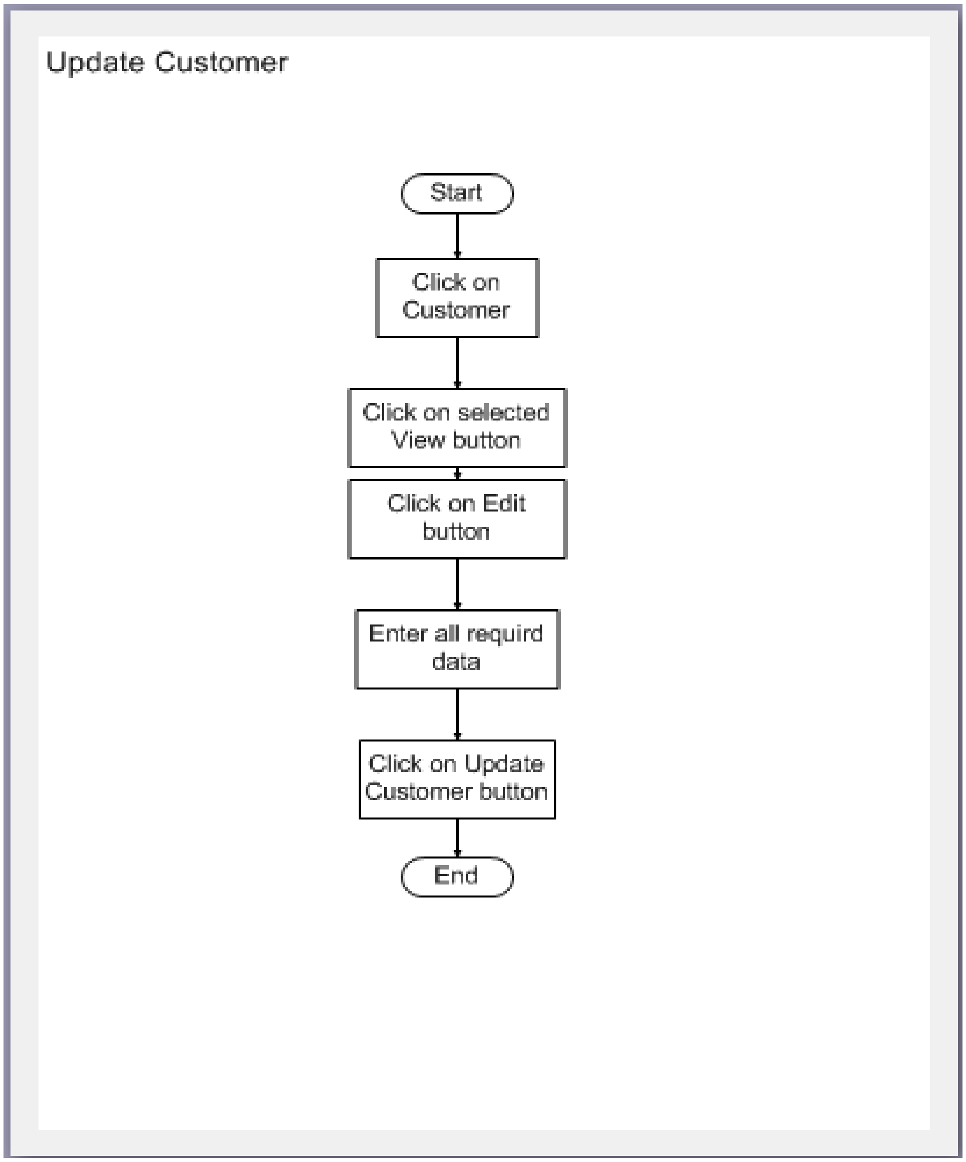
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.4 Customer Add



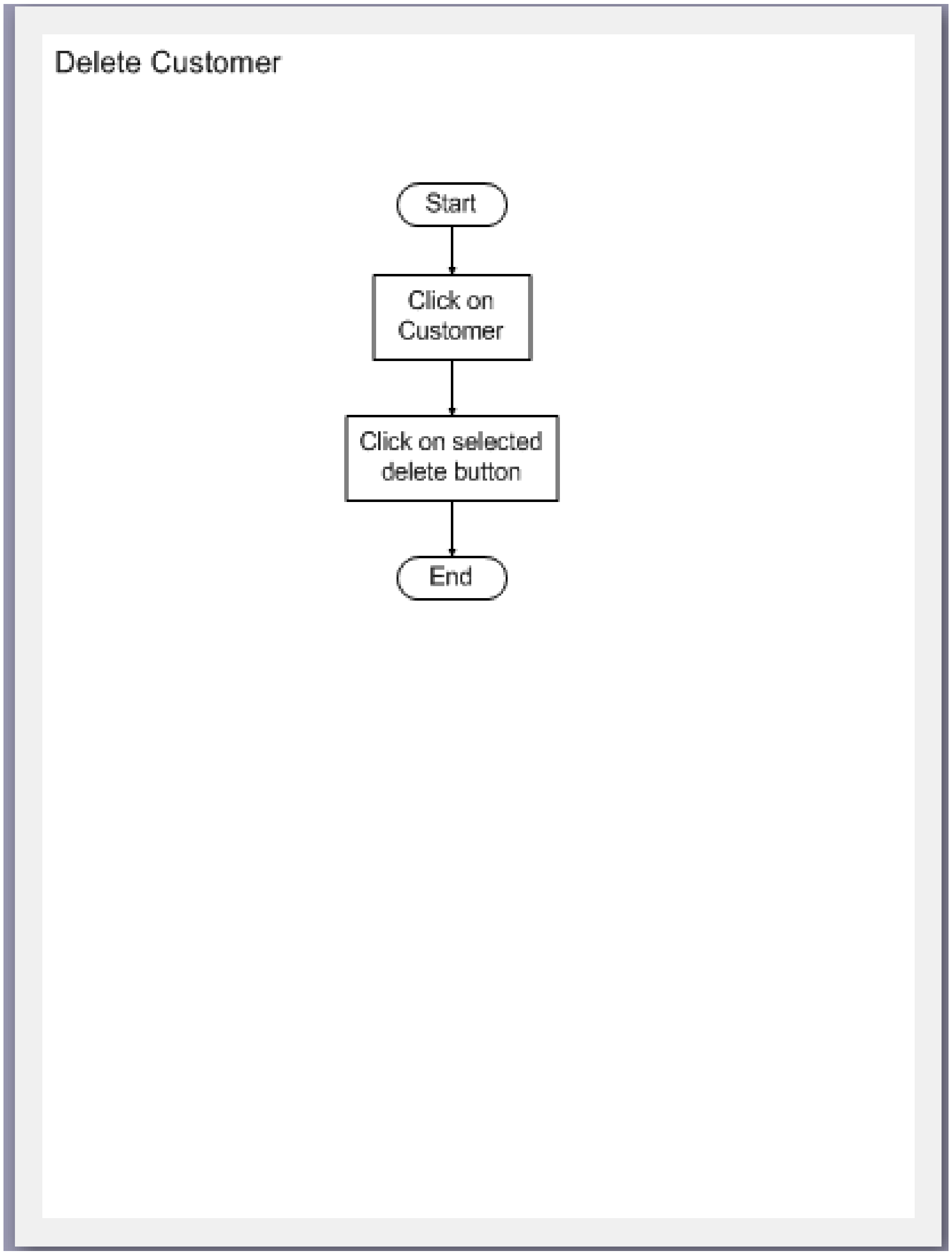
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.5 Customer Update



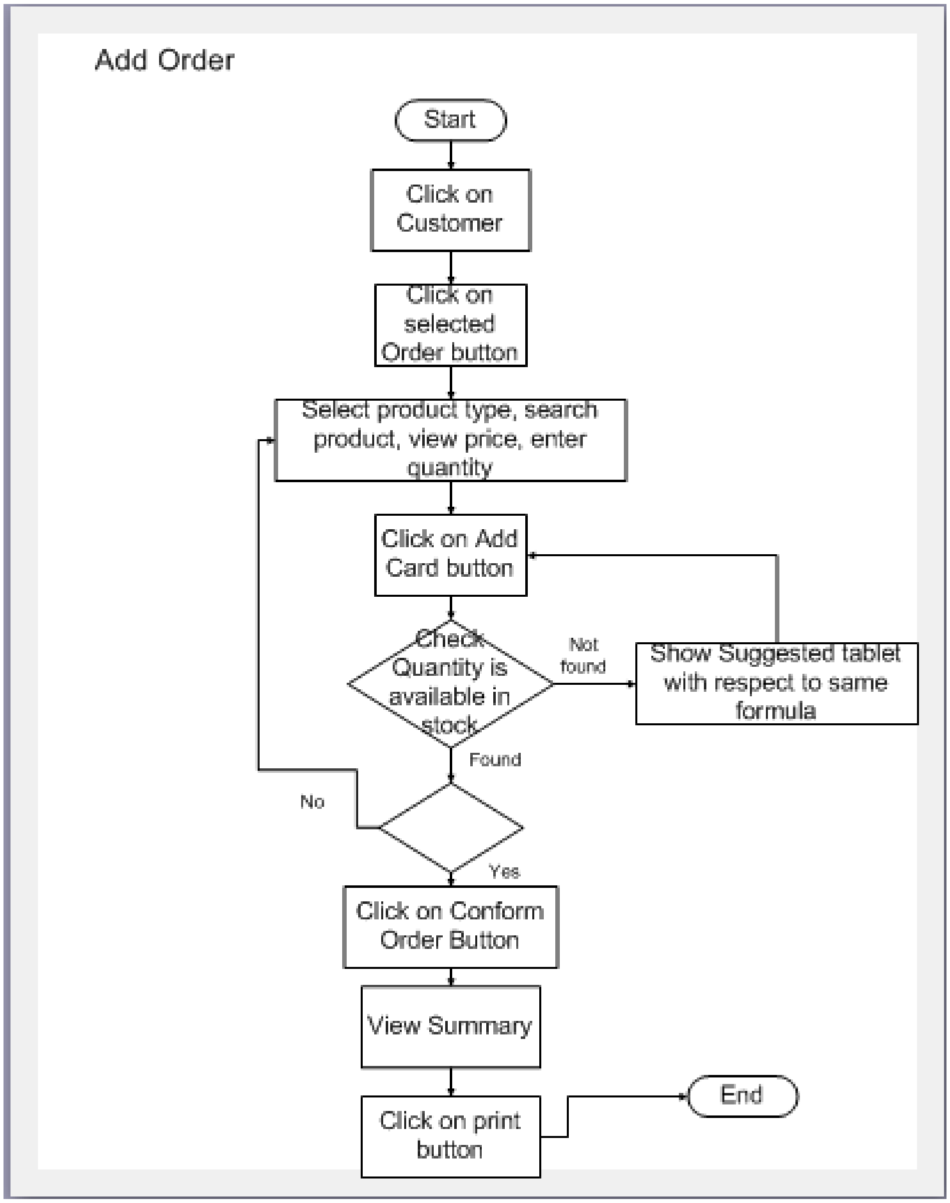
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.6 Customer Delete



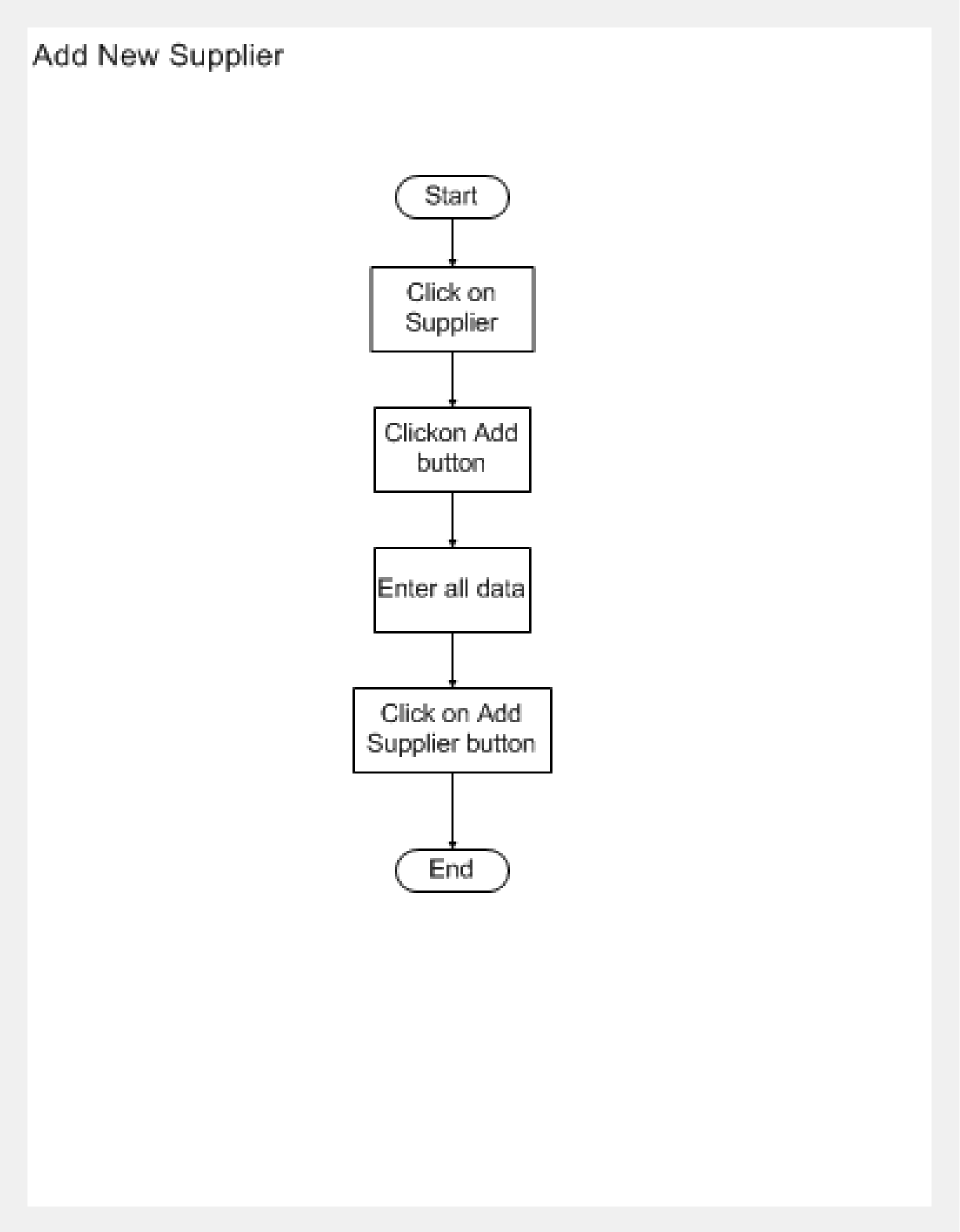
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.7 Perform Order


Add Order
Start
Click on Customer
Click on selected Order button
Select product type, search product, view price, enter quantity
Click on Add Card button
Check Quantity is available in stock
Not found
Show Suggested tablet with respect to same formula
Found
No
Yes
Click on Conform Order Button
View Summary
Click on print button
End

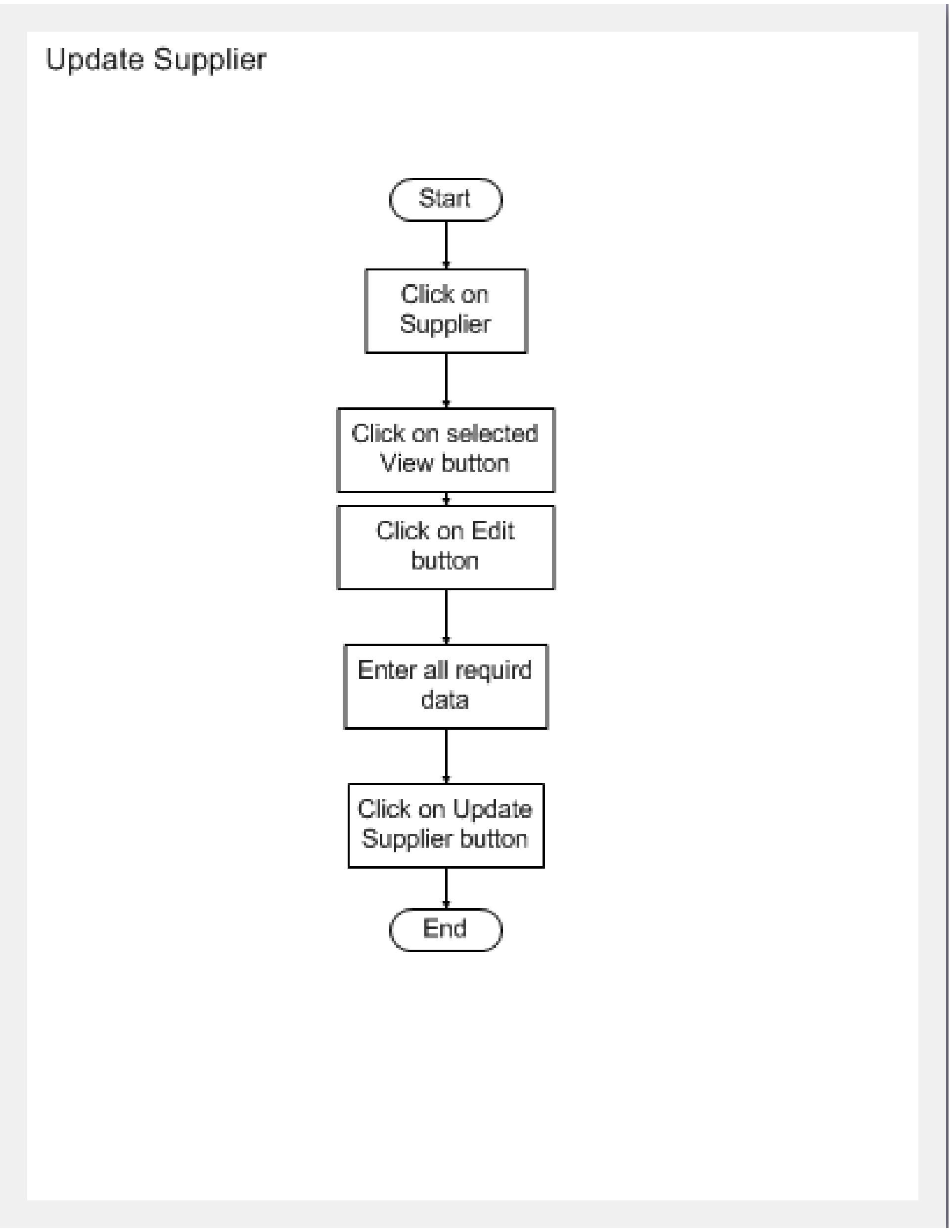
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.8 Supplier Add



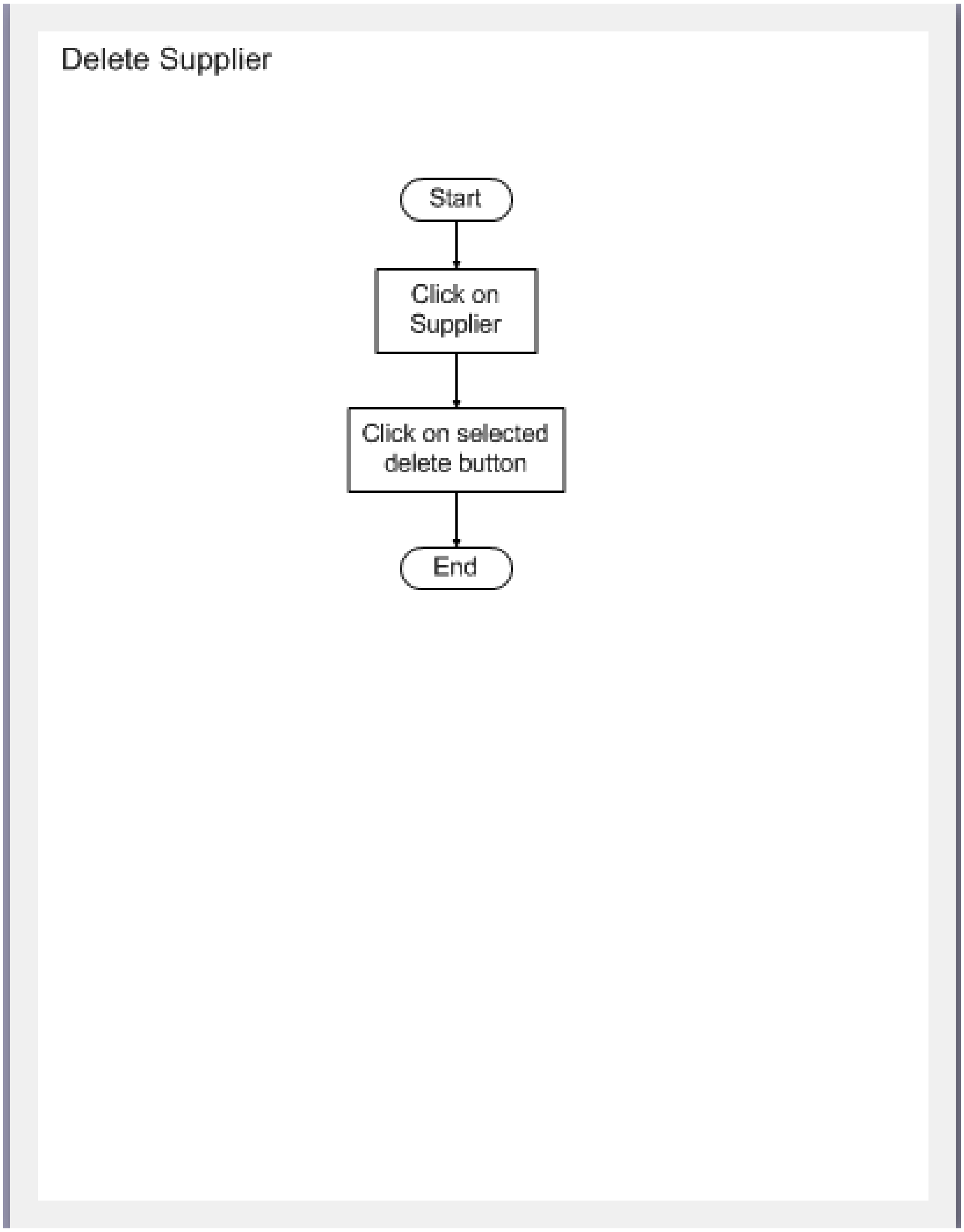
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.9 Supplier Update



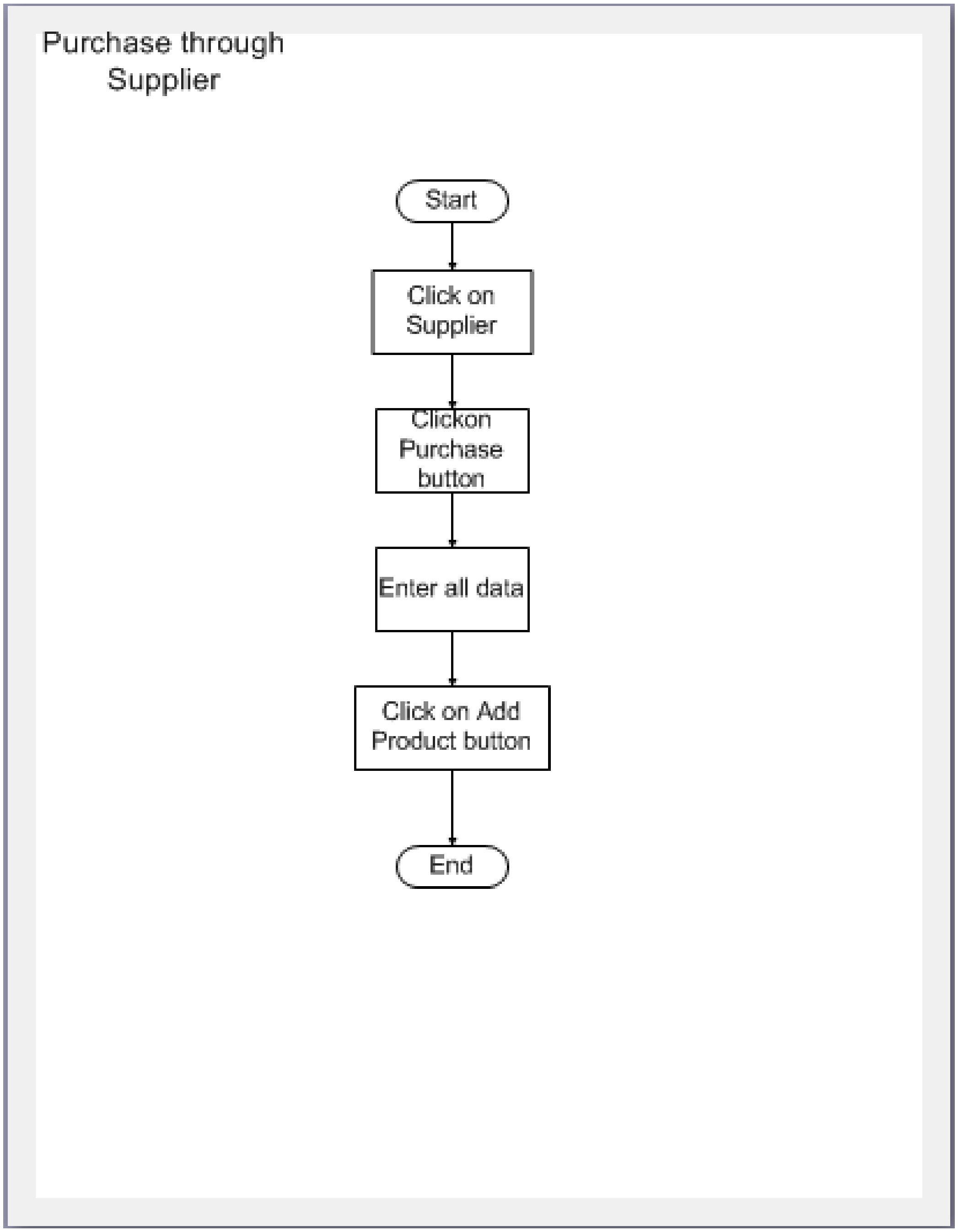
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.10 Supplier Delete



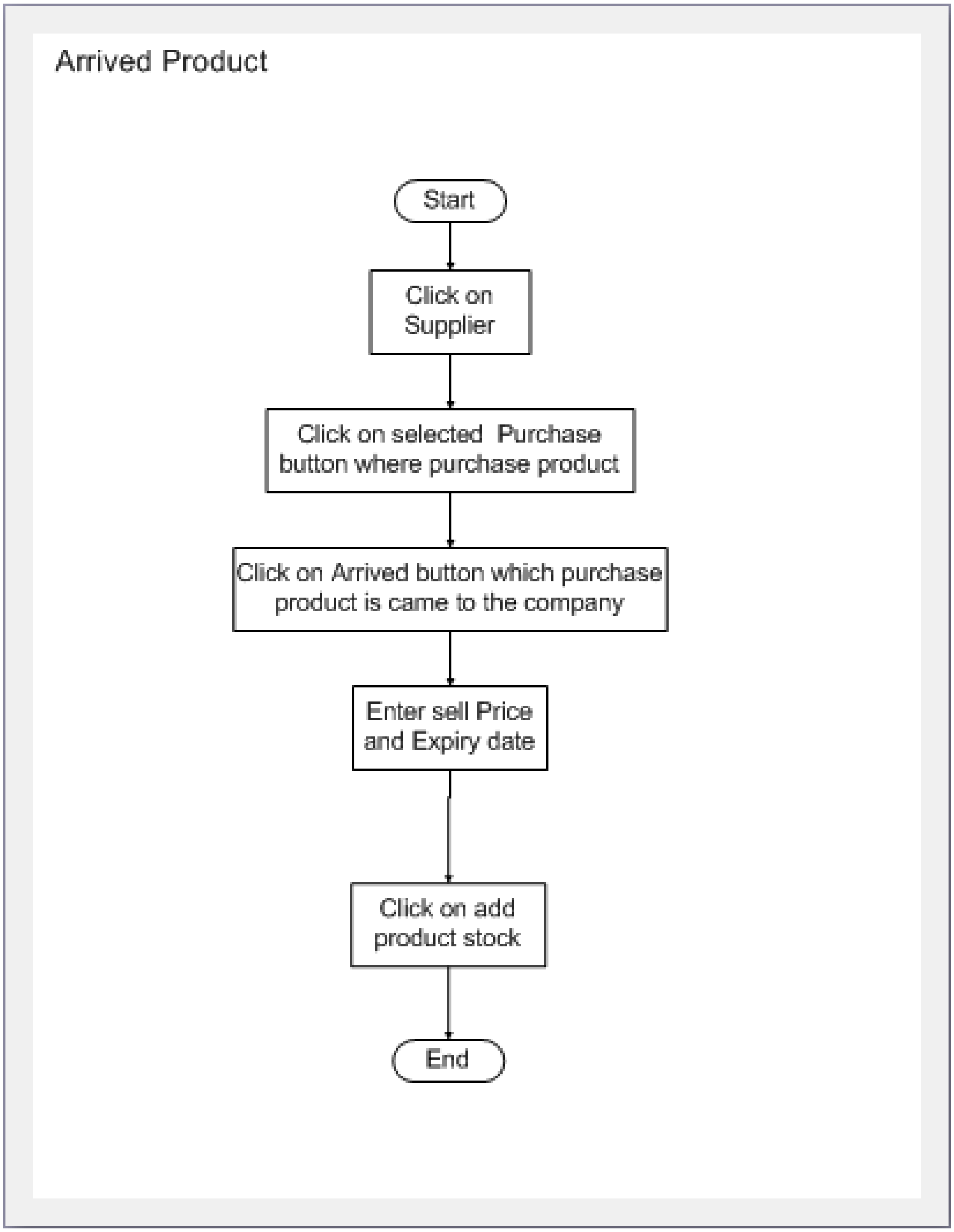
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.11 Purchase



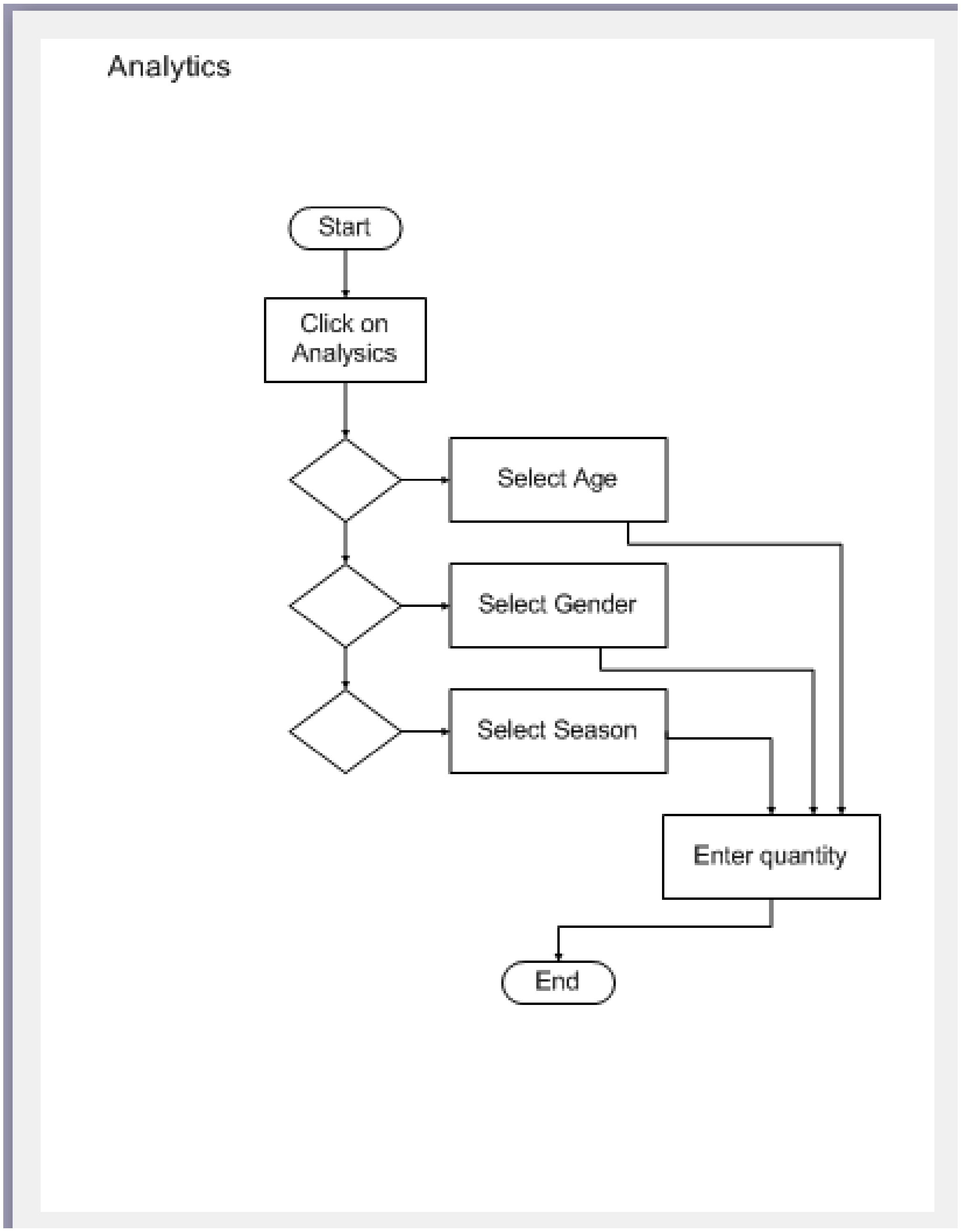
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.12 Arrived


Arrived Product
Start
Click on Supplier
Click on selected Purchase button where purchase product
Click on Arrived button which purchase product is came to the company
Enter sell Price and Expiry date
Click on add product stock
End

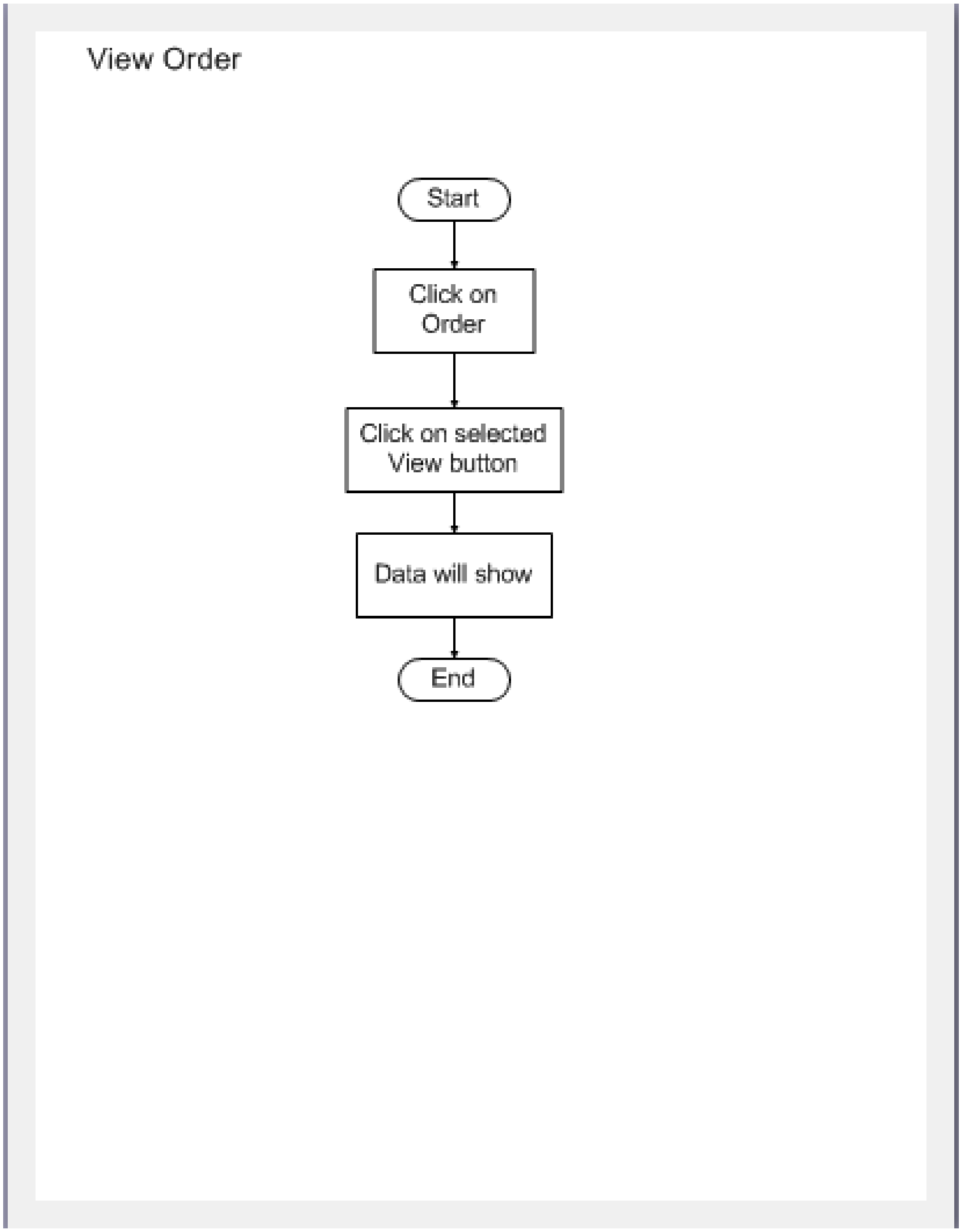
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.13 Analytics



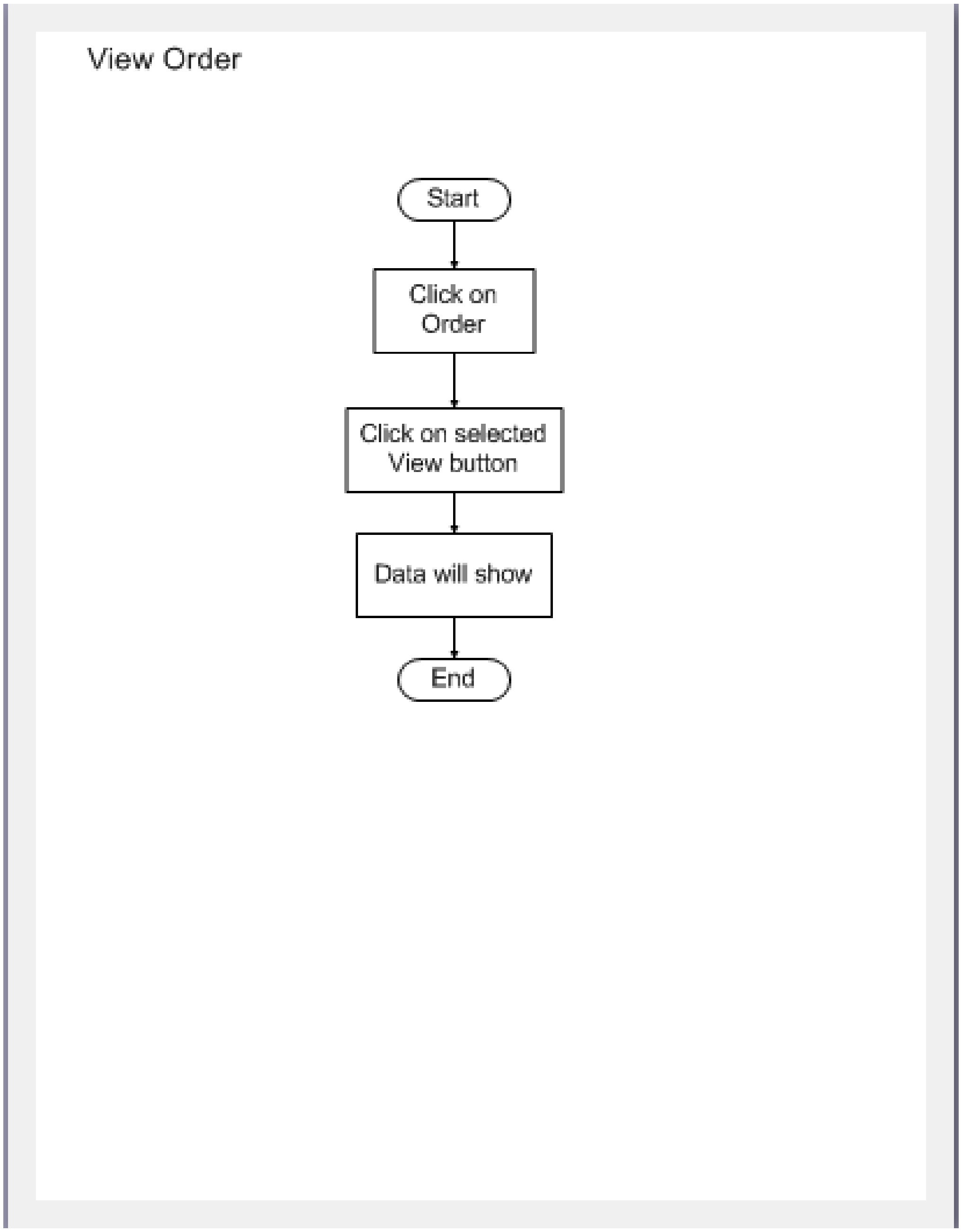
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.14 OfflineOrderView



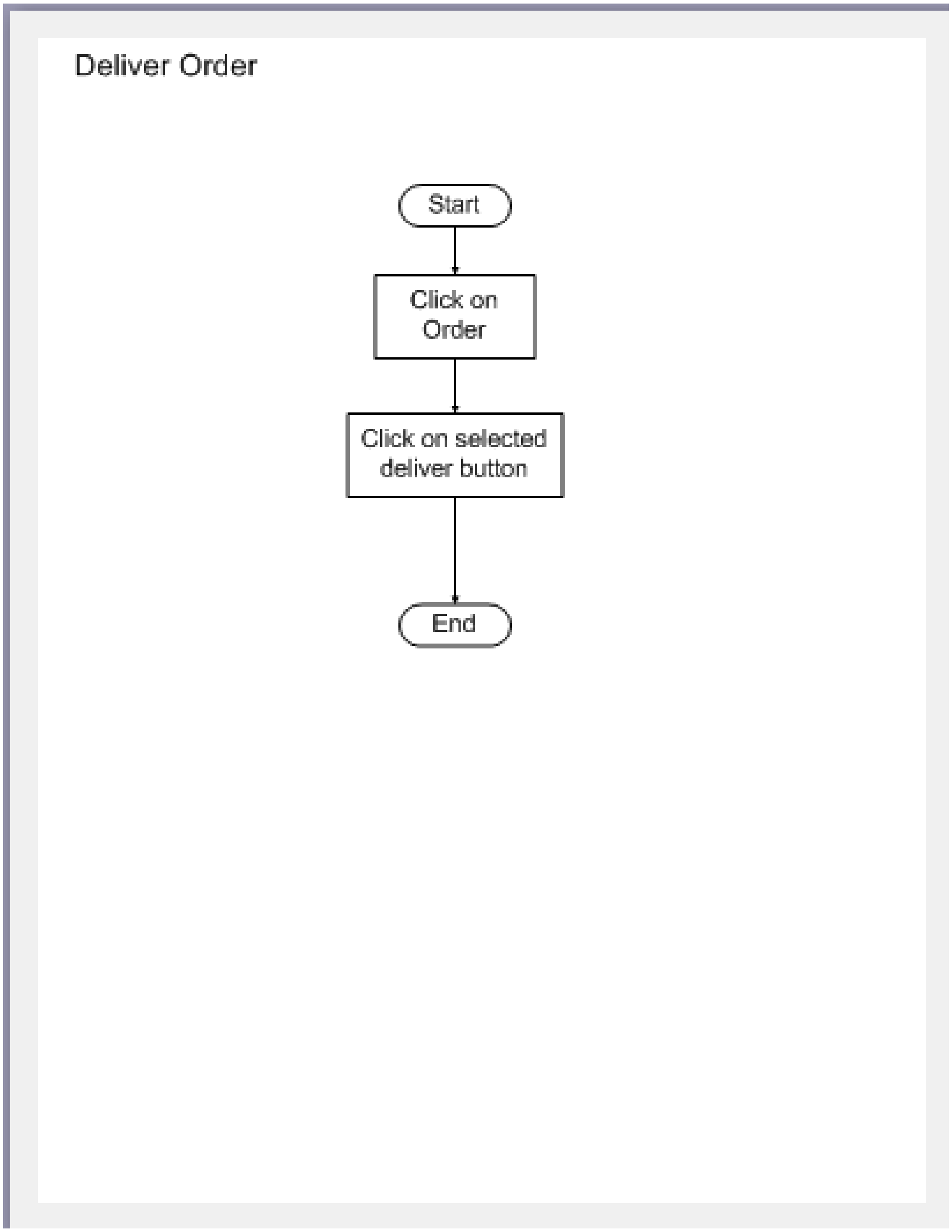
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.4.15 OnlineOrderView



This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

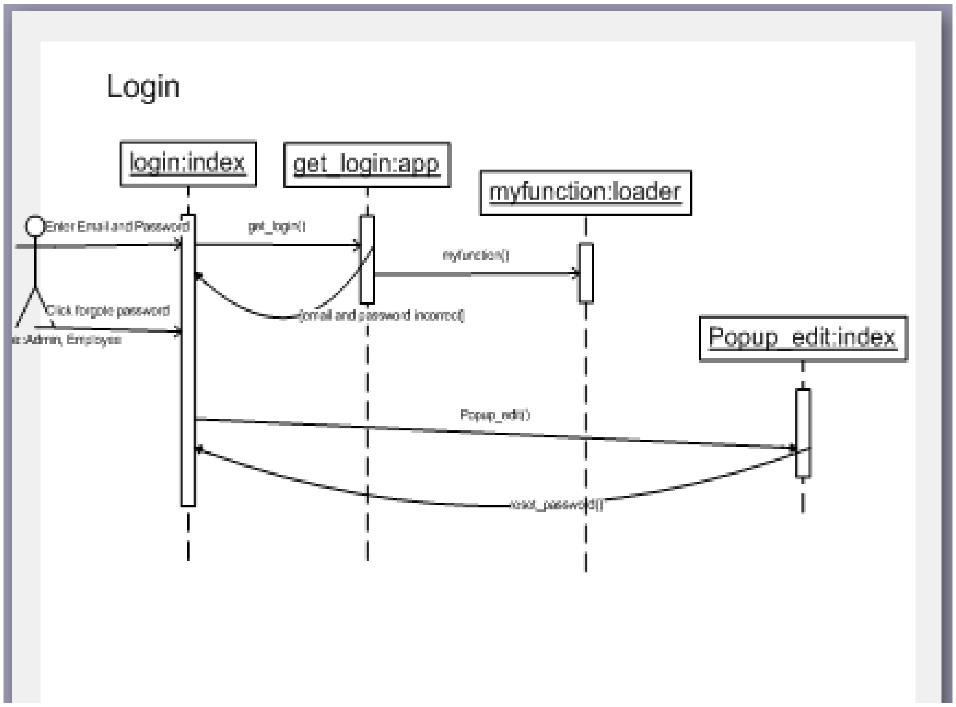
3.1.4.16 OnlineOrderDeliver



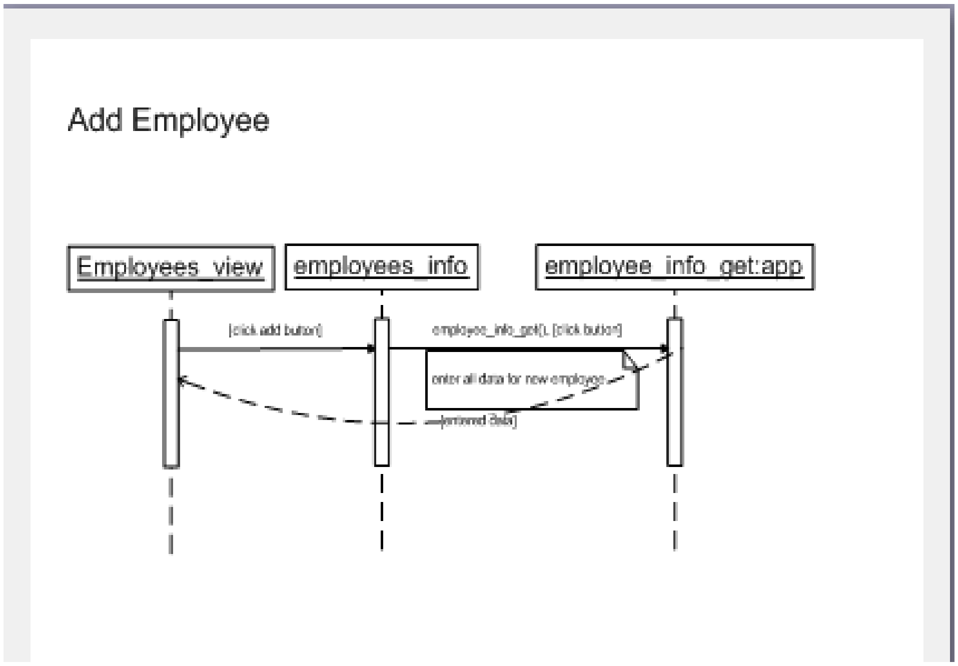
This represent the work flow or process. A flowchat can also be defined as a diagrammatic representation of an algorithm step by step approach to solving a task

3.1.5 Sequence Diagrams

3.1.5.1 Login

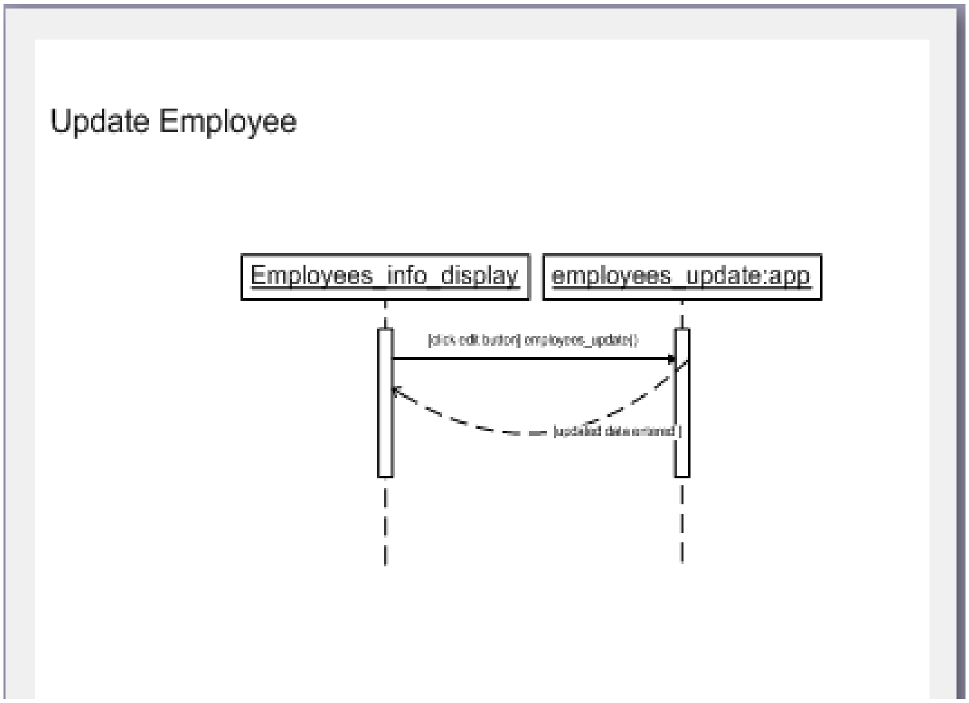


3.1.5.2 Employee Add

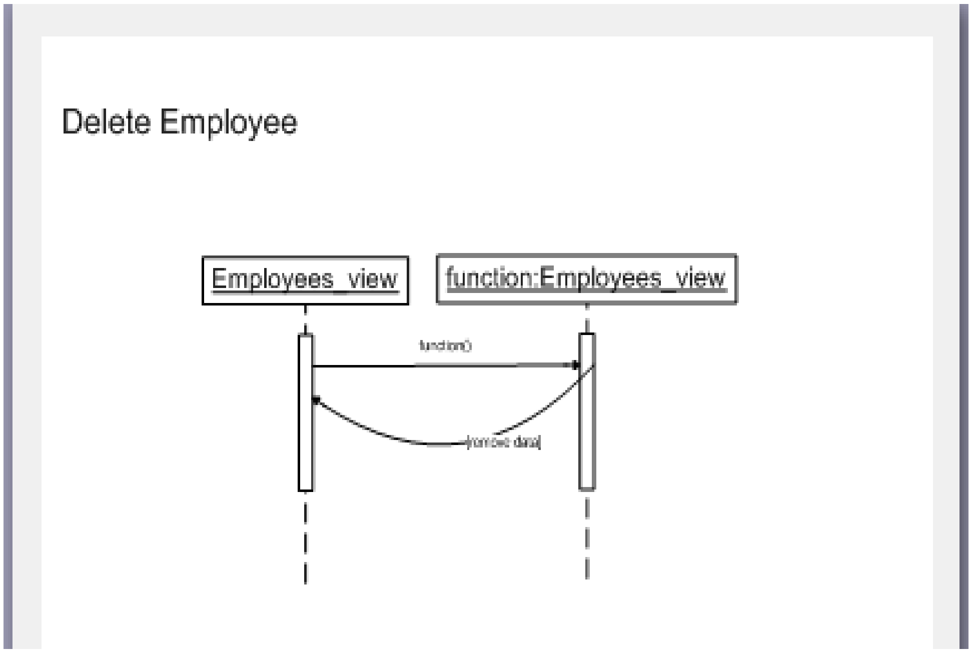


A sequence diagram shows object interactions arranged in time sequence. It depicts the object involved the scenario and the sequence of message exchange between the object.

3.1.5.3 Employee Update

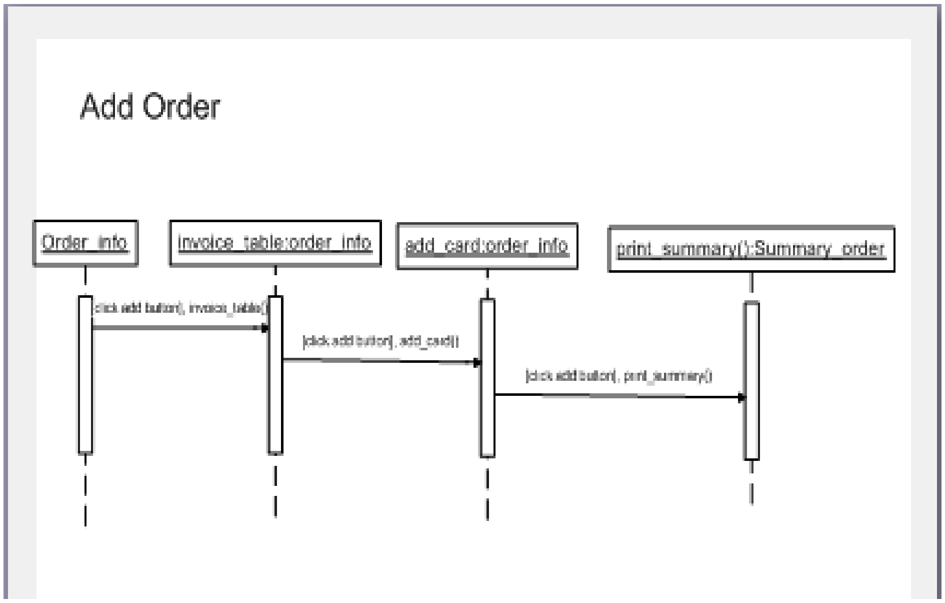


3.1.5.4 Employee Delete

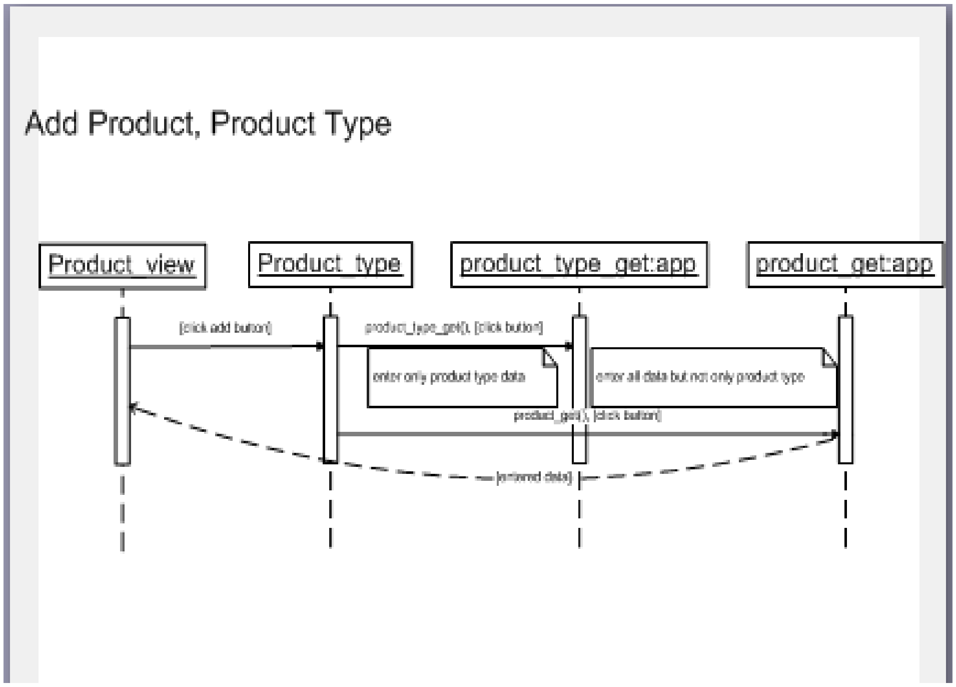


A sequence diagram shows object interactions arranged in time sequence. It depicts the object involved the scenario and the sequence of message exchange between the object.

3.1.5.5 Order Add

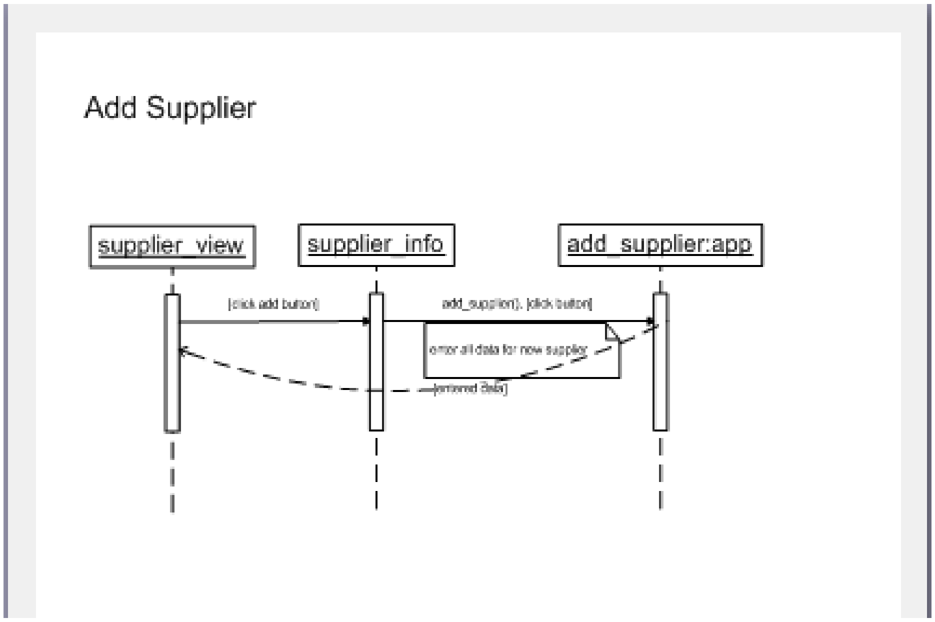

Add Order
Order_info
invoice_table:order_info
add_card:order_info
print_summary():Summary_order
[click add button], invoice_table()
[click add button], add_card()
[click add button], print_summary()

3.1.5.6 Product Add


Add Product, Product Type
Product_view
Product_type
product_type_get:app
product_get:app
[click add button]
product_type_get(), [click button]
enter only product type data
enter all data but not only product type
product_get(), [click button]
[entered data]

A sequence diagram shows object interactions arranged in time sequence. It depicts the object involved the scenario and the sequence of message exchange between the object.

3.1.5.7 Supplier Add



A sequence diagram shows object interactions arranged in time sequence. It depicts the object involved the scenario and the sequence of message exchange between the object.

3.2 Results

3.2.1 Overall Description

- This product is creating for professional Medical Store Because Pakistan haven’t a good
- Medical Store Management System which have all functionalities but in this have many function such as maintain inventory sales report stock report and others.
- The functionality of the product is to store large amounts of data uploaded by the admin

3.2.2 Product Functions

TeleMedicine Functions:

- Create Employees for performing work, add users, supplier, Product.area
- Update Employee, Supplier, Product, Customer.
- Delete Employee, Customer, Supplier, Product, Order, Stock.
- Purchase Tablets/Syrup/Capsule/etc.
- Print Invoice
- Stock report

- Sales report
- If any medicine is not available in stock so suggest other medicine with respect to formula
- Full detailed/Description of Any Product
- Forgot password by email using authentication of firebase

3.2.3 User Classes and Characteristics

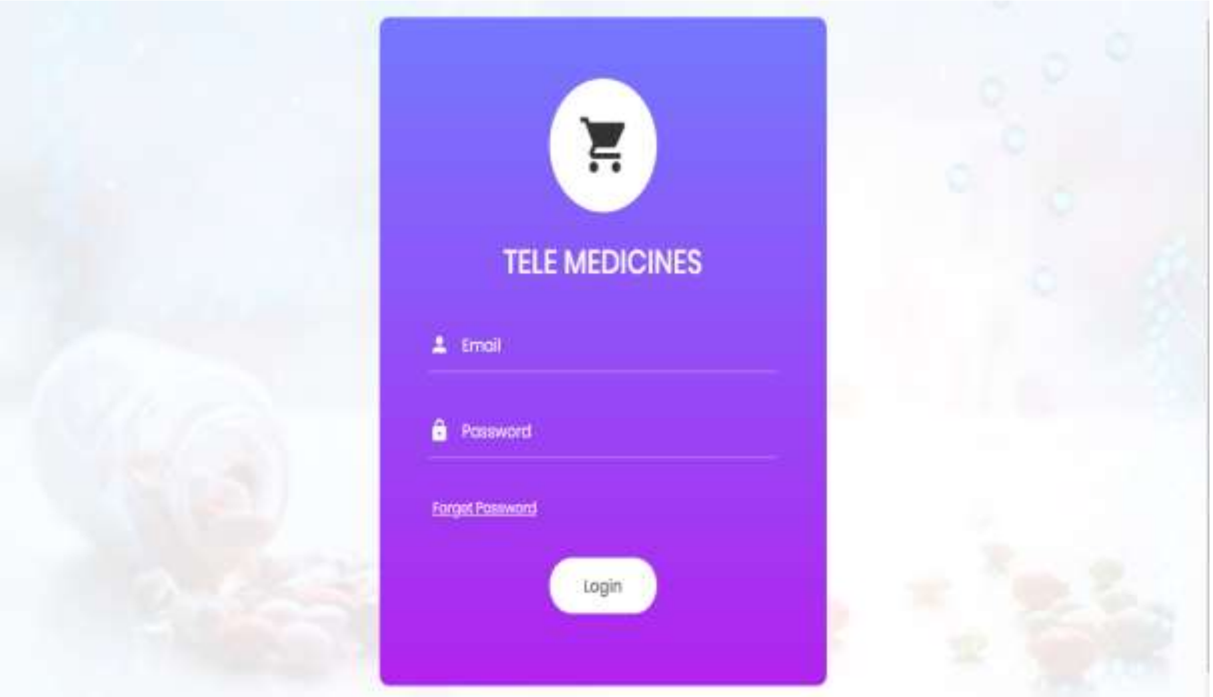
There are 3 basic user classes involving multiple roles.

- One is for Admin who has more rights than any other user and can insert/modify any kind of data. Such as supplier, Stock and Product. They have the best technical expertise and the highest privilege levels. They are the all type of rights.
- Other is Employee who have only some right Such as add a customer, perform order, view order that is only the thing, there is no other rights who have admin just because of security reasons.
- Third user class is the one who have only the right which is perform Order.

3.3 Features Overview

The features included in TeleMedicine are below.

3.3.1 Login Screen

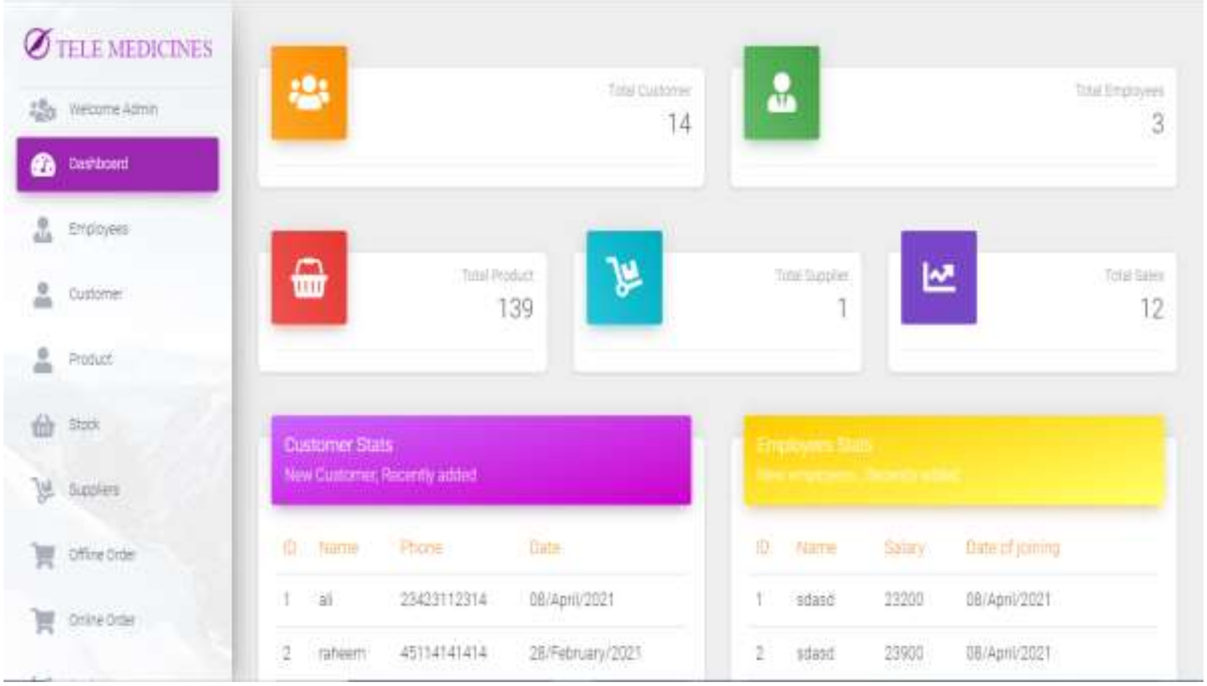


3.3.2 Created Modules

The modular structure dictated that the features be split up according to users and their privileges, resulting in a more secure ad robust system which restricts unauthorized accessibility and at the same time provides the necessary functionality as needed by the user. The two following modules were created on web.

3.3.2.1 Administrative Module

Screen: Administrator Dashboard



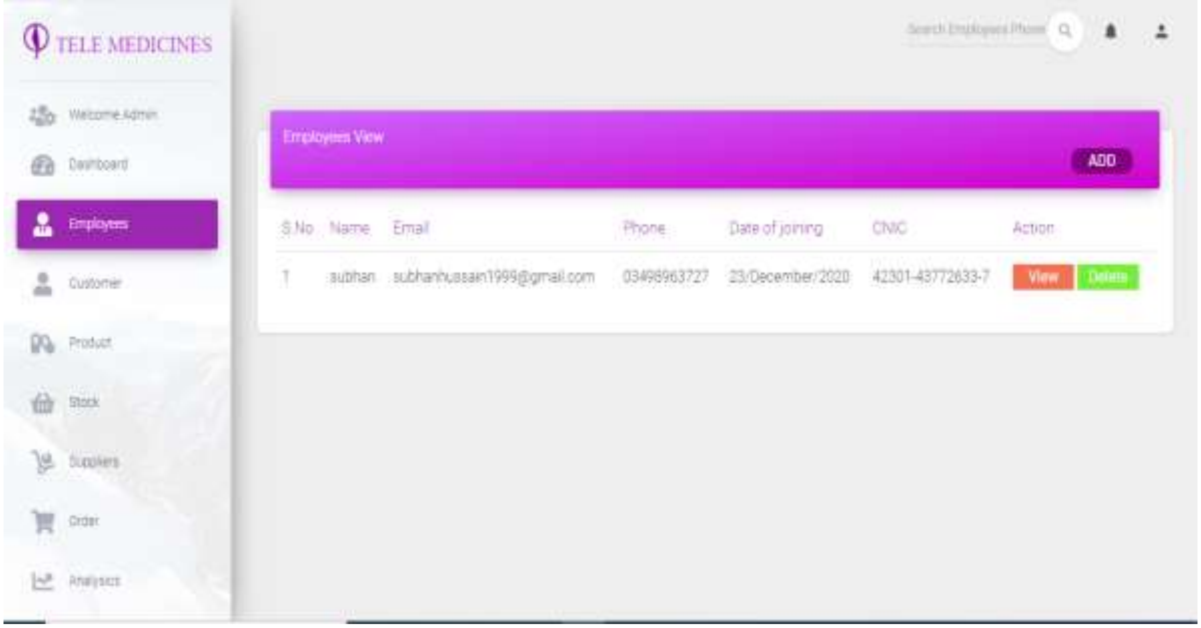
Customer Stats
New Customer, Recently added

ID	Name	Phone	Date
1	ali	23423112314	08/April/2021
2	raheem	45114141414	28/February/2021

Employees Stats
New employees, Recently added

ID	Name	Salary	Date of joining
1	sdasd	23200	08/April/2021
2	sdasd	23900	08/April/2021

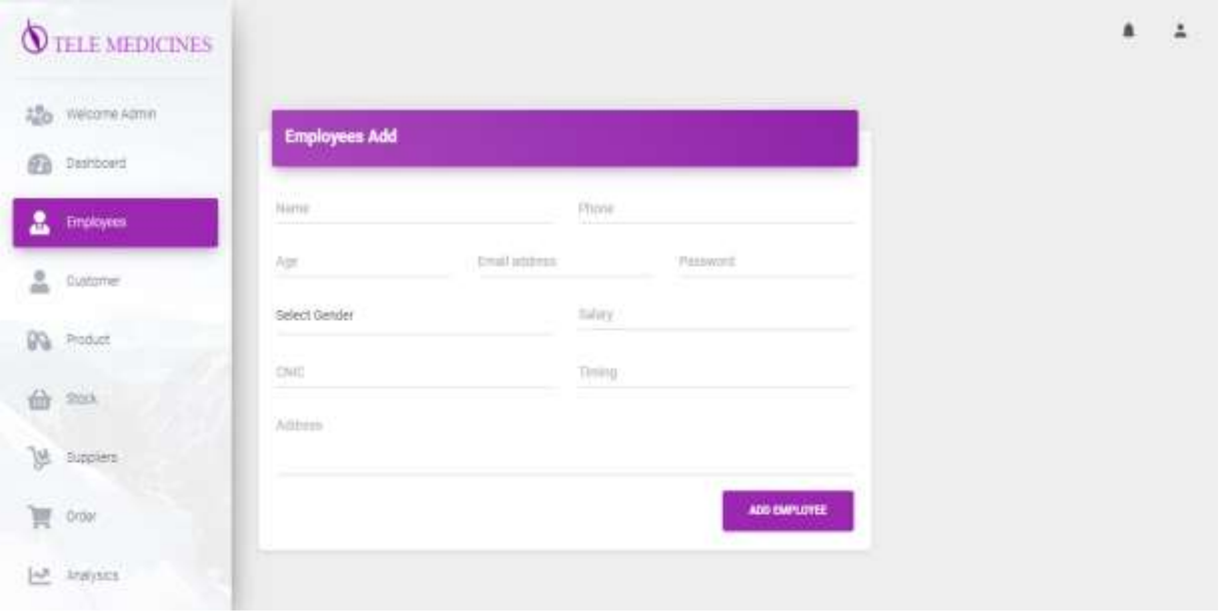
Screen: All Employees



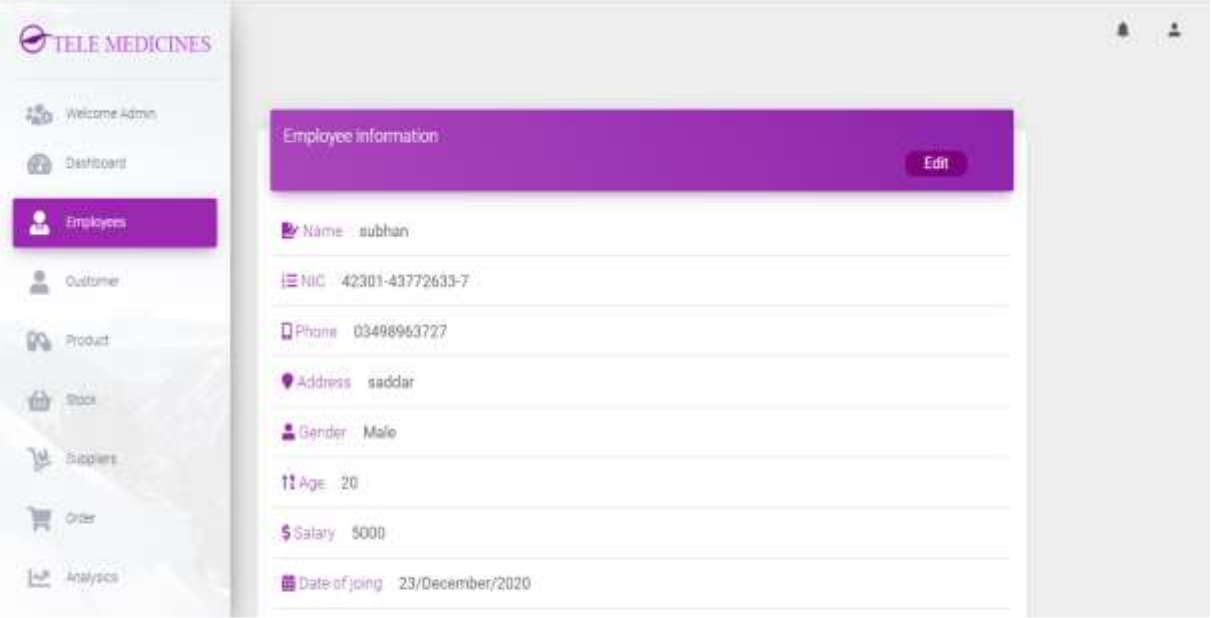
Employees View

S.No	Name	Email	Phone	Date of joining	CNIC	Action
1	subhan	subhanhussain1999@gmail.com	03498963727	23/December/2020	42301-43772633-7	View Delete

Screen: Create Employee



Screen: View Employee



Name subhan

NIC 42301-43772633-7

Phone 03498963727

Address saddar

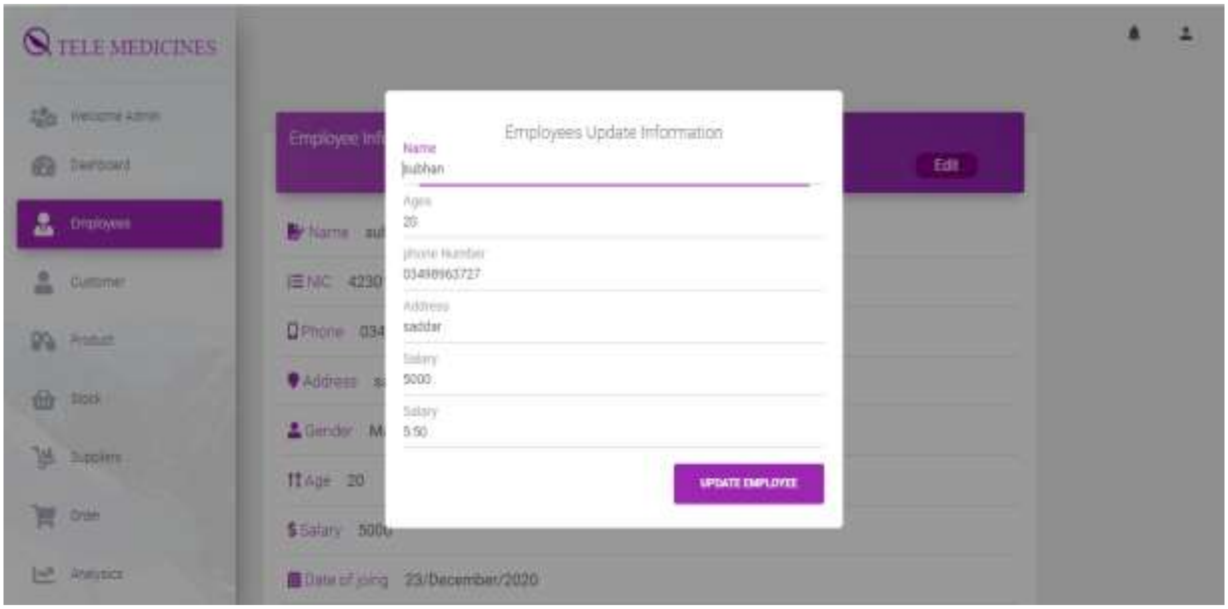
Gender Male

Age 20

Salary 5000

Date of joing 23/December/2020

Screen: Update Employee



Employees Update Information

Name
subhan

Ages
20

phone Number
03498963727

Address
saddar

Salary
5000

Salary
5:50

Name sul

NIC 4230

Phone 034

Address s

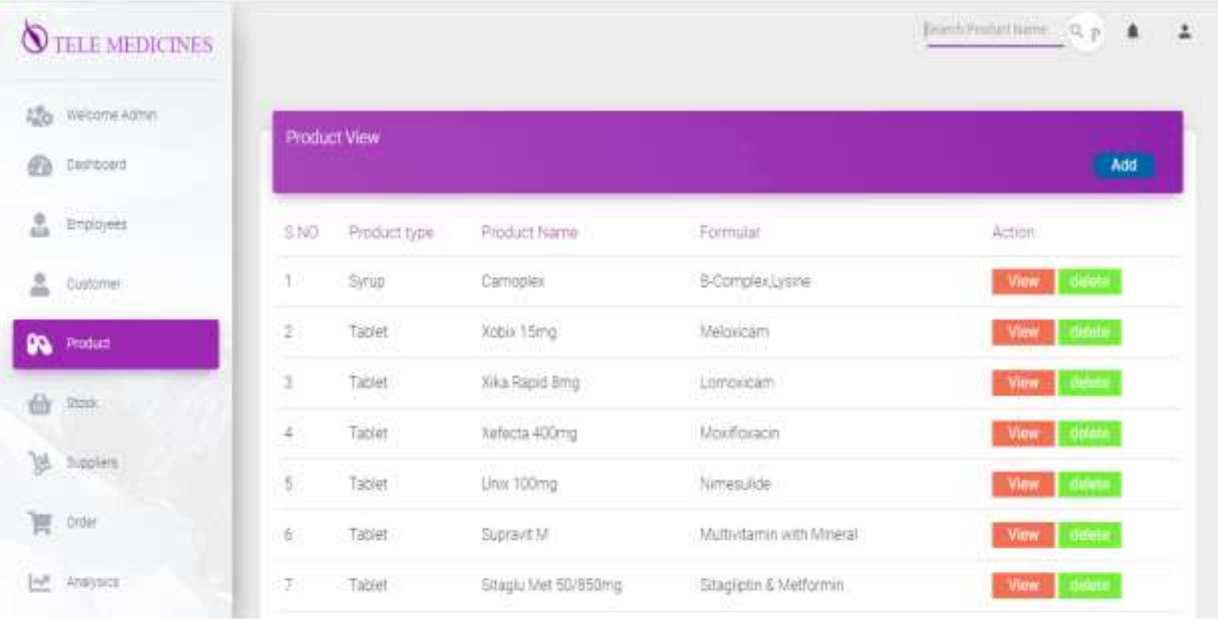
Gender M

Age 20

Salary 500

Date of joing 23/December/2020

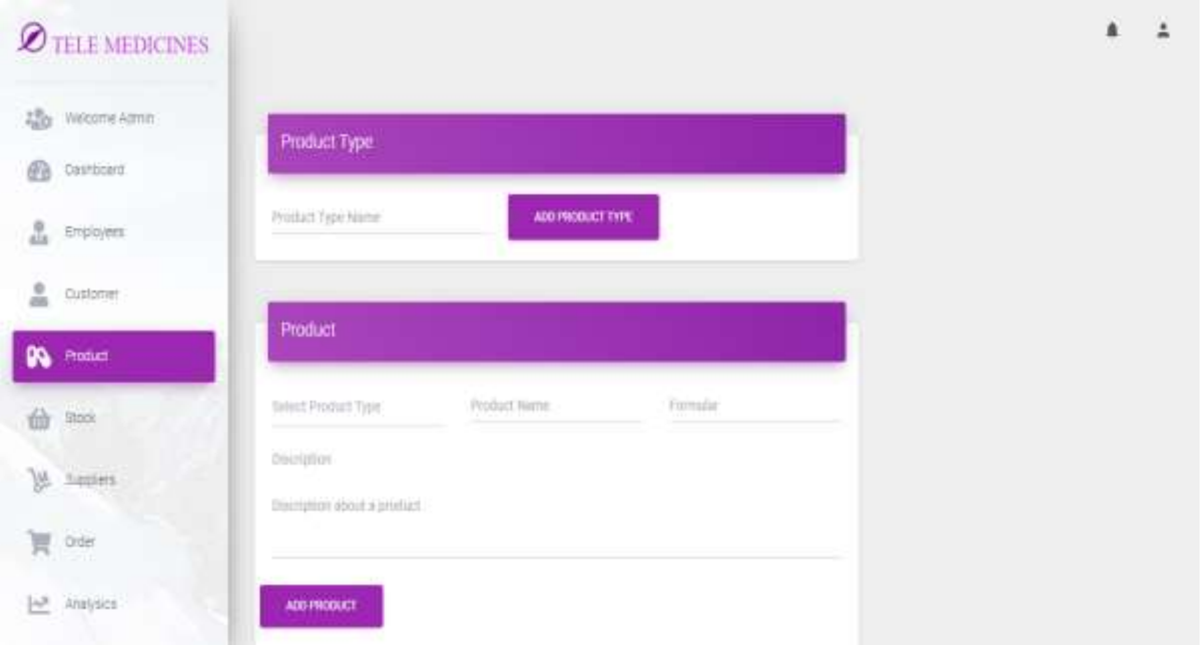
Screen: All Product



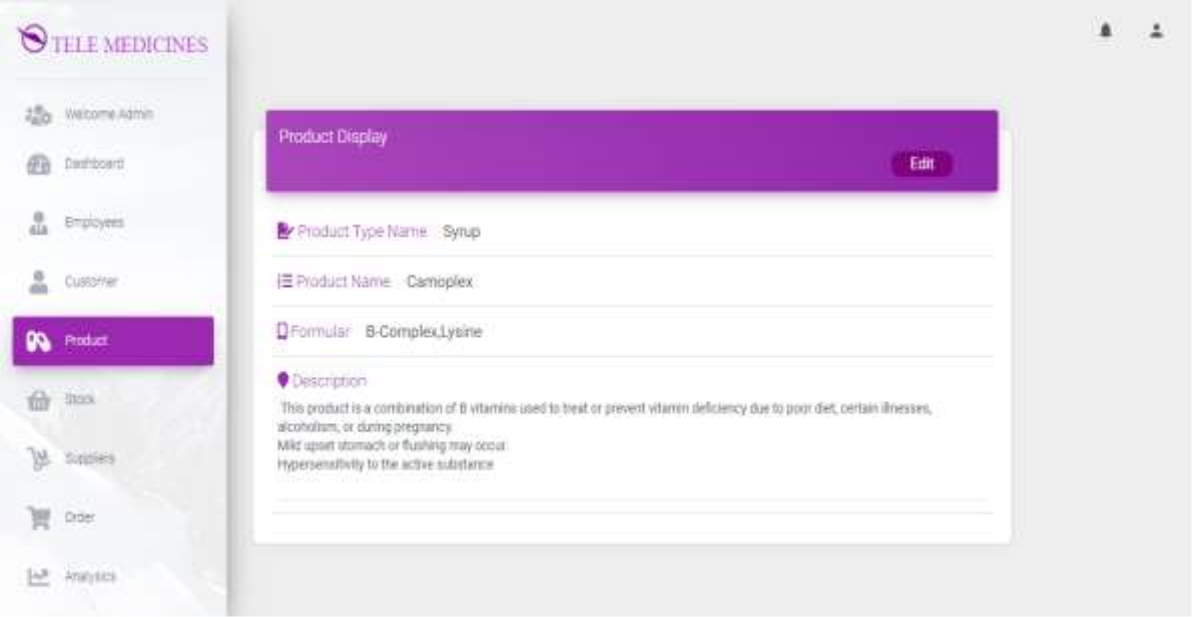
Product View

S.NO	Product type	Product Name	Formular	Action
1	Syrup	Camoplex	B-Complex,Lysine	View delete
2	Tablet	Xobix 15mg	Meloxicam	View delete
3	Tablet	Xika Rapid 8mg	Lornoxicam	View delete
4	Tablet	Xefecta 400mg	Moxifloxacin	View delete
5	Tablet	Unix 100mg	Nimesulide	View delete
6	Tablet	Supravit M	Multivitamin with Mineral	View delete
7	Tablet	Sitaglu Met 50/850mg	Sitagliptin & Metformin	View delete

Screen: Add Product



Screen: View Product



Product Type Name Syrup

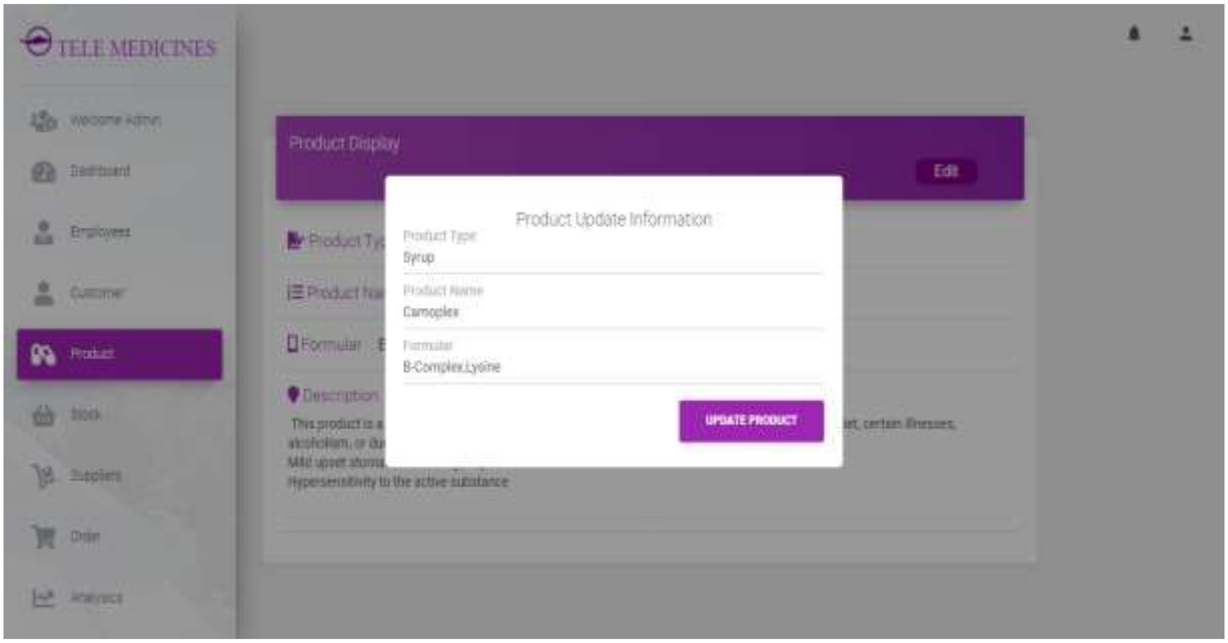
Product Name Camoplex

Formular B-Complex,Lysine

Description

This product is a combination of B vitamins used to treat or prevent vitamin deficiency due to poor diet, certain illnesses, alcoholism, or during pregnancy.
Mild upset stomach or flushing may occur.
Hypersensitivity to the active substance

Screen: Update Product



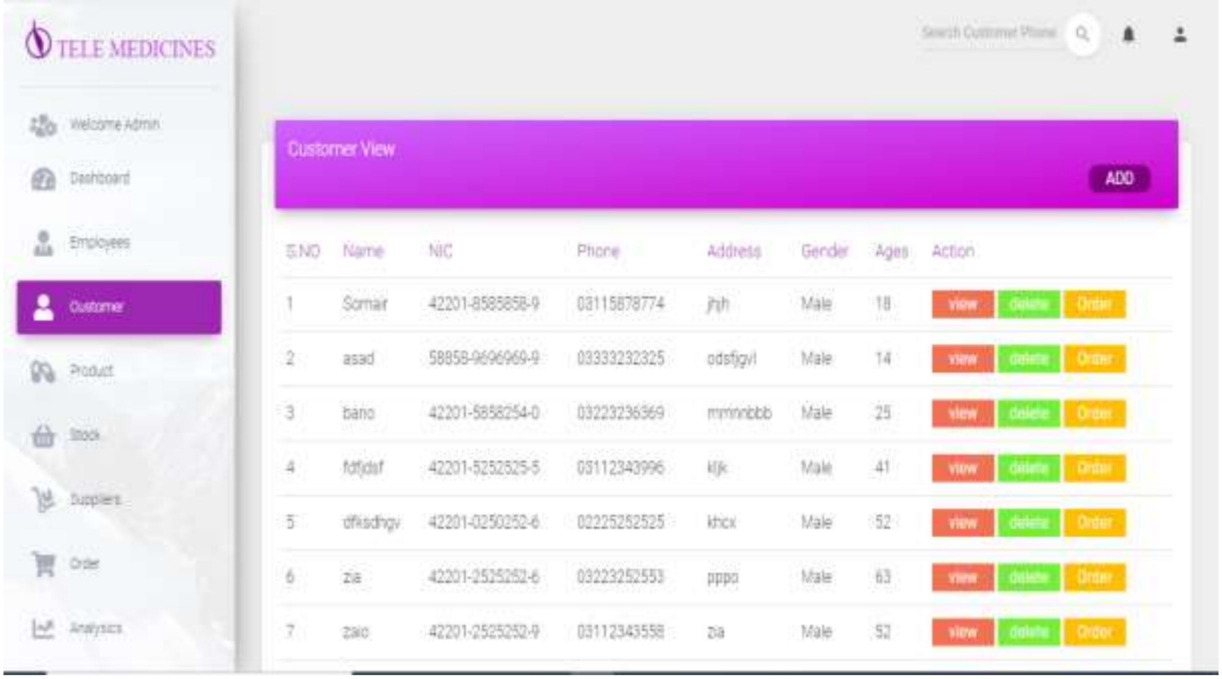
Product Update Information

Product Type
Syrup

Product Name
Camoplex

Formular
B-Complex,Lysine

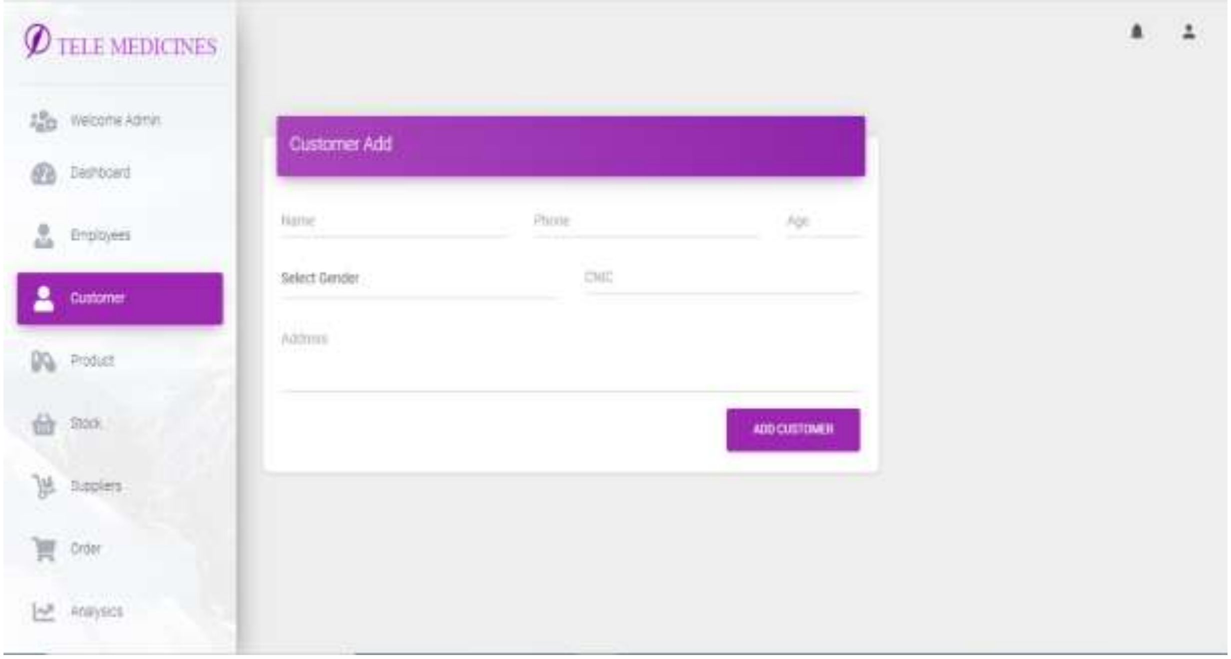
Screen: All Customer



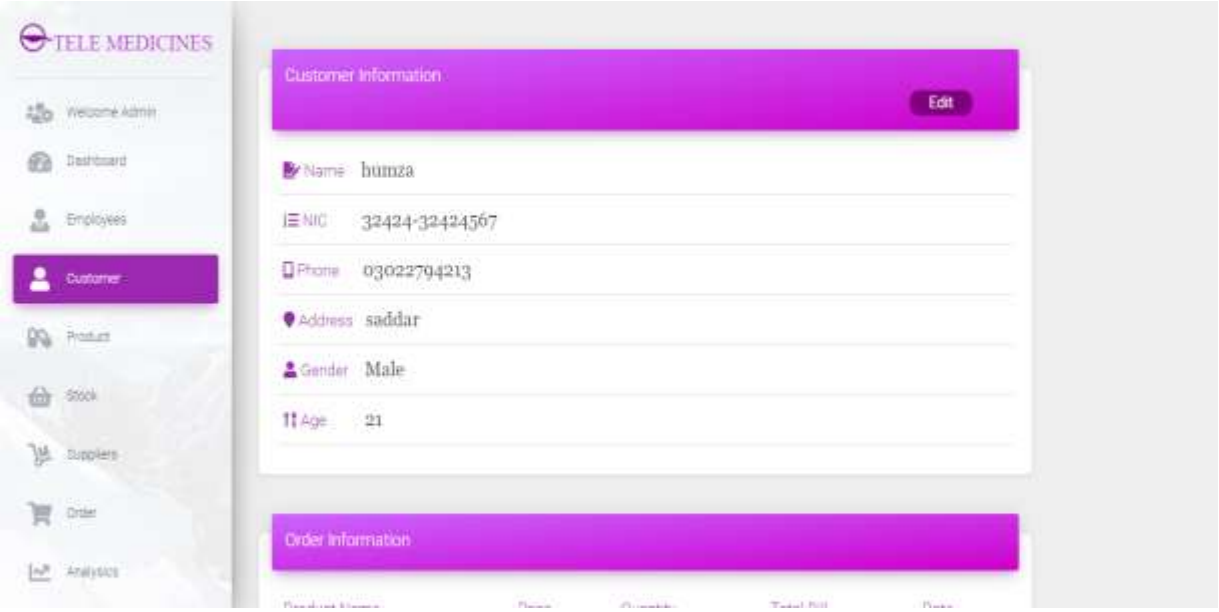
Customer View

S.NO	Name	NIC	Phone	Address	Gender	Ages	Action
1	Somair	42201-8585858-9	03115878774	jhjh	Male	18	view delete Order
2	asad	58858-9696969-9	03333232325	odsfjgvl	Male	14	view delete Order
3	bano	42201-5858254-0	03223236369	mmnnbbb	Male	25	view delete Order
4	fdfjdsf	42201-5252525-5	03112343996	kljk	Male	41	view delete Order
5	dfksdhgv	42201-0250252-6	02225252525	khcx	Male	52	view delete Order
6	zia	42201-2525252-6	03223252553	pppo	Male	63	view delete Order
7	zaio	42201-2525252-9	03112343558	zia	Male	52	view delete Order

Screen: Add Customer



Screen: View Customer



Name humza

NIC 32424-32424567

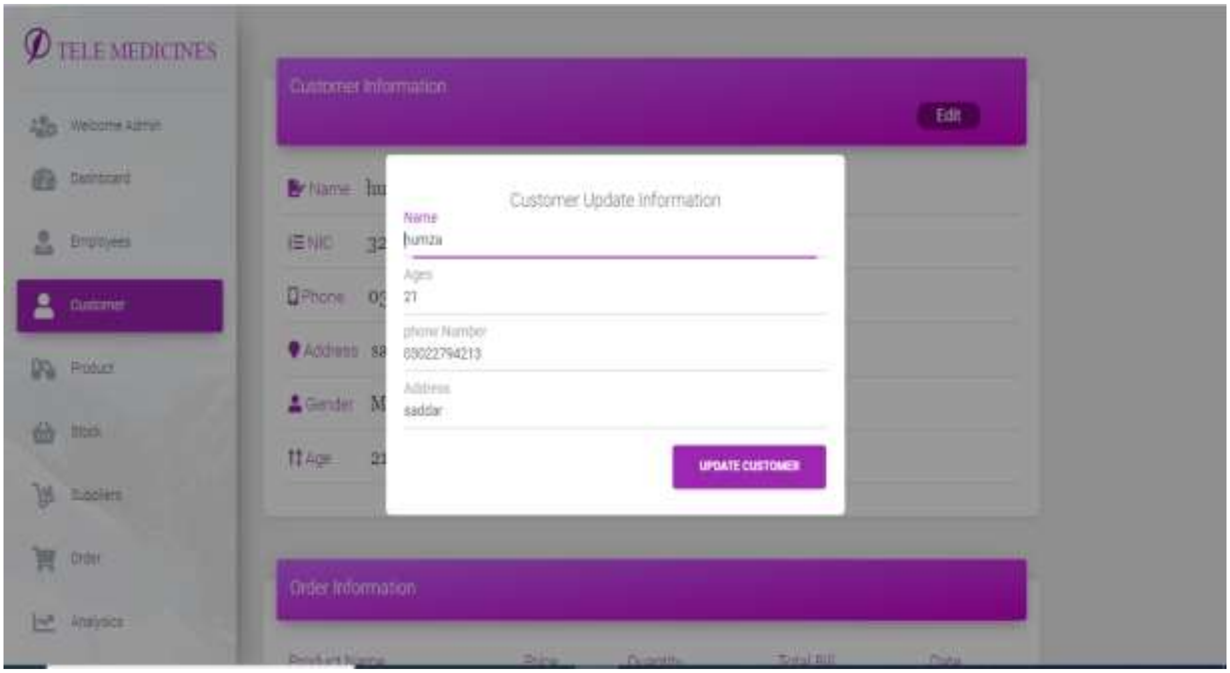
Phone 03022794213

Address saddar

Gender Male

Age 21

Screen: Update Customer



Name hu
NIC 32
Phone 03
Address 88
Gender M
Age 21

Customer Update Information

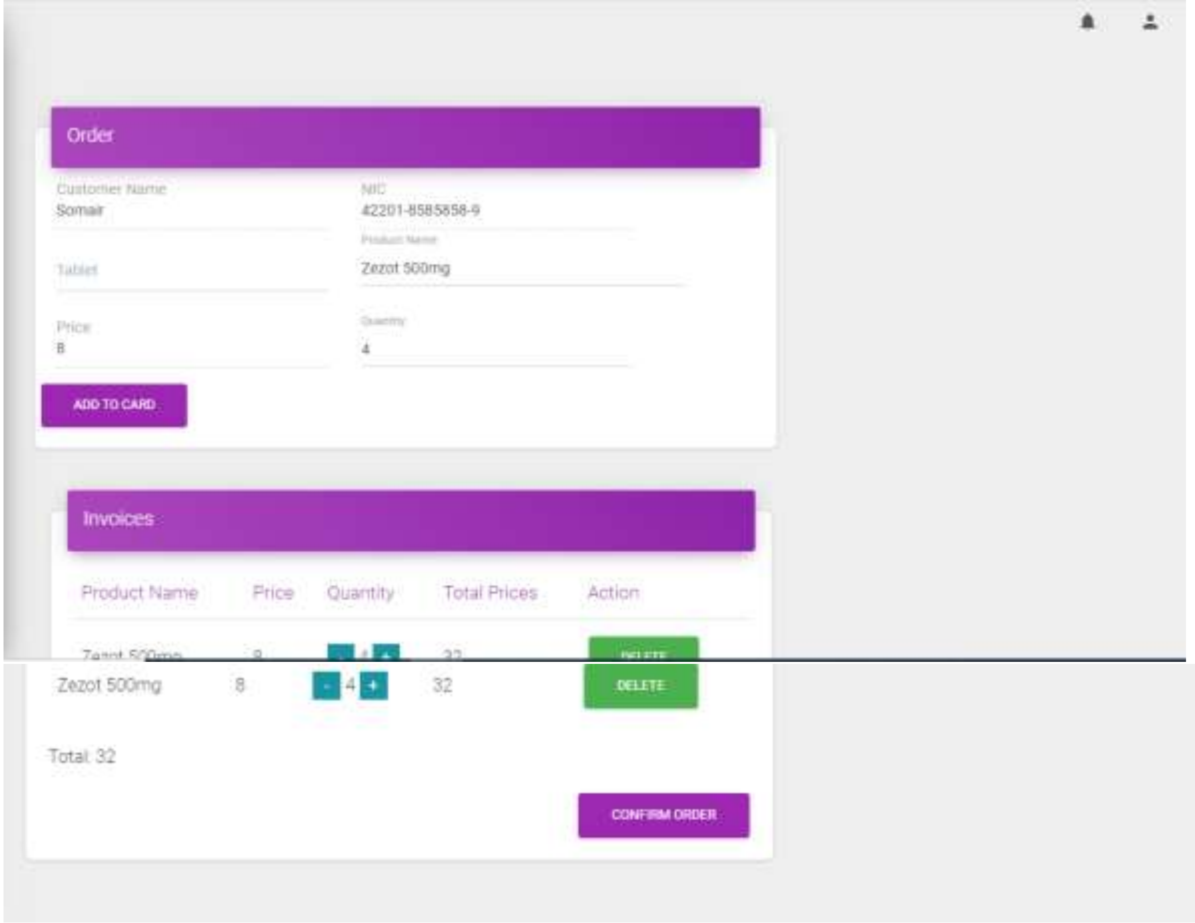
Name
humza

Ages
21

phone Number
03022794213

Address
saddar

Screen: Add Order



Customer Name
Somair

NIC
42201-8585858-9

Tablet

Product Name
Zezot 500mg

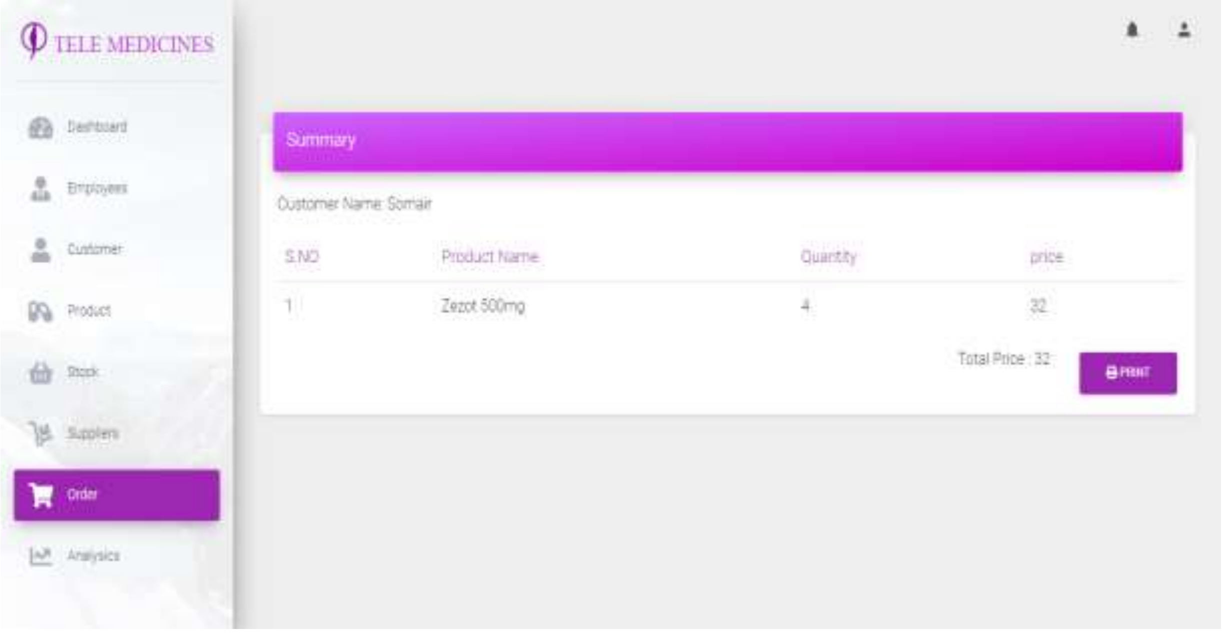
Price
8

Quantity
4

Product Name	Price	Quantity	Total Prices	Action
Zezot 500mg	8	- 4 +	32	DELETE

Total: 32

Screen: Conform Order



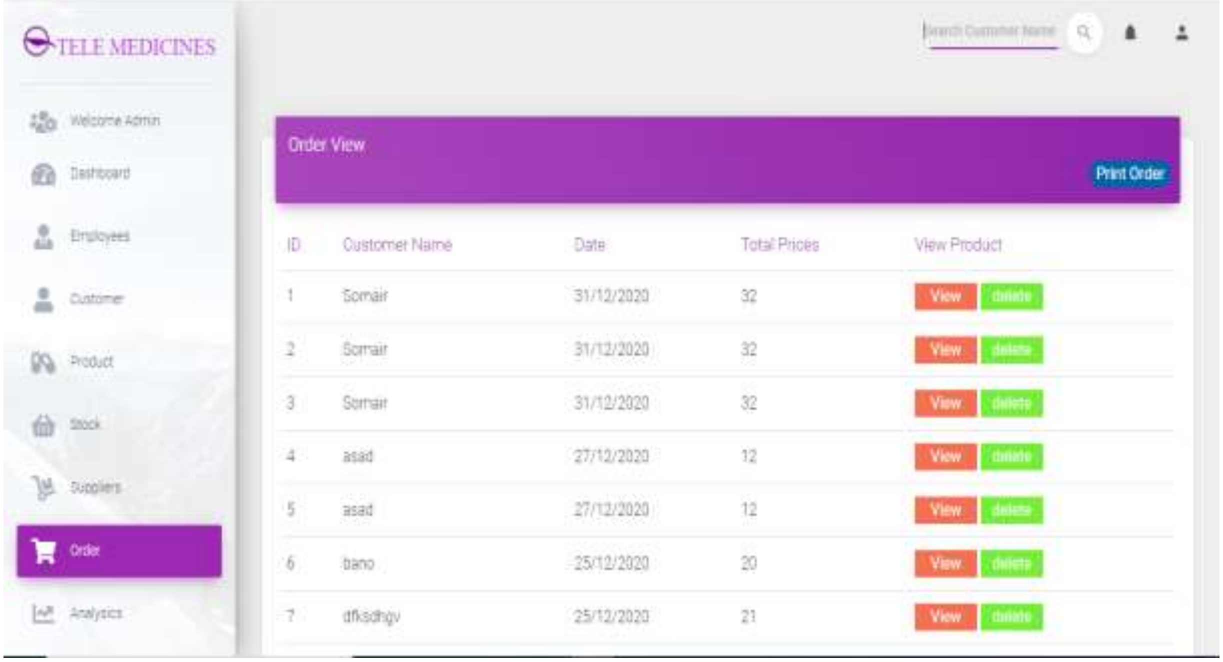
Summary

Customer Name: Somair

S.NO	Product Name	Quantity	price
1	Zezot 500mg	4	32

Total Price : 32

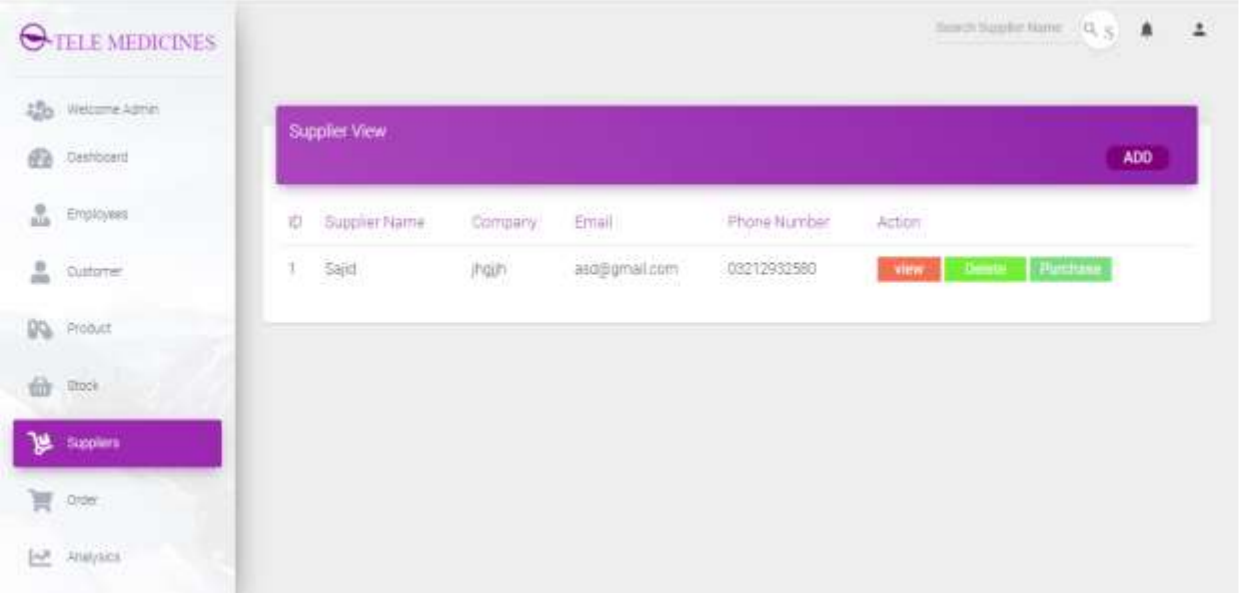
Screen: Print Sales Order



Order View

ID	Customer Name	Date	Total Prices	View Product
1	Somair	31/12/2020	32	View delete
2	Somair	31/12/2020	32	View delete
3	Somair	31/12/2020	32	View delete
4	asad	27/12/2020	12	View delete
5	asad	27/12/2020	12	View delete
6	bano	25/12/2020	20	View delete
7	dfksdhgv	25/12/2020	21	View delete

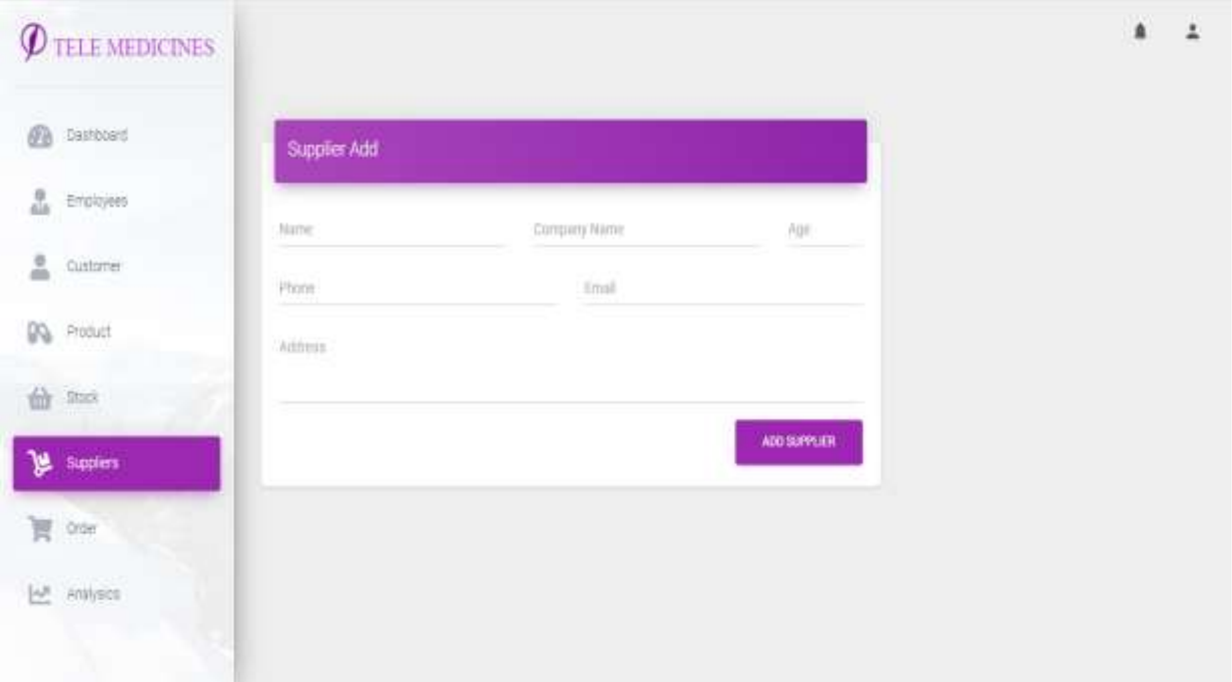
Screen: All Supplier



Supplier View

ID	Supplier Name	Company	Email	Phone Number	Action
1	Sajid	jhgjh	asd@gmail.com	03212932580	view Delete Purchase

Screen: Add Supplier



Supplier Add

Name

Company Name

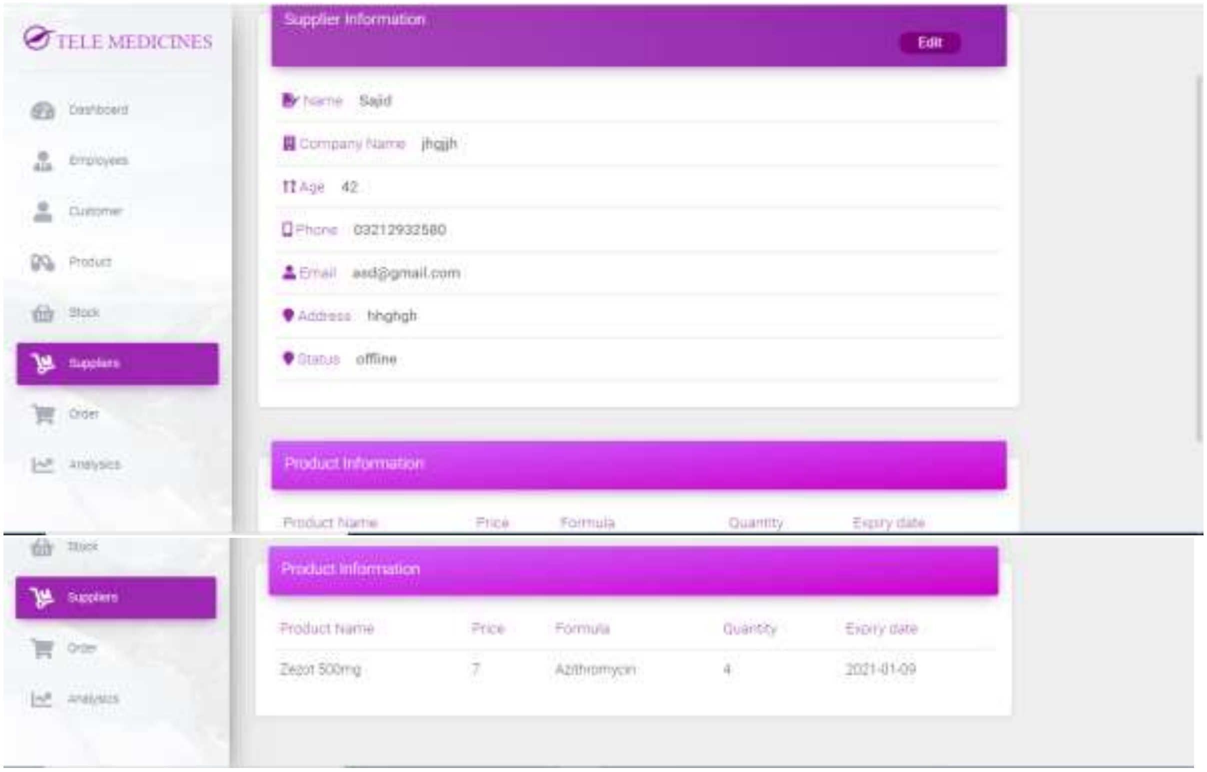
Age

Phone

Email

Address

Screen: View Supplier



Supplier Information

Name Sajid

Company Name jhgjh

Age 42

Phone 03212932580

Email asd@gmail.com

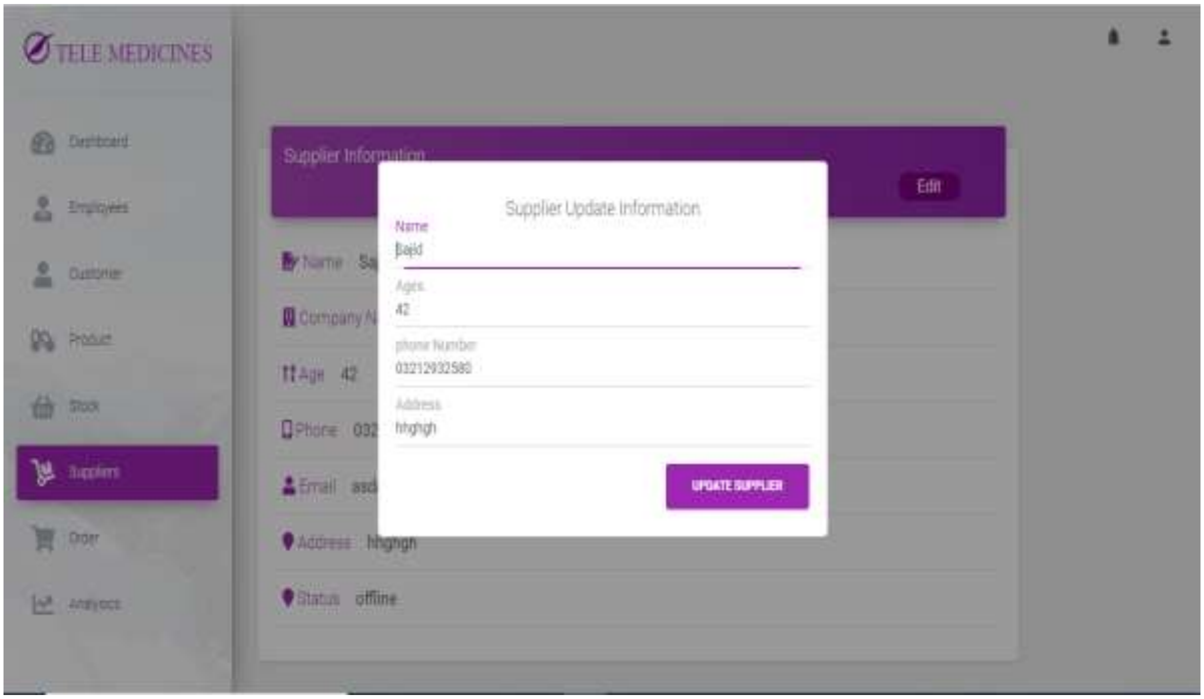
Address hhghgh

Status offline

Product Information

Product Name	Price	Formula	Quantity	Expiry date
Zezot 500mg	7	Azithromycin	4	2021-01-09

Screen: Update Supplier



Supplier Update Information

Name
Sajid

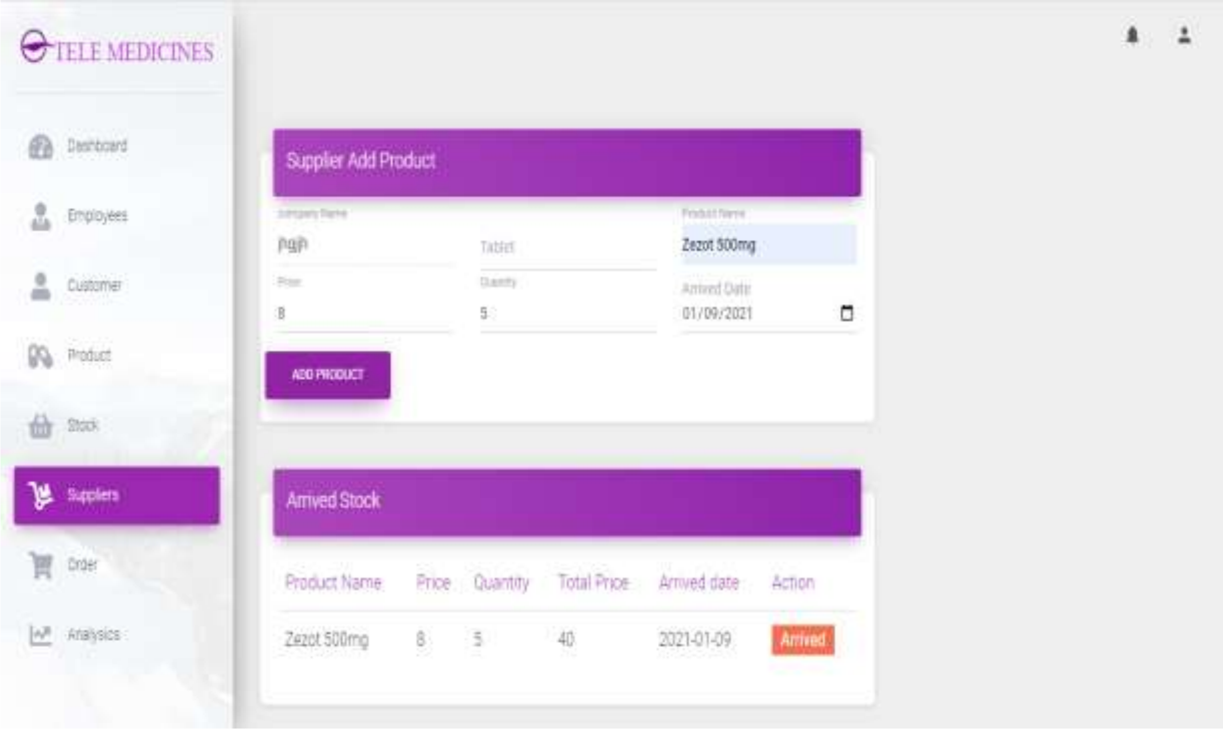
Ages
42

phone Number
03212932580

Address
hhghgh

UPDATE SUPPLIER

Screen: Purchase Product



Supplier Add Product

Company Name: jhgjh

Tablet

Product Name: Zezot 500mg

Price: 8

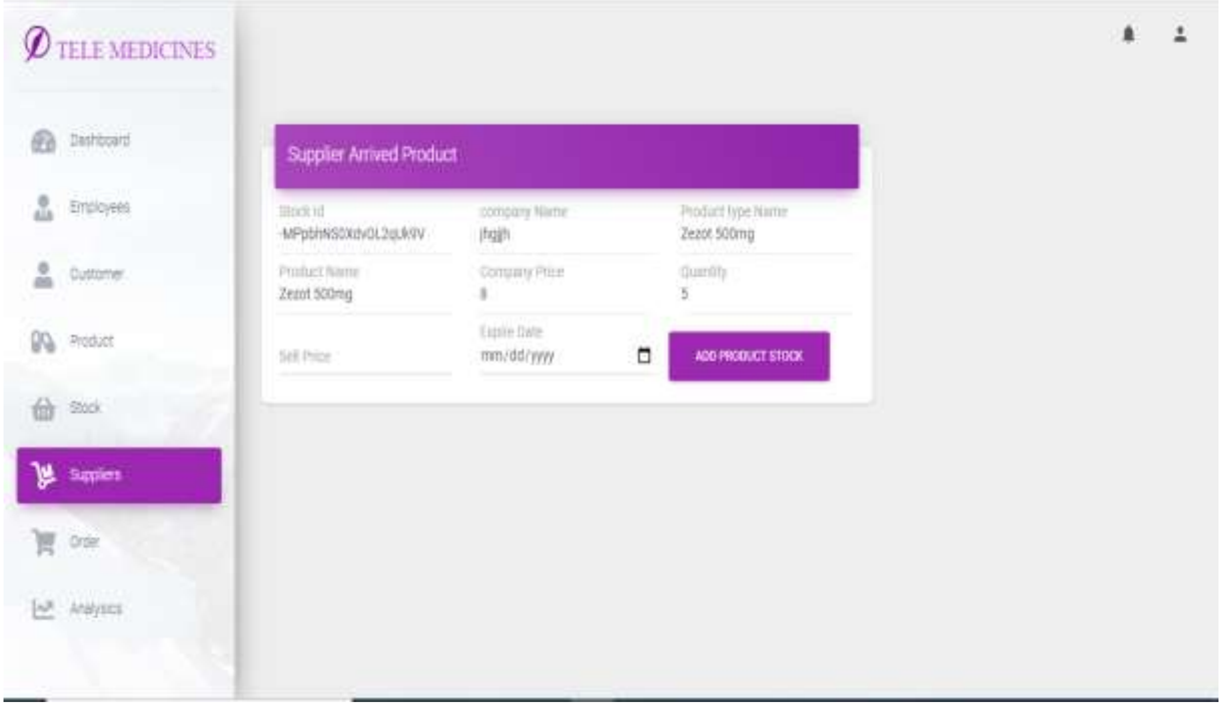
Quantity: 5

Arrived Date: 01/09/2021

Arrived Stock

Product Name	Price	Quantity	Total Price	Arrived date	Action
Zezot 500mg	8	5	40	2021-01-09	Arrived

Screen: Arrived Product



Supplier Arrived Product

Stock id: -MPpbhNS0XdvOL2qUk9V

company Name: jhgjh

Product type Name: Zezot 500mg

Product Name: Zezot 500mg

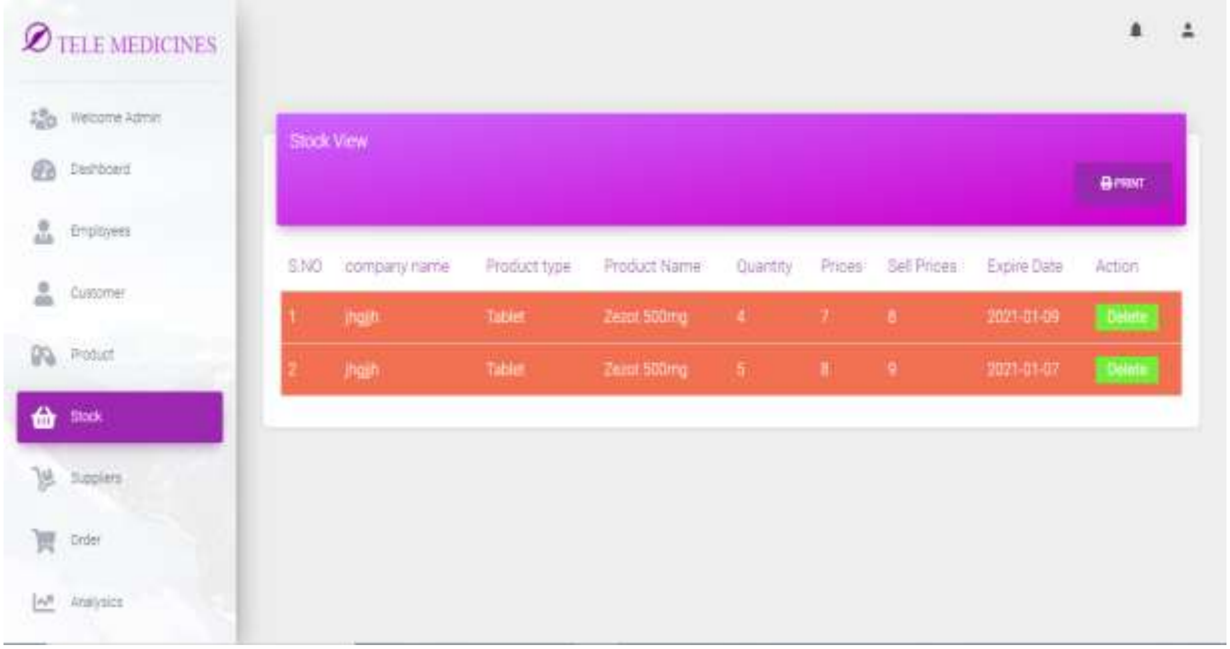
Company Price: 8

Quantity: 5

Sell Price

Expire Date: mm/dd/yyyy

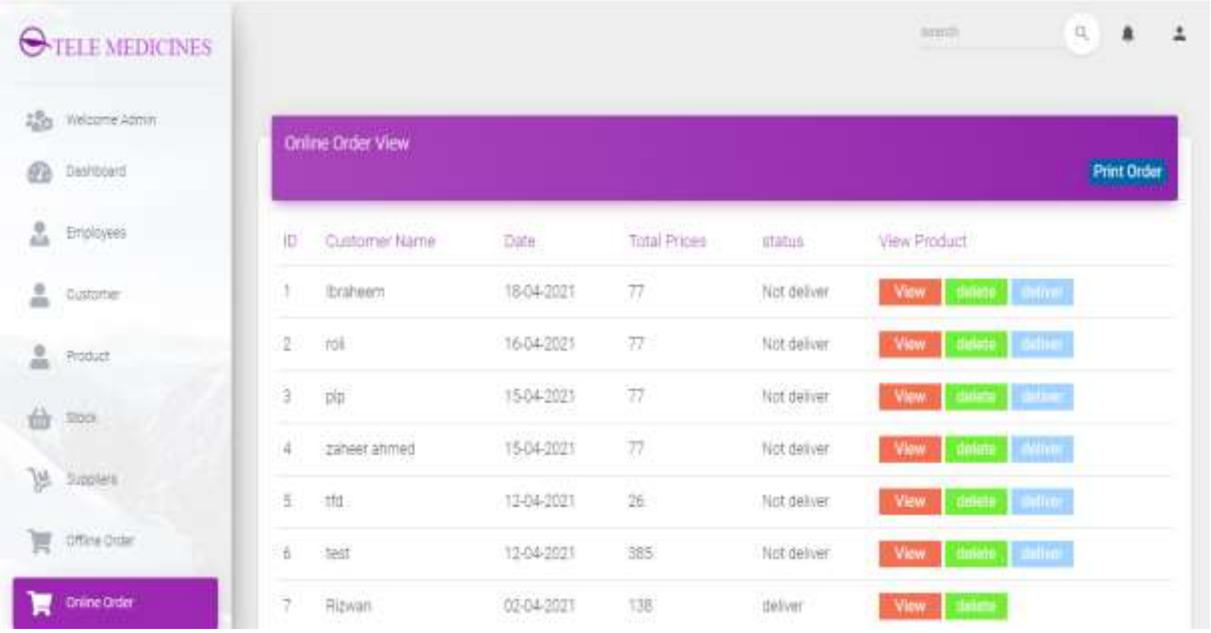
Screen: Stock



Stock View

S.NO	company name	Product type	Product Name	Quantity	Prices	Sell Prices	Expire Date	Action
1	jhgjh	Tablet	Zezol 500mg	4	7	8	2021-01-09	Delete
2	jhgjh	Tablet	Zezol 500mg	5	8	9	2021-01-07	Delete

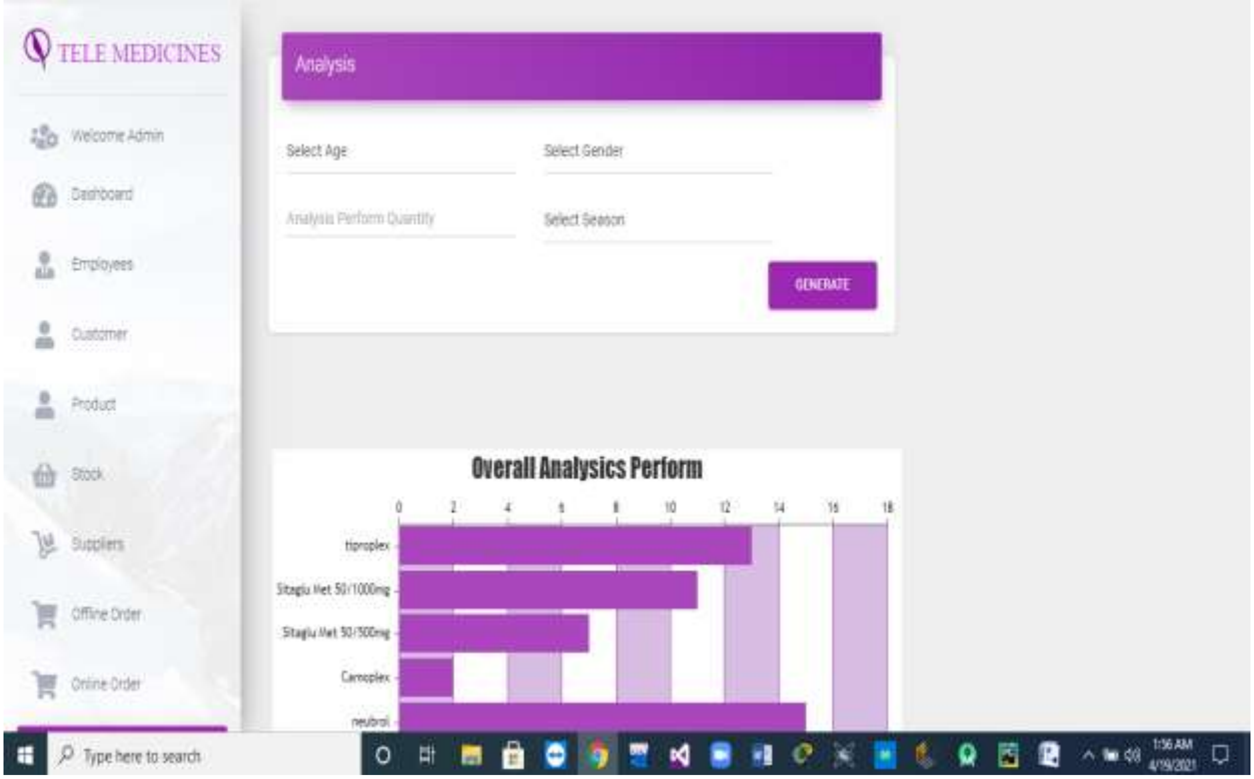
Screen: Online Order



Online Order View

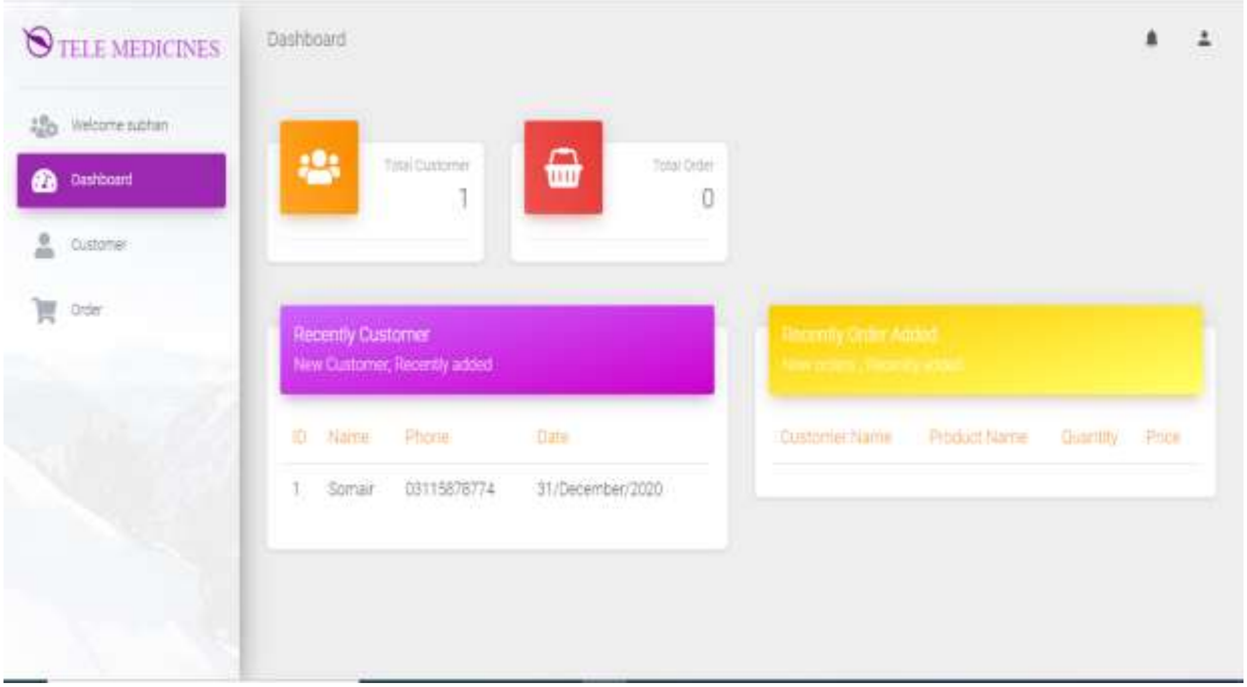
ID	Customer Name	Date	Total Prices	status	View Product
1	Ibraheem	18-04-2021	77	Not deliver	View delete deliver
2	roli	16-04-2021	77	Not deliver	View delete deliver
3	plp	15-04-2021	77	Not deliver	View delete deliver
4	zaheer ahmed	15-04-2021	77	Not deliver	View delete deliver
5	tfd	12-04-2021	26	Not deliver	View delete deliver
6	test	12-04-2021	385	Not deliver	View delete deliver
7	Rizwan	02-04-2021	138	deliver	View delete

Screen: Analytics



3.3.2.2 Employee Module

Screen : Employee Dashboard

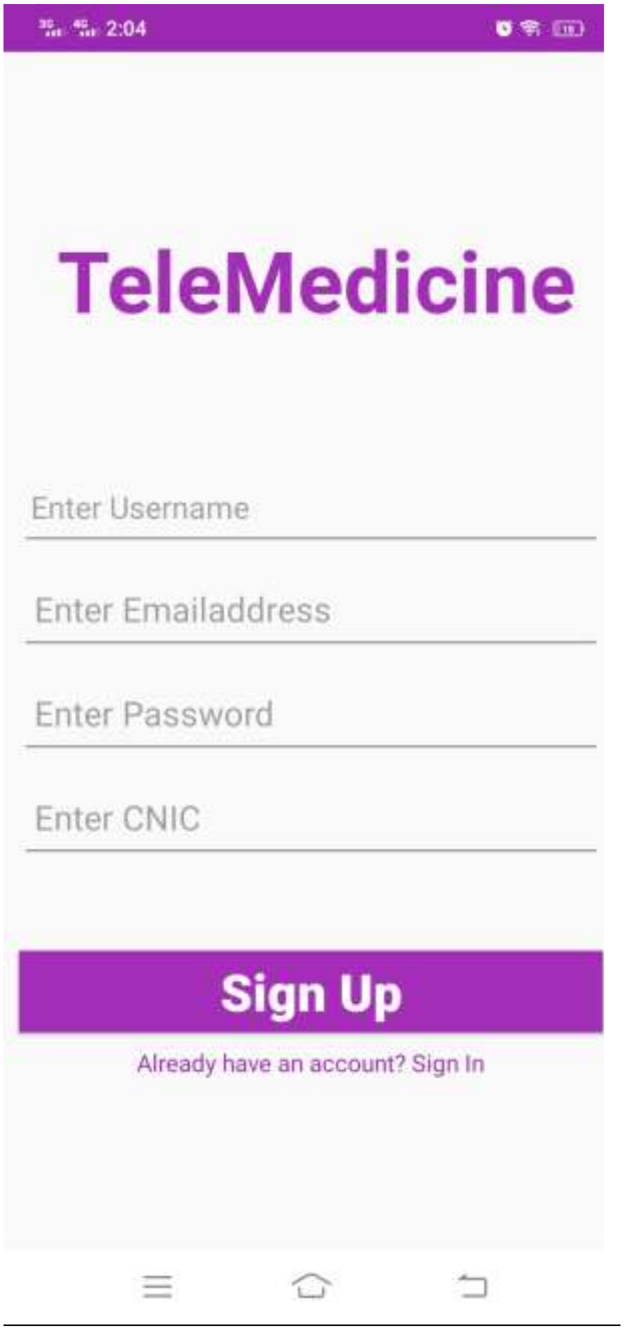


ID	Name	Phone	Date
1	Somair	03115876774	31/December/2020

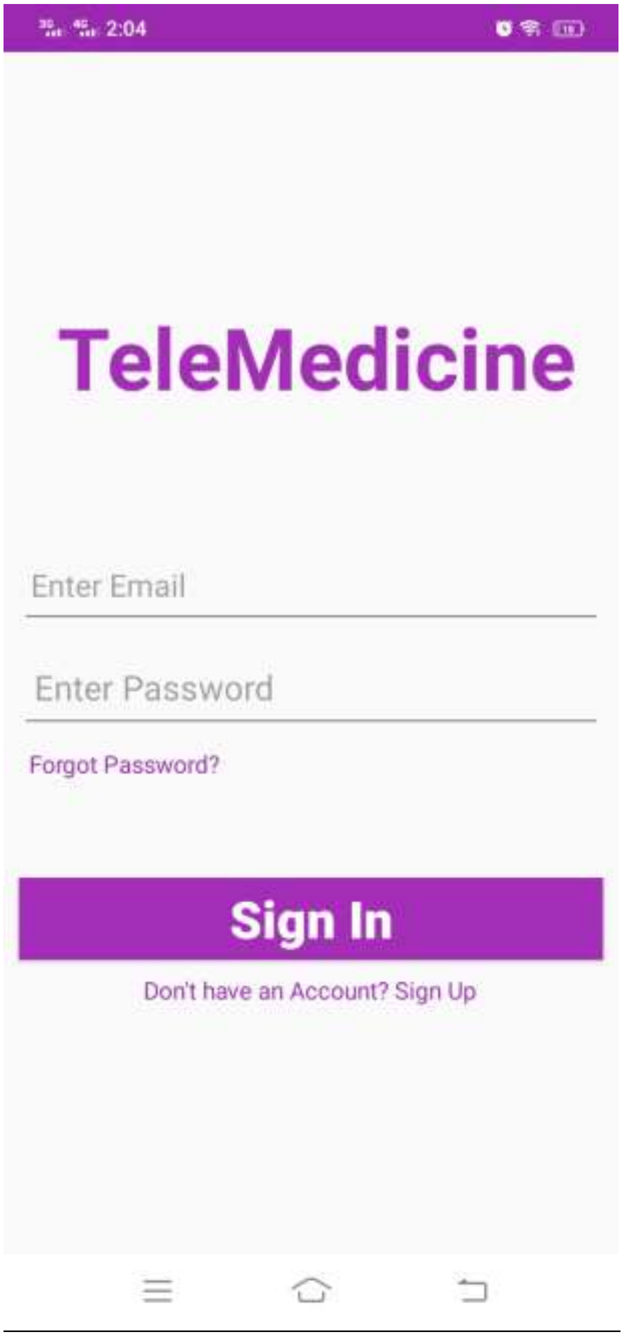
Employee have only customer and order privilege.

3.3.2.3 Android App

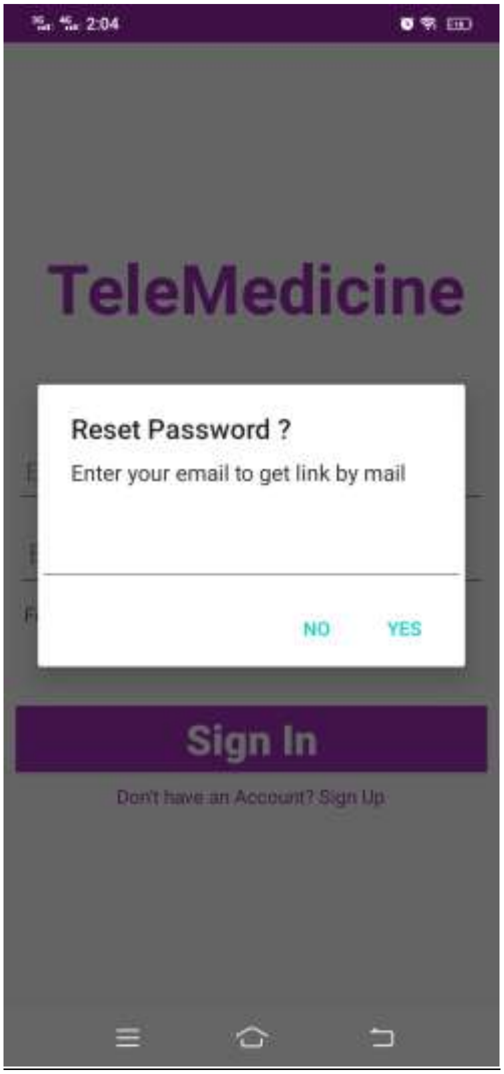
Screen : Sign Up



Screen : Sign In



Screen : Forgot Password



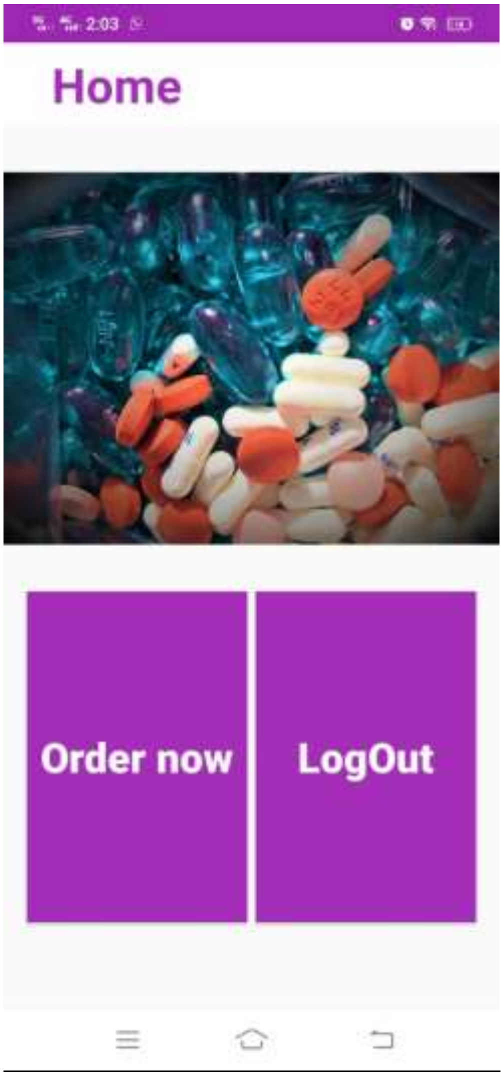
2:04

TeleMedicine

Reset Password ?

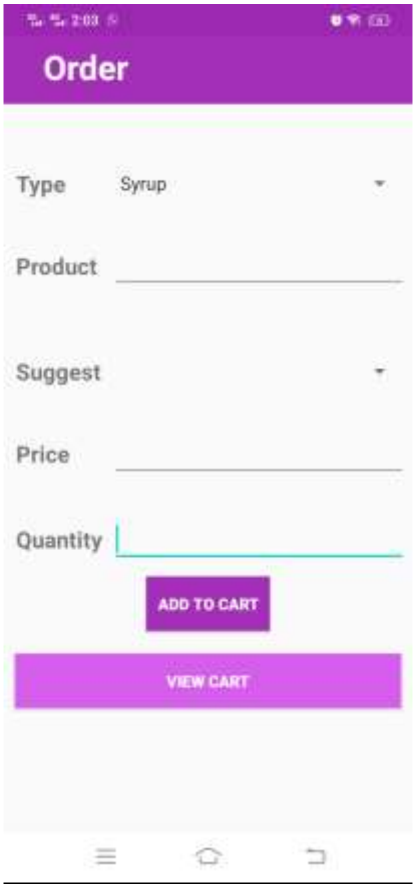
Enter your email to get link by mail

Screen : Home



2:03

Screen : Order



Type Syrup

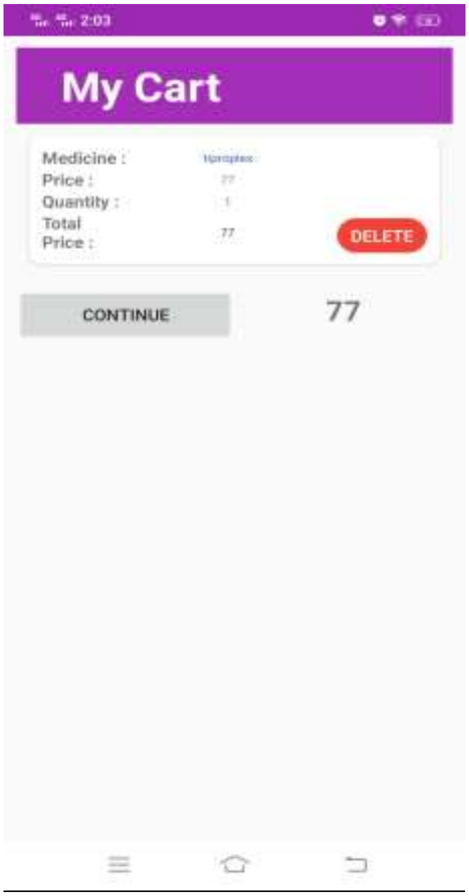
Product

Suggest

Price

Quantity

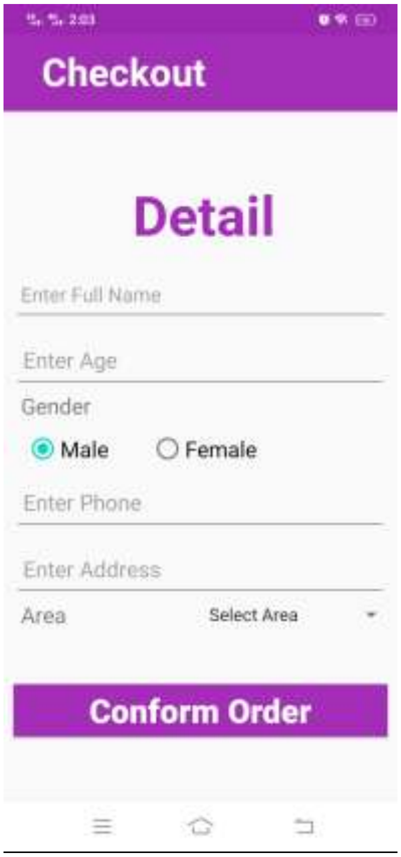
Screen : My Cart



Medicine : [illegible]
Price : 77
Quantity : 1
Total Price : 77

77

Screen : Check Out



Checkout

Detail

Enter Full Name

Enter Age

Gender

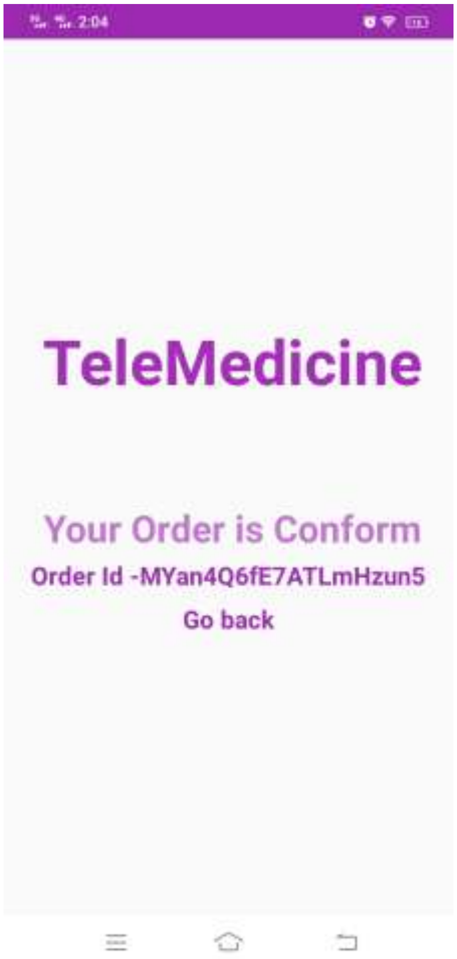
● Male ○ Female

Enter Phone

Enter Address

Area Select Area

Screen : Conformation



TeleMedicine

Your Order is Conform

Order Id -MYan4Q6fE7ATLmHzun5

4 Conclusion

The entire duration of the project has been a considerable learning opportunity with unique challenges that we tried to face head-on and with as much enthusiasm as we can manage. The achievement of our stated goals has proved to be a culmination of an entire year’s worth of learning new things and making mistakes, the end product is something which we are proud to present in front of the committee.

5 References

- https://stackoverflow.com/questions/51484261
- https://www.w3schools.com/python/ref_string_maketrans.asp
- https://github.com/datacamp/courses-introduction-to-python/blob/master/chapter2.md
- Others

6 Appendix A – Development Environment

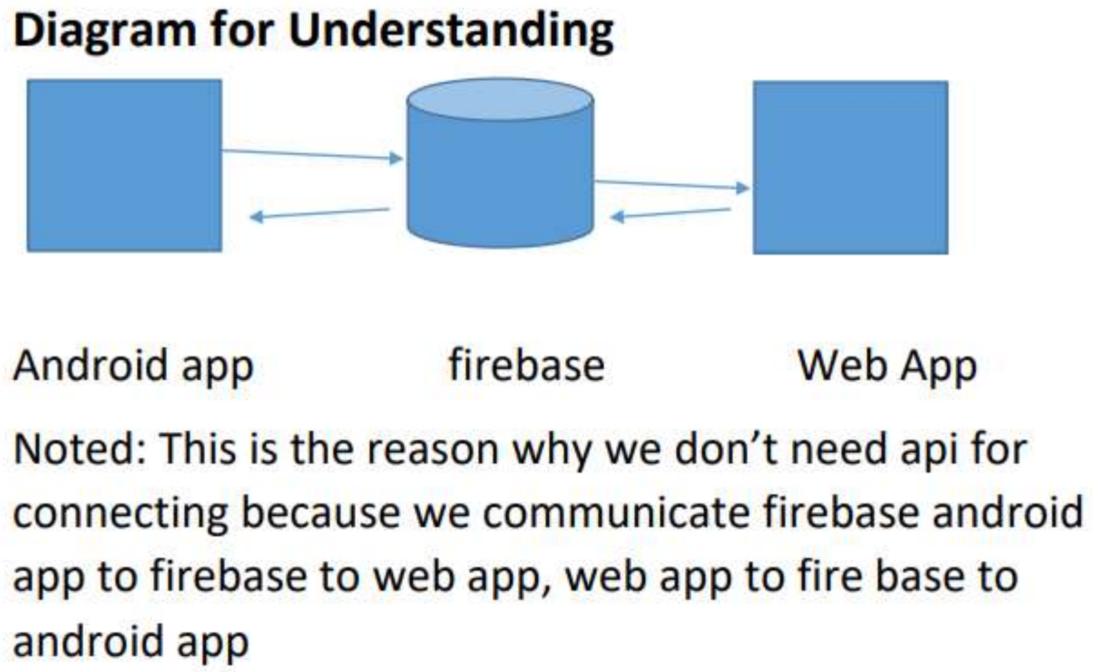


Diagram for Understanding

Noted: This is the reason why we don't need api for connecting because we communicate firebase android app to firebase to web app, web app to fire base to android app

6.1 Operating Systems(s)

6.1.1 Windows

- Microsoft Windows is a progression of graphical interface working frameworks created, advertised, and sold by Microsoft
- Microsoft presented a working climate named Windows on November 20, 1985 as a graphical working framework shell for MS-DOS because of the developing revenue in graphical UIs (GUIs). Microsoft Windows came to overwhelm the world's PC market with more than 90% portion of the overall industry, surpassing Mac OS, which had been presented in 1984.
- As of April 2014, the latest adaptations of Windows for PCs, cell phones, worker PCs and implanted gadgets are separately Windows 8.1,Windows Phone 8.1.
- Developer – Microsoft
- Version – Windows 10
- Environment – 64 Bit
- Website–windows.microsoft.com

6.2 Programming Language(s)

6.2.1 Python

Python is a broadly utilized universally useful, undeniable level programming language. It was made by Guido van Rossum in 1991 and further created by the Python Software Foundation. It was planned with an accentuation on code meaningfulness, and its sentence structure permits software engineers to communicate their ideas in less lines of code. Python is a programming language that allows you to work rapidly and incorporate frameworks all the more productively.

Major Implementations: Flask Framework, Pycharm Python

Language Paradigms:

Multi-paradigm: structured,imperative, object-oriented,event-driven, task-driven,functional, generic,reflective, concurrent

Developer& Designer– Microsoft

Appeared in – 2000

Website - msdn.microsoft.com/en-us/library/aa288436(v=vs.71).aspx

Also we use java script.

Language name which we use:

- Python
- Java script
- Html + Css

6.3 Programming Paradigm

6.3.1 Object-oriented programming

Object-oriented programming (OOP) is a programming worldview that addresses the idea of "objects" that have information fields (ascribes that portray the article) and related strategies known as techniques. Items, which are generally occasions of classes, are utilized to cooperate with each other to plan applications and PC programs.

6.3.2 Object-oriented analysis and design

Object-oriented analysis and design (OOAD) is a mainstream specialized way to deal with investigating, planning an application, framework, or business by applying the item situated worldview and visual demonstrating all through the advancement life cycles to cultivate better partner correspondence and item quality.

6.4 Communication and Leadership

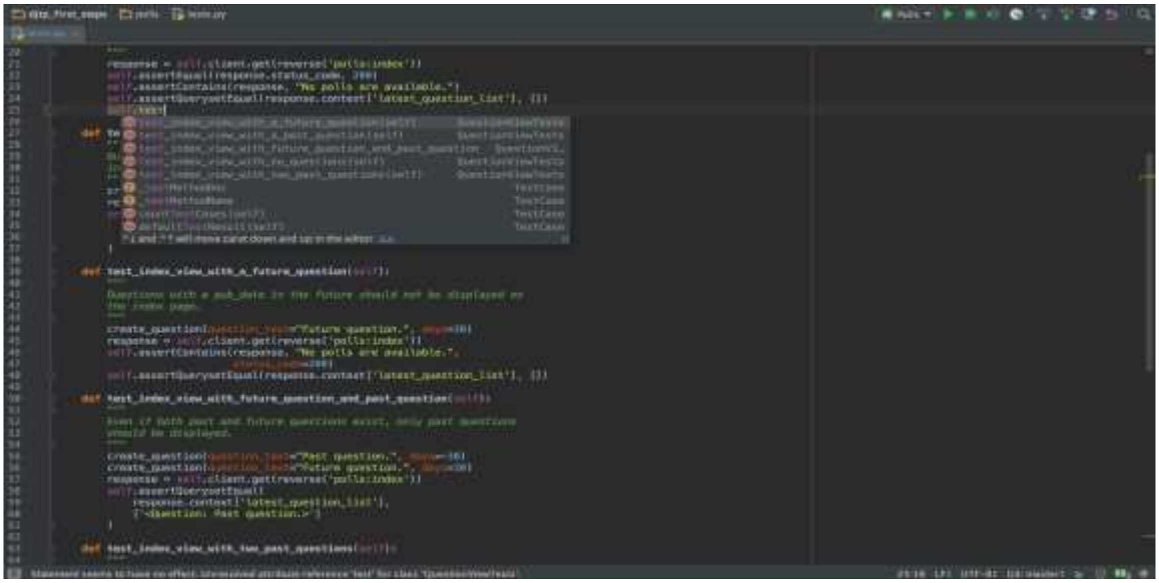
6.4.1 Democratic Decentralized

The crucial delegation of power among group mates remains that power is equally distributed as well as the responsibility. We both had the veto power to decide upon the development processes and methodologies and the functions and modules were distributed while we worked separately on them with the consensus and co-operation of our partner. The power did not rest with any one particular individual and overall it was a constructive and to-and-fro development.

7 Appendix B – Tools and Software

Pycharm 2020.2.1

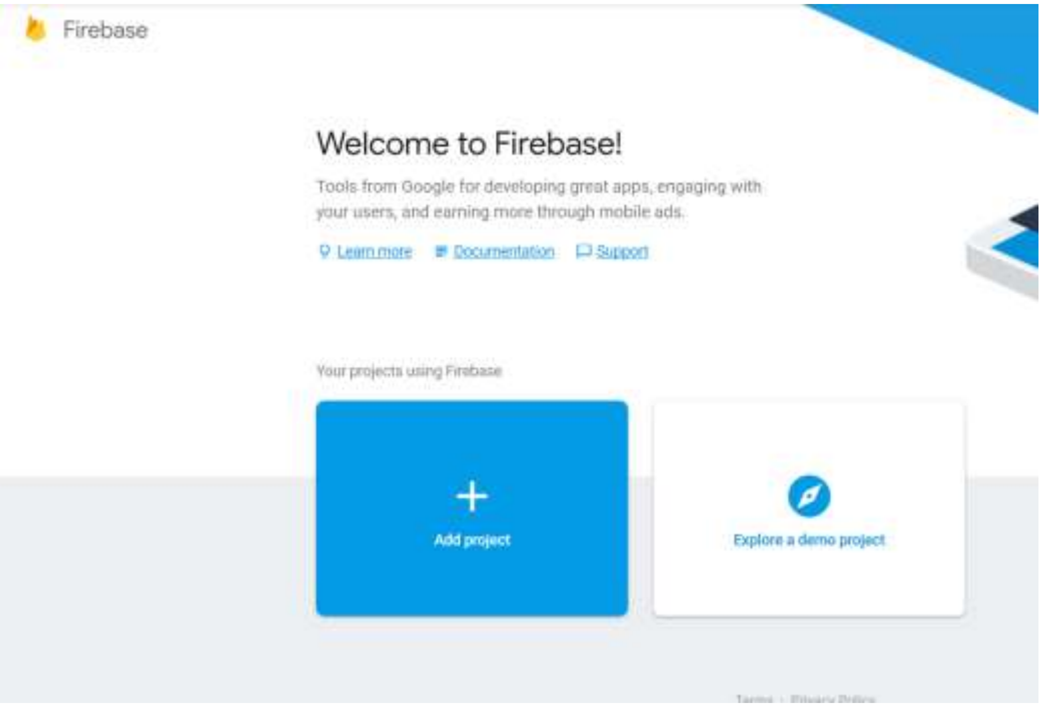
7.1 Pycharm



```
        """
        response = self.client.get(reverse('polls:index'))
        self.assertEqual(response.status_code, 200)
        self.assertContains(response, "No polls are available.")
        self.assertQuerysetEqual(response.context['latest_question_list'], [])
        self.test|
            test_index_view_with_a_future_question(self)          QuestionViewTests
            test_index_view_with_a_past_question(self)            QuestionViewTests
            test_index_view_with_future_question_and_past_question   QuestionVi...
            test_index_view_with_no_questions(self)               QuestionViewTests
            test_index_view_with_two_past_questions(self)         QuestionViewTests
            _testMethodDoc                                               TestCase
            _testMethodName                                              TestCase
            countTestCases(self)                                         TestCase
            defaultTestResult(self)                                      TestCase
        )

    def test_index_view_with_a_future_question(self):
        """
        Questions with a pub_date in the future should not be displayed on
        the index page.
        """
        create_question(question_text="Future question.", days=30)
        response = self.client.get(reverse('polls:index'))
        self.assertContains(response, "No polls are available.",
                            status_code=200)
        self.assertQuerysetEqual(response.context['latest_question_list'], [])

    def test_index_view_with_future_question_and_past_question(self):
        """
        Even if both past and future questions exist, only past questions
        should be displayed.
        """
        create_question(question_text="Past question.", days=-30)
        create_question(question_text="Future question.", days=30)
        response = self.client.get(reverse('polls:index'))
        self.assertQuerysetEqual(
            response.context['latest_question_list'],
            ['<Question: Past question.>']
        )

    def test_index_view_with_two_past_questions(self):
```

PyCharm is a coordinated improvement climate (IDE) utilized in PC programming, explicitly for the Python language. It is created by the Czech organization JetBrains. ... PyCharm is cross-stage, with Windows, macOS and Linux forms.

7.2 Firebase



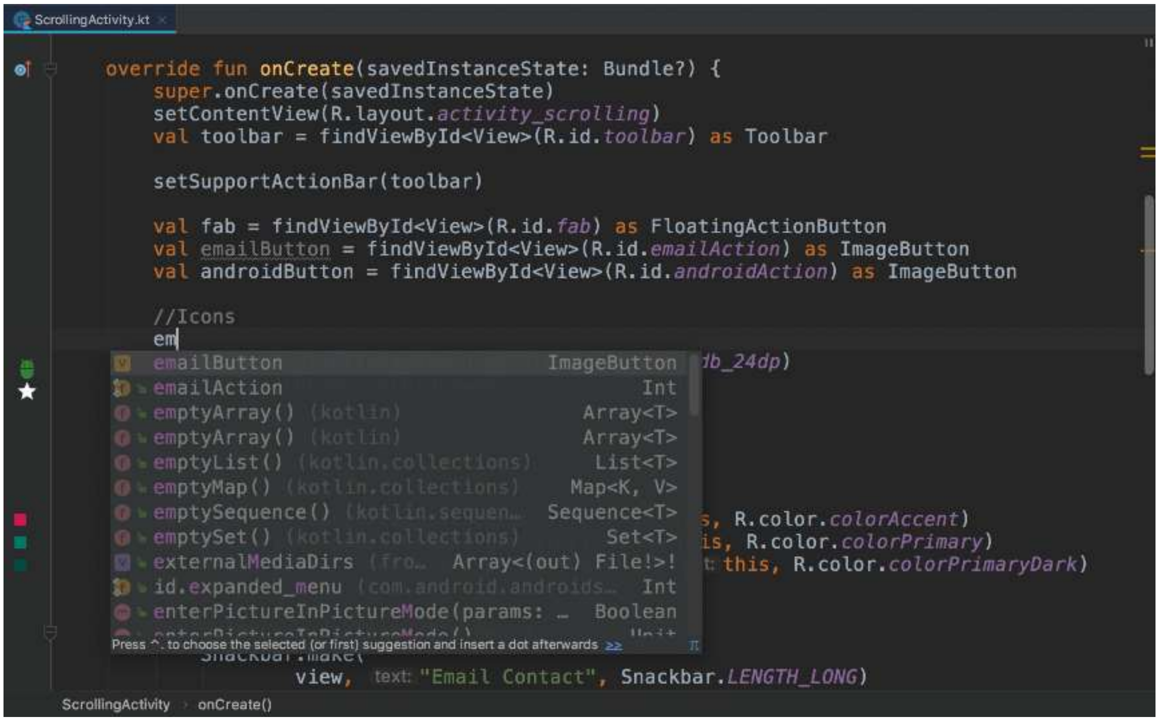
Firebase

Welcome to Firebase!

Tools from Google for developing great apps, engaging with your users, and earning more through mobile ads.

Firebase is Google's versatile application advancement stage that helps you fabricate, improve, and become your application. Here it is again in greater letters, for sway: Firebase is Google's portable application advancement stage that helps you construct, improve, and become your application.

7.3 Android Studio



```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_scrolling)
    val toolbar = findViewById<View>(R.id.toolbar) as Toolbar

    setSupportActionBar(toolbar)

    val fab = findViewById<View>(R.id.fab) as FloatingActionButton
    val emailButton = findViewById<View>(R.id.emailAction) as ImageButton
    val androidButton = findViewById<View>(R.id.androidAction) as ImageButton

    //Icons
    em
                                              db_24dp)

                                              s, R.color.colorAccent)
                                              is, R.color.colorPrimary)
                                              t: this, R.color.colorPrimaryDark)

        Snackbar.make(
                view,  text: "Email Contact", Snackbar.LENGTH_LONG)
```

```
emailButton                                      ImageButton
emailAction                                              Int
emptyArray() (kotlin)                               Array<T>
emptyArray() (kotlin)                               Array<T>
emptyList() (kotlin.collections)                     List<T>
emptyMap() (kotlin.collections)                    Map<K, V>
emptySequence() (kotlin.sequen…                  Sequence<T>
emptySet() (kotlin.collections)                       Set<T>
externalMediaDirs (fro…           Array<(out) File!>!
id.expanded_menu (com.android.androids…                  Int
enterPictureInPictureMode(params: …              Boolean
enterPictureInPictureMode()                             Unit
Press ^. to choose the selected (or first) suggestion and insert a dot afterwards >>
```

Android Studio is the authority coordinated improvement climate for Google's Android working framework, based on JetBrains' IntelliJ IDEA programming and planned explicitly for Android advancement.

8 Appendix C – Code

8.1 Web_Code

Email.py

```
#<----------- Import File -------------->
from Controller import  Configure
import pyrebase
#<------------------------------------->
# <--------------   FireBase Initiailzation ---------->
firebase = pyrebase.initialize_app(Configure.firebaseConfig)
auth = firebase.auth()
db = firebase.database()
storage = firebase.storage()
#<------------------------------------------------->
def a(email1,password):
   admin_session_id = auth.create_user_with_email_and_password(email1, password)
   data={
      "email":str(email1),
      "password":str(password)
   }
db.child('Admin').child(admin_session_id['localId']).push(data)
if __name__ == '__main__':
   a("admin",password="admin")
```

Login_validation.py

```
#<----------- Import File -------------->
from Controller import  Configure
from Controller import  libraries
#<-------------------------------------->
# <--------------   FireBase Initiailzation --------->
firebase = libraries.pyrebase.initialize_app(Configure.firebaseConfig)
auth = firebase.auth()
db = firebase.database()
storage = firebase.storage()
#<-------- Get local id from admin database ------->
def Admin_local_id(local_id):
   result=db.child("Admin").child(local_id).get().val()
   if result!=None:
      for d3,(key,value) in enumerate(result.items()):
         admin_id=key
admin_email=db.child("Admin").child(local_id).child(admin_id).get().val()
         return admin_id,admin_email["email"]
```

```
    else:
        return None
```

auth.py

```
from flask import render_template,request,session,g,flash,Flask,redirect,url_for
from Controller import Configure
import pyrebase
firebaseConfig = {
    "apiKey": "AIzaSyAAZSDa4A3ytc-QbeIYCV86QPzlexS2HOg",
    "authDomain": "telemedicines-123c0.firebaseapp.com",
    "databaseURL": "https://telemedicines-123c0.firebaseio.com",
    "projectId": "telemedicines-123c0",
    "storageBucket": "telemedicines-123c0.appspot.com",
    "messagingSenderId": "490806536805",
    "appId": "1:490806536805:web:73a3b462b3e52df8c5748c",
    "measurementId": "G-QPEMY0KWSQ"
  }
def connect_firebase():
# add a way to encrypt those, I'm a starter myself and don't know how
    username: "telemedicinefyp@gmail.com"
    password: "telemedicinefyp123"
    firebase = pyrebase.initialize_app(firebaseConfig)
    auth = firebase.auth()
    #authenticate a user > descobrir como não deixar hardcoded
    user = auth.sign_in_with_email_and_password(username, password)
    #user['idToken']
    # At pyrebase's git the author said the token expires every 1 hour, so it's needed to refresh
it
    user = auth.refresh(user['refreshToken'])
    #set database
    db = firebase.database()
    return db
```

Customer.py

```
from Controller import  libraries
from Controller import  Configure
firebase = libraries.pyrebase.initialize_app(Configure.firebaseConfig)
auth = firebase.auth()
db = firebase.database()
storage = firebase.storage()
def customer_info(customer_name,nic,phone_number,age,gender,address,emp_id):
    try:
        date=libraries.datetime.datetime.now()
        date=date.strftime("%d/%B/%Y")
        data={
            'customer_name':str(customer_name),
            'NIC':str(nic),
            'employee_id':None,
            'phone_number':str(phone_number),
            'age':str(age),
            'gender':str(gender),
            'address':str(address),
            'date':str(date),
            'employees_id': str(emp_id)
        }
        db.child('Customer').push(data)
        return True
    except:
        return False
def customer_id(customer_id):
    try:
        customer_id1=""
        for k, (key, value) in enumerate(customer_id.items()):
                customer_id1 = key
return  customer_id1
    except:
        return 1
def get_customer_information(customer_result,nic):
    try:
        customer_name=""
        customer_nic=""
        phone=""
        address=""
        gender=""
        ages=""
        id=""
        for k, (key, value) in enumerate(customer_result.items()):
            if value["NIC"] == nic:
                customer_name = value["customer_name"]
                customer_nic = nic
                phone=value["phone_number"]
                address=value["address"]
                gender=value["gender"]
                ages=value["age"]
                id=key
        return customer_name,customer_nic,phone,address,gender,ages
    except:
        return False
def update_customer_data(user_data,nic,name,age,phone,address):
    try:
        for d2, (key, value) in enumerate(user_data.items()):
            if nic == value["NIC"]:
                data = {
                    'customer_name': name,
                    'age': age,
                    'NIC': nic,
                    'phone_number': phone,
                    'address': address
                }
                print("key ", key)
                db.child("Customer").child(key).update(data)
                return True
    except:
        return False
def get_customerid(customer_result,nic):
    try:
        cust_id=""
        for k, (key, value) in enumerate(customer_result.items()):
            if value["NIC"] == nic:
                cust_id=key
        return cust_id
    except:
        return False
def get_customer_order_information(customer_result,cust_id):
    try:
        order_id=""
        product_name=""
        price=""
        quantity=""
        totalbill=""
        record_list=[]
        for k, (key, value) in enumerate(customer_result.items()):
            list=[]
            if value["Customer id"] == cust_id:
                product_name = value["Product Name"]
                quantity=value["Quantity"]
                price=value["Price"]
                totalbill=value["Total bill"]
                list.append(product_name)
                list.append(quantity)
                list.append(price)
                list.append(totalbill)
                record_list.append(list)
        print(record_list)
        return record_list
    except:
        return False
```

Order.py

```
from flask import render_template,request,session,g,flash,Flask,redirect,url_for
from Controller import Configure
```

```
import pyrebase
from datetime import datetime
firebase = pyrebase.initialize_app(Configure.firebaseConfig)
auth = firebase.auth()
db = firebase.database()
storage = firebase.storage()
def delete_temp_order():
    try:
        db.child("Temp_Order").remove()
        return True
    except:
        return False
#--------------------Sorted date ho ri hey
def sorted_Expire_date(array):
    array.sort(key=lambda date: datetime.strptime(date, "%Y-%m-%d"))
    return array
def Order_list(Order_list):
    Product_list = []
    Order_list = Order_list.split(',')
    count = 0
    length = int(len(Order_list) / 3)
    for i in range(0, length):
        product = []
        for j in range(0, 3):
            product.append(Order_list[count])
            count = count + 1
        Product_list.append(product)
    return Product_list,length
def order(product_id,quantity,prices):
    try:
        data={
            "product id":str(product_id),
            "quantity":str(quantity) ,
            "price":str(int(prices)*int(quantity))
        }
        db.child('Temp_Order').push(data)
        return True
    except:
        return False
def Order_info(customer_id,employee_id,total_price,time,date,status):
    try:
        data={
            'Customer_id':str(customer_id),
            'Employee_id':str(employee_id),
            'Total_price':str(total_price),
            'Time':str(time),
            'Date':str(date),
            'Status':str(status)
        }
        print(data)
        db.child('Order').push(data)
        return True
    except:
        return False
```

8.2 App_Code

SignUp.java

```
package com.example.telemedicine;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.basgeekball.awesomevalidation.AwesomeValidation;
import com.basgeekball.awesomevalidation.ValidationStyle;
import com.basgeekball.awesomevalidation.utility.RegexTemplate;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.kaopiz.kprogresshud.KProgressHUD;
import java.util.regex.Pattern;
public class SignUp extends AppCompatActivity {
    EditText txtusername, txtemail, txtpassword, txtcnic;
    Button btnsignup;
    TextView txtalredyhaveaccount;
    KProgressHUD hud;
    DatabaseReference databaseReference;
    FirebaseDatabase firebaseDatabase;
    private FirebaseAuth firebaseAuth;
    AwesomeValidation awesomeValidation;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sign_up);
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        txtusername = findViewById(R.id.txtUsername);
        txtemail = findViewById(R.id.txtEmail);
        txtpassword = findViewById(R.id.txtPassword);
        txtcnic = findViewById(R.id.txtCnic);
        btnsignup = findViewById(R.id.btnSignup);
        txtalredyhaveaccount = findViewById(R.id.txtAlredyhaveaccount);
        databaseReference =
FirebaseDatabase.getInstance().getReference("OnlineCustomer");
        firebaseAuth = FirebaseAuth.getInstance();
        awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
        //Validation Start
        awesomeValidation.addValidation(this,R.id.txtUsername,
                RegexTemplate.NOT_EMPTY,R.string.invalid_name);
        awesomeValidation.addValidation(this,R.id.txtEmail,
                Patterns.EMAIL_ADDRESS,R.string.invalid_email);
        awesomeValidation.addValidation(this,R.id.txtPassword,
                ".{6,}",R.string.invalid_password);
        awesomeValidation.addValidation(this,R.id.txtCnic,
                "\\d{5}[-|]\\d{7}[-|]\\d", R.string.invalid_cnic);
        txtalredyhaveaccount.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(),SignIn.class));
            }
        });
        btnsignup.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final String username = txtusername.getText().toString().trim();
                final String email = txtemail.getText().toString().trim();
                String password = txtpassword.getText().toString().trim();
                final String cnic = txtcnic.getText().toString().trim();
                if(awesomeValidation.validate()){
```

```
                firebaseAuth.createUserWithEmailAndPassword(email, password)
                        .addOnCompleteListener(SignUp.this, new
OnCompleteListener<AuthResult>() {
                            @Override
                            public void onComplete(@NonNull Task<AuthResult> task) {if
(task.isSuccessful()) {
customer information = new customer(username,email,cnic);
FirebaseDatabase.getInstance().getReference("OnlineCustomer")
.child(FirebaseAuth.getInstance().getCurrentUser().getUid())
.setValue(information).addOnCompleteListener(new OnCompleteListener<Void>() {
@Override
public void onComplete(@NonNull Task<Void> task) {
hud = KProgressHUD.create(SignUp.this)
.setStyle(KProgressHUD.Style.SPIN_INDETERMINATE)
.setLabel("Please wait")
.setDetailsLabel("Account Created")
.setCancellable(true)
.setAnimationSpeed(2)
.setDimAmount(0.5f)
.show();
startActivity(new Intent(getApplicationContext(), SignIn.class));
                                        }
                                    });
                                } else {
                                    Toast.makeText(SignUp.this, "Authentication Failed",
Toast.LENGTH_SHORT).show();
                                }
                            }
                        });
            }else{
             }
        }
    });
}
@Override
public void onBackPressed() {
    super.onBackPressed();
    Intent i=new Intent(SignUp.this,SignIn.class);
    startActivity(i);
    }
}
```

9 Appendix D – Team Members

Subhan Hussain (September 9, 1999) I am a student of Bachelors of Computer Science and I am currently in the final semester at PAF-KIET. Have furnished my skills at the .NET framework using C# and also Python. Email: a03112343992@gmail.com

Sahar Saleem (November 14, 1998) I am a student of Bachelors of Computer Science and I am currently in the final semester at PAF-KIET. Have furnished my skills at the .NET framework using C# and also Python. Email: saharsaleem2017@gmail.com

10 Annexure I – Copy of Approved FYP Proposal

SEMESTER 8TH	**FALL 2020**	**YEAR 2020**	**2020-2021**
TITLE OF PROPOSED PROJECT			
TeleMedicine			
Project Category (choose one)	**□ Product based**	**□ Research-based**	
SUPERVISOR INFORMATION			
Supervisor Name:	**(leave blank)**	**Organization/ Designation**	**(leave blank)**
Contact No:	**(leave blank)**	**email:**	**(leave blank)**
STUDENT(S) INFORMATION			
S#	**Student ID**	**Name**	
1	**8684**	**SAHAR SALEEM**	
Contact No:	**+923410339879**	**email:**	**SAHARSALEEM2017@GMAIL.COM**
2	**9315**	**SUBHAN HUSSAIN**	
Contact No:	**+923498963727**	**email:**	**a03112343992@gmail.com**
PROJECT AREA/TOOLS			
Tools Required:	ANDROID STUDIO, PYCHAM		
Area/Specialization:	AI		
SUMMARY OF PROPOSED PROJECT (MAXIMUM 300 WORDS)			
Our project TeleMedicine design for medical stores. It is very fast , reliable and easy to use. It provides all the information to the admin about the inventory. It will mange and control all the stock. our project convert the manual chemist inventory in to automated inventory system. It manage all the records in a very organized form Telemedicine is a web application to maintain all type of data and our android application will capable to sell medicine online. A user can order online . This idea will provide medicine in lockdown situation because it's a home-base delivery application.			

PROJECT OBJECTIVE(S)/OUTCOMES

.

The principle objective of the Medical Shop Management System is to deal with the subtleties of Medical Shop. Our main objective is to manage the data of medical shop,stock and sells . Our project is made to reduce the workload manually and use digital or online technology to manage the medical store It is very use friendly , fast and reliable

FUNCTIONAL FEATURES

For FYP 1:

1. Web application
2. Medical store complete inventory maintains
3. Invoice
4. Complete detail of all medicine will display
5. It is also worth mentioning that this medicine is for a disease

For FYP 2:

6. If a Medicine is not available, then suggest another Medicine with same formula
7. Analytics perform on the bases of season, age, gender, area, etc, e.g. kis season mein kon c med zada sale ho rahi hay to is say pata chalay ga k kis season mein kon c bimari zada hoti hay
8. Online ordering through mobile app (only customer order)

For Proposal Defense Purpose

PROPOSED ADVISORY COMMITTEE		
S#	**Faculty Name**	**Signature**
1	Nadeem qamar	
2	Abdul wahab	
3	Marks	6/10
4		

FYP Committee

FYP COMMITTEE			
S#	**Member(s) Name**	**Designation**	**Signature**
1	**Syed Nabeel Ali (FYP Coordinator)**	Lecturer	
2	**Mr. Kashif Bashir (Head of the Department)**	Asst. Professor	
3	**Dr. Muhammad Khalid Khan (Director CoCIS)**	Professor	
	Date	00/00/0000	

Jury remarks:

- Medical store ki poori inventory maintain ho ge
- Invoice banay ge
- Agar aik medicine available nahein hay , to us he formulae ki dusri medicine suggest ker day dusri company ki
- Analytics perform kernay hein

  - With respect to season, age , gender, area, etc, e.g. kis season mein kon c med zada sale ho rahi hay to is say pata chalay ga k kis season mein kon c bimari zada hoti hay
- Purchaser jo medicine purchase ker raha hay wo medicine kin kin merz mein use ho sakti hay, ye mention ho
- Her medicine ki complete info bhi display ho
- Web + mob app
- Ye idea stds ka nahein hay, faculty ka idea hay laikin stds apni merzi say is idea per agree hein kaam kernay k liye
- ~~Ye project live chalana hay warna saari requiments poooti honay k bawajood fail~~

~~Kuch requirements medical store walay ki khud ki hon ge , wo bhi impelement kerni hein~~

11 Annexure II – Software Planning & Costing Planning

a. Milestones for Semester One

Week	Student Task
Meeting 1	Requirement gather
Meeting 2	Diagrams and Prototype on paper
Meeting 3	View of web
Meeting 4	Perform Order and create entire database
Meeting 5	Perform Supplier work, Stock
Meeting 6	Perform all operation

b. Milestones for Semester Two

Week	Student Task
Meeting 1	Gathering Live DataSet
Meeting 2	Perform Analytics
Meeting 3	View of Android App
Meeting 4	Perform Order and Insert data in database
Meeting 5	Testing
Meeting 6	Bug Removing and Testing

11.1 Hardware &Software Requirements

11.1.1 Hardware

- Android
- Tablet
- General PC

11.1.2 Software

- chrome
- Firefox

11.2 Costing

The project is expected to cost at least PKR 90,000

12 Annexure IV – Test Case Specifications

Tests Designed By: Subhan Hussain & Sahar Saleem

Tests Executed By: Subhan Hussain & Sahar Saleem

Test Case ID: 001

Test Priority: High

Test Title: Regression Testing

Pre-conditions: The software has been fully developed after new enhancements and updates

	Test Steps	Test data	Expected Result	Actual Result	Staus (Pass/Fail)
1	Open Final Compilation	Project	Compile Successfully	Compiled Successfully	Pass
2	Compile				
3	Debug				

Post-conditions: The project is functioning adequately

Test Case ID: 002

Test Priority: High

Test Title: Scalability Testing

Pre-conditions: The GUI design for html5 and Smart Phone is complete

	Test Steps	Test data	Expected Result	Actual Result	Staus (Pass/Fail)
1	Change Viewing Platform	Project	GUI remains intact	GUI remained intact	Pass
2	Load the project				
3	Run the project				

Post-conditions: The screen adjusted itself rather than distorting any images and View