ATM BANKING MACHINE

# OUTLINE:

In this project we are making the entire atm machine in java. Through which a person can easily deposit

and transect all types of amounts and it also show a current and remaining balance as well. In this context we are using file handling, java-fx and simple java Oop.

Specifications:

We begin our design process by presenting a requirements document that specifies the ATM system’s overall purpose and what it must do. Throughout the case study, we refer to the requirements document to determine what functionality the system must include.

A local bank intends to install a new automated teller machine (ATM) to allow users (i.e., bank customers) to perform basic financial transactions. Each user can have only one account at the bank. ATM users should be able to view their account balance, withdraw cash (i.e., take money out of an account) and deposit funds (i.e., place money into an account).

For this we have

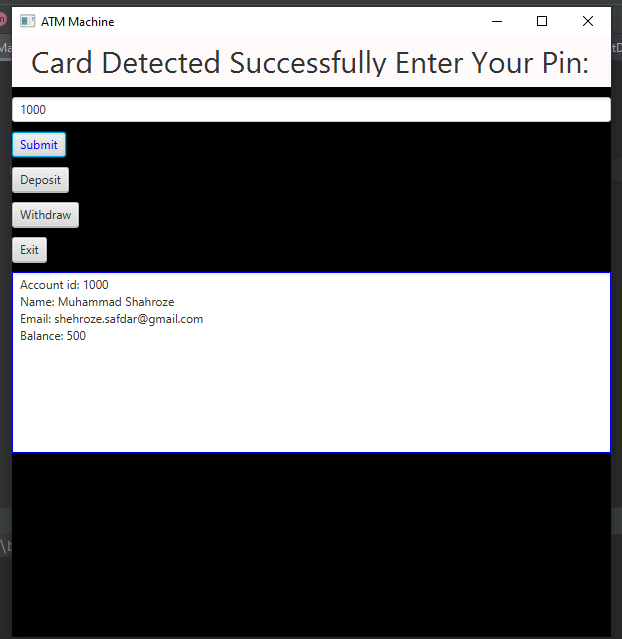
* A keyboard that receives input from user
* Screen that displays messages on the screen
* A cash dispenser that dispenses cash to user
* And a deposit slot

So, on the output screen we have username, password and login button. From which the user input its data and then can access to the Atm machine. Then we make menus from which the user withdraw cash, remain current balance, transaction and deposit slip or balance and exit.

These are the classes for our project

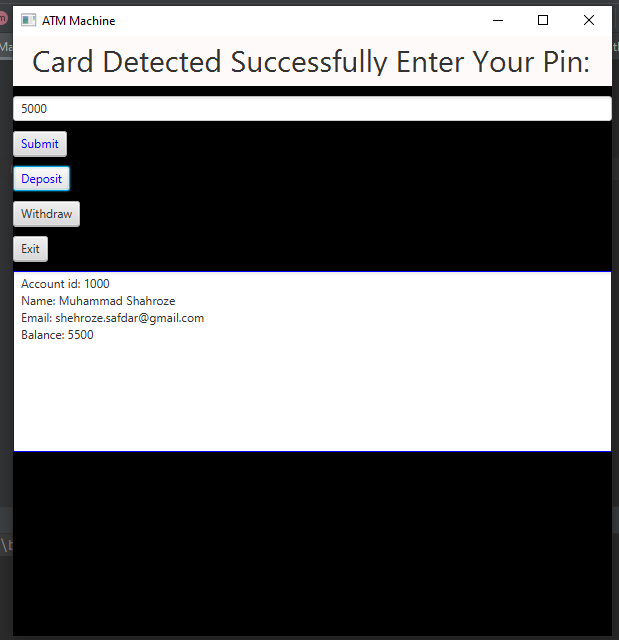
• ATM • screen • keypad • cash dispenser • deposit slot • account • bank database • balance inquiry • withdrawal • deposit

WE MADE A SIMPLE USER INTERFACE BY USING JAVA-FX AND HERE IS THE OUTPUT:



We entered the account id which was already saved in the file by using file handling with all the account details. We can easily withdraw, deposit any amount of cash by using the button of GUI.

We used Java FX without any scene-builder and here is the look of another function of The ATM Machine i.e., deposit.



**CODE FOR THE JAVA FX AND MAIN ACCOUNT CLASS:**

Here is the code for the main Java FX Class and the Main Class of Account that is holding all the data.

package com.shehroze.atm;  
  
import com.shehroze.atm.bank.Bank;  
import javafx.application.Application;  
import javafx.scene.Parent;  
import javafx.scene.Scene;  
import javafx.scene.control.Button;  
import javafx.scene.control.TextArea;  
import javafx.scene.control.TextField;  
import javafx.scene.layout.Background;  
import javafx.scene.layout.VBox;  
import javafx.scene.paint.Color;  
import javafx.scene.text.Font;  
import javafx.stage.Stage;  
  
  
public class CashMachineApp extends Application {  
  
 private TextField field = new TextField();  
 private TextField textField=new TextField();  
  
 private CashMachine cashMachine = new CashMachine(new Bank());  
  
 private Parent createContent() {  
 VBox vbox = new VBox(10);  
 vbox.setPrefSize(600, 600);  
 vbox.setBackground(Background.*fill*(Color.*BLACK*));  
 field.setText("Card Detected Successfully Enter Your Pin:");  
 field.setFont(Font.*font*("Bold", 30));  
 field.setBackground(Background.*fill*(Color.*SNOW*));  
 field.setMaxHeight(50);  
 field.setMinHeight(50);  
  
  
 TextArea areaInfo = new TextArea();  
 areaInfo.setBackground(Background.*fill*(Color.*BLUE*));  
  
  
 Button btnSubmit = new Button("Submit");  
 btnSubmit.setOnAction(e -> {  
 int id = Integer.*parseInt*(textField.getText());  
 cashMachine.login(id);  
  
 areaInfo.setText(cashMachine.toString());  
 btnSubmit.setTextFill(Color.*BLUE*);  
 });  
  
 Button btnDeposit = new Button("Deposit");  
 btnDeposit.setOnAction(e -> {  
 int amount = Integer.*parseInt*(textField.getText());  
 cashMachine.deposit(amount);  
  
 areaInfo.setText(cashMachine.toString());  
 btnDeposit.setTextFill(Color.*BLUE*);  
 });  
  
 Button btnWithdraw = new Button("Withdraw");  
 btnWithdraw.setOnAction(e -> {  
 int amount = Integer.*parseInt*(textField.getText());  
 cashMachine.withdraw(amount);  
  
 areaInfo.setText(cashMachine.toString());  
 btnWithdraw.setTextFill(Color.*BLUE*);  
 });  
  
 Button btnExit = new Button("Exit");  
 btnExit.setOnAction(e -> {  
 cashMachine.exit();  
  
 areaInfo.setText(cashMachine.toString());  
 });  
  
 vbox.getChildren().addAll(field,textField, btnSubmit, btnDeposit, btnWithdraw, btnExit, areaInfo);  
 return vbox;  
 }  
  
 @Override  
 public void start(Stage stage) throws Exception {  
 stage.setTitle("ATM Machine");  
 stage.setScene(new Scene(createContent()));  
 stage.show();  
 }  
  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
}

**CODE FOR MAIN JAVA CLASS (ACCOUNT):**

package com.shehroze.atm.bank;  
  
public abstract class Account {  
  
 private AccountData accountData;  
  
 public Account(AccountData accountData) {  
 this.accountData = accountData;  
 }  
  
 public AccountData getAccountData() {  
 return accountData;  
 }  
  
 public void deposit(int amount) {  
 updateBalance(getBalance() + amount);  
 }  
  
 public boolean withdraw(int amount) {  
 if (canWithdraw(amount)) {  
 updateBalance(getBalance() - amount);  
 return true;  
 } else {  
 return false;  
 }  
 }  
  
 protected boolean canWithdraw(int amount) {  
 return getBalance() >= amount;  
 }  
  
 public int getBalance() {  
 return accountData.getBalance();  
 }  
  
 private void updateBalance(int newBalance) {  
 accountData = new AccountData(accountData.getId(), accountData.getName(), accountData.getEmail(),  
 newBalance);  
 }  
}