PYTHON PROJECT-CA3

Pre-submission Report

DESIGN A ATM MANAGEMENT SYSTEM USING PYTHON.

Project Members:

NAME	Roll no	Reg no
Preeti kumari	RK21PBA05	12110546
Suram Praveen	RK21PBB48	12112042
Deepak Sharma	RK21PBB49	12111779

1.Introduction:

ATM Simulator project is written in Python. The project file contains a python script. This is a simple console based system which is very easy to use. Talking about the system, it contains various functions which include Account Statement, Withdrawing, Depositing amount and changing the pin. Here, at first the user has to enter an existing username, when the username matches the system proceed toward the next procedure i.e asking pin number. When a user passes all these sign-in procedures, he/she can use all those features. It is too easy to use, he/she can check their respective account statements.

While depositing or withdrawing amount, he/she just has to enter the amount then the system calculates the total remaining balance of the respective account and displays to the user. And the user can view all these transactions from the account statement. In this ATM Simulator, the user can also change the pin number. For this, the user has to enter the New pin code and then confirm it in order to change the pin code. This simple console based ATM simulator provides the simple account balance management of a respective account. It contains all the essential features. There is no database connection or neither any external text or other files used in this mini project to save user's data. Everything is set inside the source code whether its pin code or the amount.

2.Functions In ATM:

ATMs are Automated Teller Machines that are used to carry day-to-day financial transactions. ATMs can be used to withdraw money or to deposit money or even to know the information of an account like the balance amount, etc. They are convenient and easy to use,and it allows consumers to perform quick self-service transactions.

In this article, we will discuss the ATM Management System which is an application that provides users with every aspect that an actual Automated Teller Machine i.e., ATM should have. It is a menu-driven program having ATM functions which include:

1. Enter Name, Account number, Account type to be shown during transactions.

2.Shows the information about the person who is doing the transaction.

3.Enter amount to deposited in the account.

4.Shows the Balance in the account.

5.Enter amount to be withdrawn from the account, and then it shows available balance.

6.Cancel the transaction.

setvalue(): This function is used here to set the data using basic input and output method.

showvalue(): This function is used to print the data.

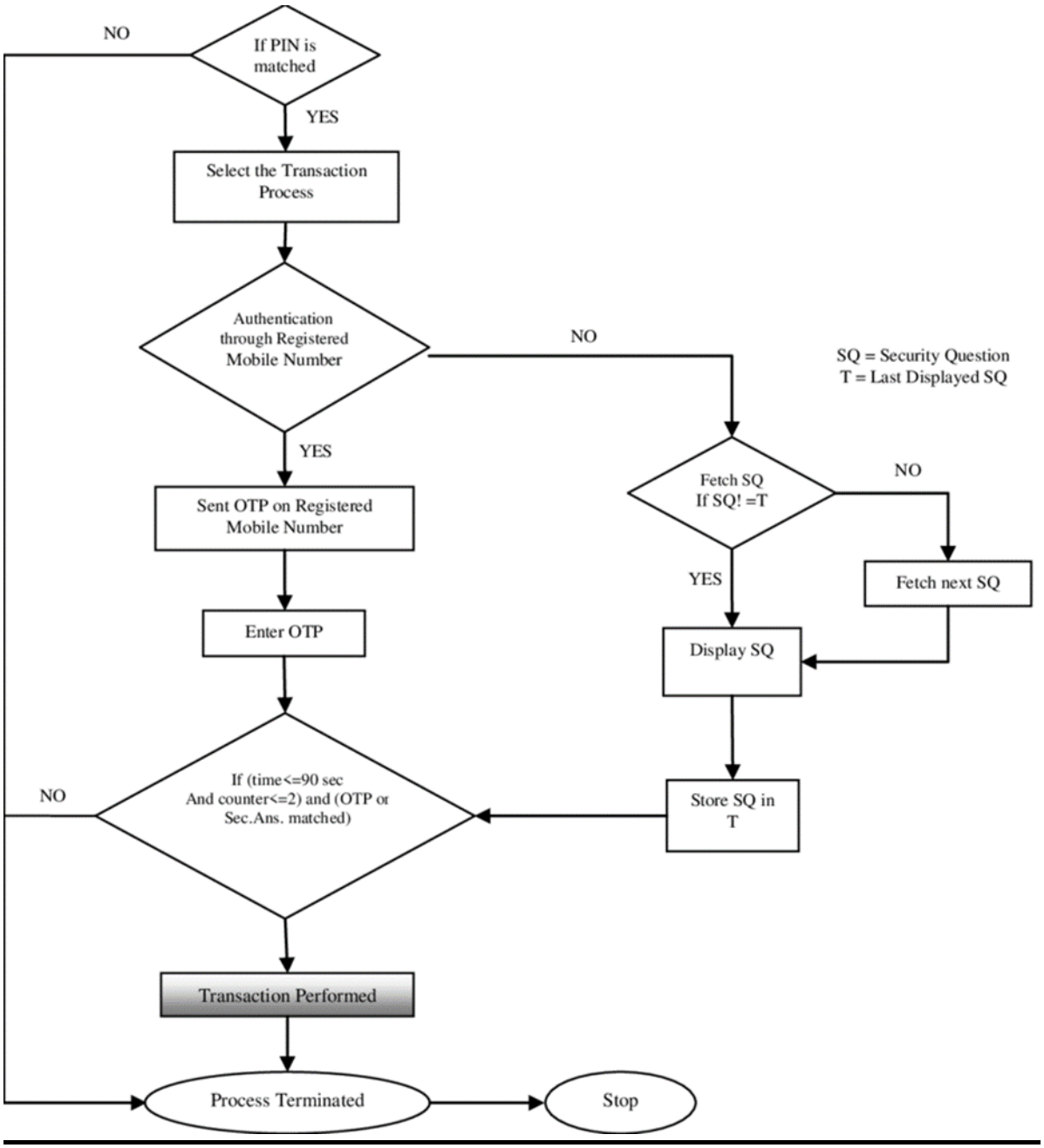
deposit(): This function helps to deposit money in a particular account.

showbal(): This function shows the total balance available after deposition.

withdrawl(): This function helps to withdraw money from the account.

main(): In this function, there is a simple switch case (to make choices) inside an infinite while loop so that every time user gets to select choices.

3.Flowchart:


NO
If PIN is matched
YES
Select the Transaction Process
Authentication through Registered Mobile Number
NO
SQ = Security Question
T = Last Displayed SQ
YES
Sent OTP on Registered Mobile Number
Fetch SQ If SQ! =T
NO
YES
Fetch next SQ
Enter OTP
Display SQ
If (time<=90 sec And counter<=2) and (OTP or Sec.Ans. matched)
NO
Store SQ in T
Transaction Performed
Process Terminated
Stop

4.Responsibilities for each student:

RK21PBA05-Preeti Kumari
>> will work on the Designing the Gui and requiremets of the project.

RK21PBB48-Suram Praveen

>>will work on the main program and editing the final report.

RK21PBB49-Deepak Sharma

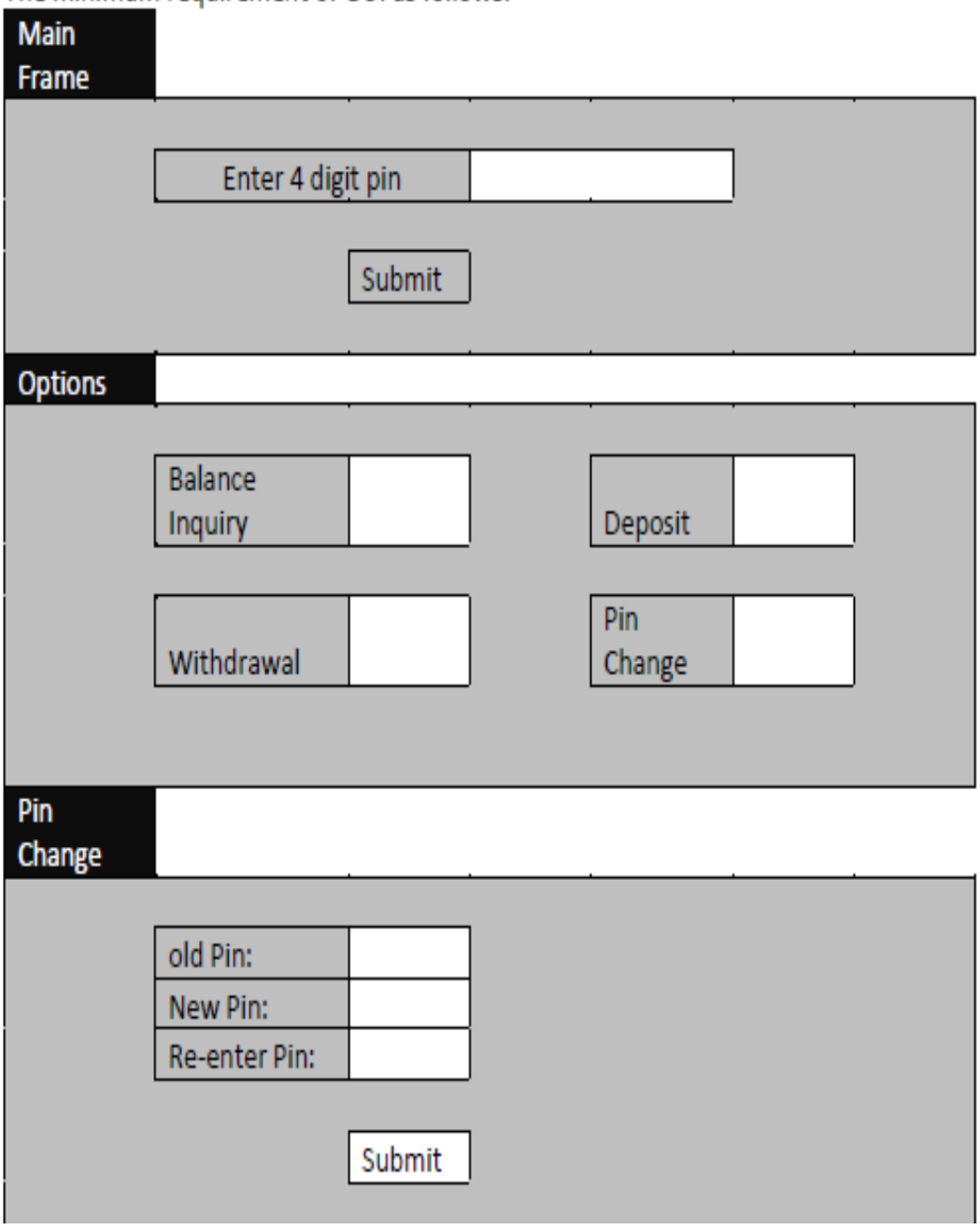
>>will work on the structure of the project and detecting the errorsand coding.

5.Framework:

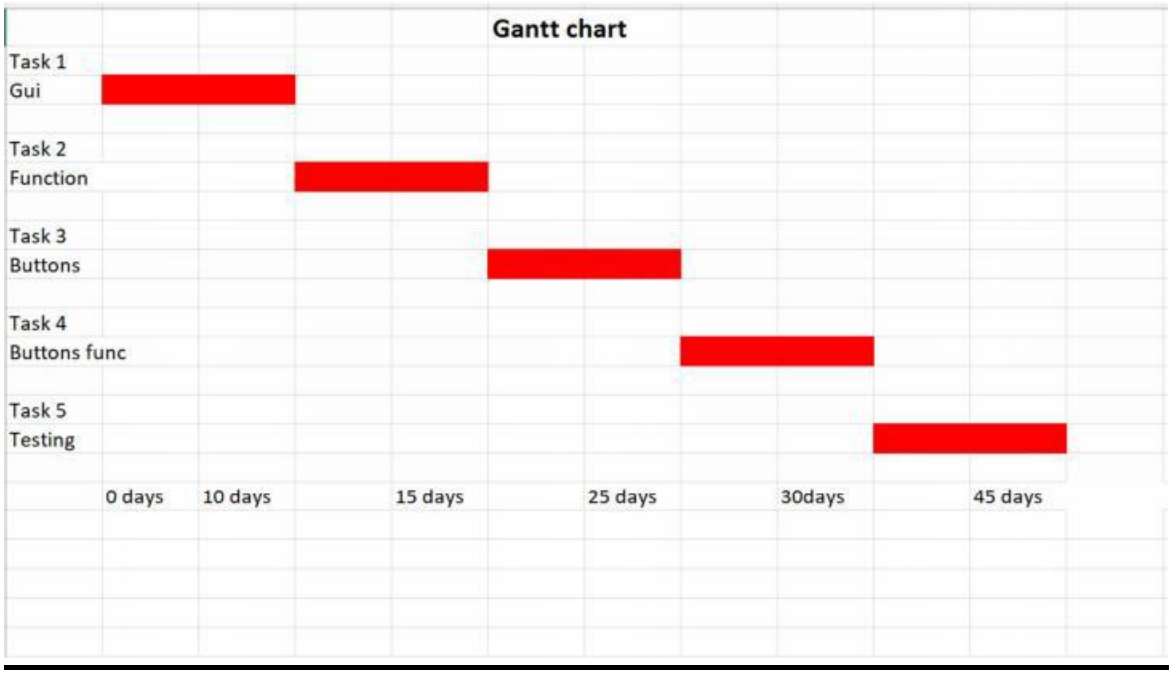
Project Title: Design a ATM Management System using python.

Project Description:

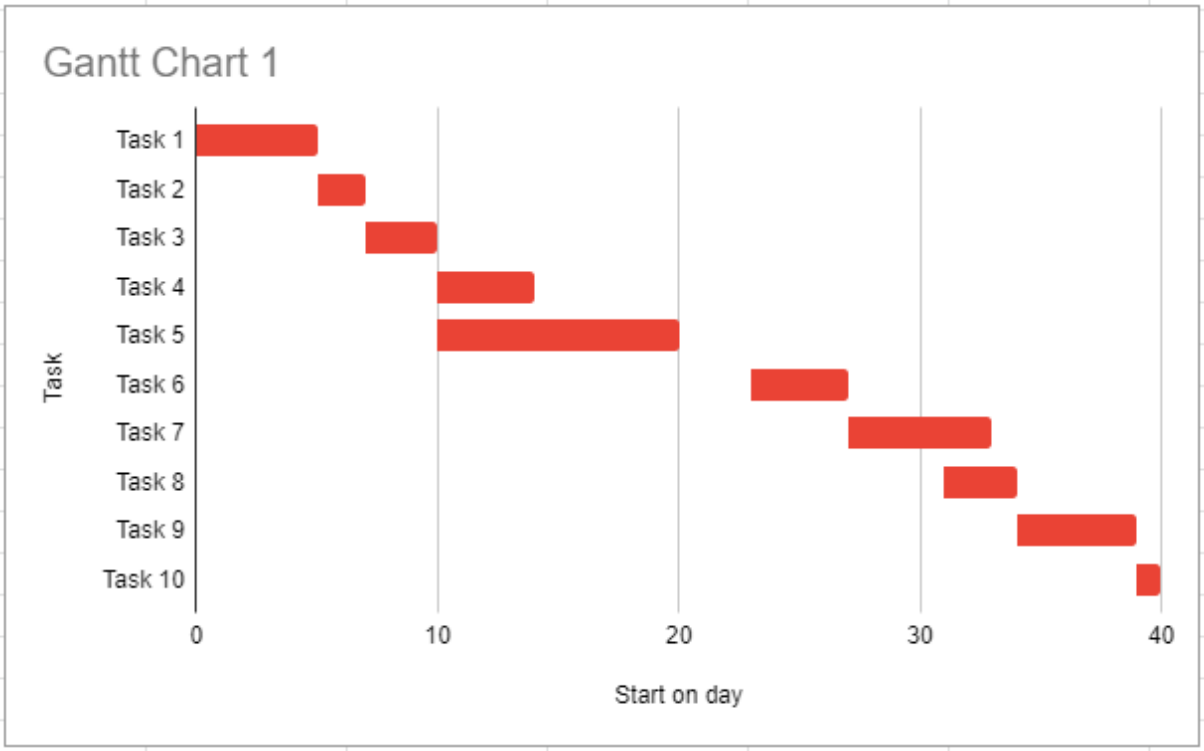
The minimum requirement of GUI as follows:



Gantt Chart:


Gantt chart
Task 1
Gui
Task 2
Function
Task 3
Buttons
Task 4
Buttons func
Task 5
Testing
0 days
10 days
15 days
25 days
30days
45 days

Gantt Chart:


Gantt Chart 1
Task
Task 1
Task 2
Task 3
Task 4
Task 5
Task 6
Task 7
Task 8
Task 9
Task 10
0
10
20
30
40
Start on day