A PRELIMINARY REPORT ON

“ATM Interface”

SUBMITTED TO THE EDUBRIDGE INDIA PRIVATE LIMITED

SUBMITTED BY

Miss.Shaila Banu M **Batch No: EON-5755**

Under The Guidance of

Amruta Deore

DEPARTMENT OF S PRAYAS CERTIFIED JAVA FULL STACK DEVELOPER 02

THANE

EDUBRIDGE INDIA PRIVATE LIMITED

2021-2022.

ACKNOWLEDGMENT

It gives all of us great pleasure in presenting the preliminary project report on **“ATM Interface”**. With due respect and gratitude we would like to take this opportunity to thank internal guide of our project **Mrs.Amruta Deore** for giving us all the help and guidance we needed. We are really grateful for his kind support. She has always encouraged us and given us the motivation to move ahead. She has put in a lot of time and effort in this project along with us and given us a lot of confidence. Also we wish to thank all the other people who have helped us in the successful completion of this project.

Miss. Shaila Banu M

ABSTRACT

The ATM(Automatic Teller Machine) Interface is the project which is used to access their bank accounts in order to make cash withdrawals and check balance. There are basically two types of bank accounts namely Checking and Savings Account in which cash operations can be done. In order to make cash operations user needs to enter the correct Account number and PIN (Personal Identification Number) number, after which cash operations are made successful. If the User failed to provide correct Account number and Pin number then automatically "Wrong Account/Pin message" displays. The amount can be withdrawn only if the entered amount is greater than the balance amount. Once the withdrawal was successful, the amount will be debited in their account.

Chapter 1

INTRODUCTION

INTRODUCTION

Automatic Teller Machine enables the clients of a bank to have access to their account without going to the bank. This is achieved with this “ATM Interface” Project. When the project is implemented, the user who uses this project will be able to see all the information and services provided by the ATM, when he enters the necessary options and arguments. The data is stored in memory and retrieved whenever necessary. The program is designed in such a way that user has to enter his/her appropriate account number and pin number. For example, if the user needs to perform withdraw operations, upon the correct entry of Account number and Pin number, it asks the user that which type of account operation he need to perform, there are basically two types of account, Checking and Saving. If the user enters the amount to be withdrawn greater than the balance then an alert message gets generated that “Balance cannot be negative”.

SCOPE

1. Cash Operations can be performed without the need of Bank
2. It is a 24 hours service.
3. The tasks are easy to learn
4. Safe and Secure with Pin number

SYSTEM REQUIREMENTS

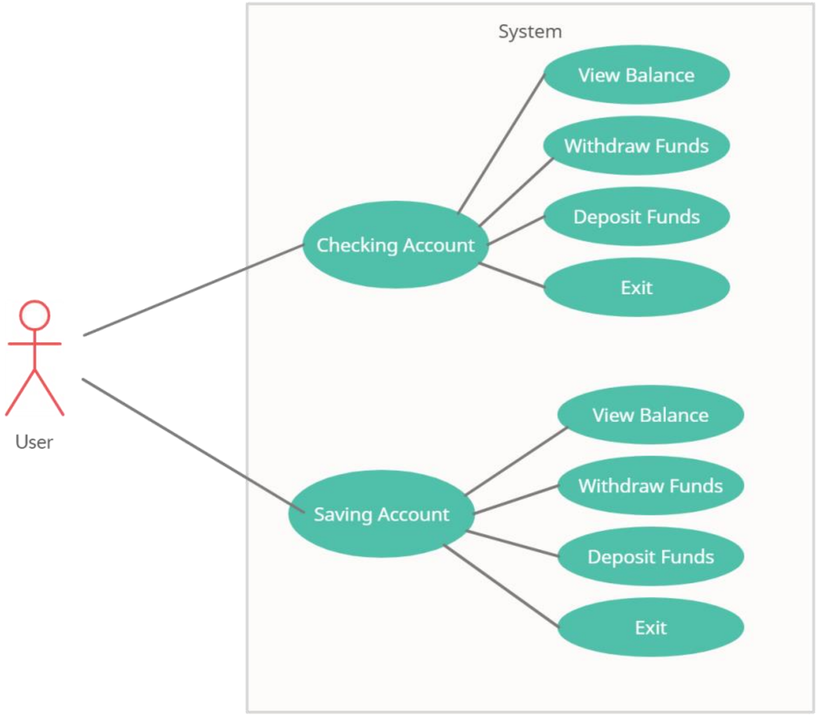
Software Requirements

1. Operating System - Windows 10
2. Platform - Eclipse IDE
3. Software language - Java

UML DIAGRAMS

Use case Diagram

The following UML use case diagram shows the working of a ATM Interface. The user has access to two types of account namely checking and Saving Account portrayed in 2 cases. Moreover, each of it has further 4 use cases that explain the particular functionality of the ATM Interface. The four use cases are; View Balance, Withdraw Funds, Deposit Funds and Exit .The access of User with different cash operations is what sum up the ATM Interface use case diagram.



MODULES

INTRODUCTION:

This module gives the brief view of the further sub modules present inside it. The code is as follows,

```
System.out.println(" *****   Welcome to the ATM Project ***** \n
\n");
System.out.println("Enter your Account Number \n");
setAccNumber(menuInput.nextInt());
System.out.println("Enter your Pin Number \n ");
setPinNumber(menuInput.nextInt());

public void getAccountType(){
                System.out.println();
                System.out.println("Select the Account type you want
to access");
                System.out.println("Type 1. Checking Account");
                System.out.println("Type 2. Saving Account");
                System.out.println("Type 3. Exit \n");
                System.out.println("Your Choice please");
                int selection = menuInput.nextInt();


                switch(selection)
                {
                case 1:
                      getChecking();
                      System.out.println();
                      break;
                case 2:
                      getSaving();
                      System.out.println();
                      break;
                case 3:
                      System.out.println("Thank you for using this
ATM");
                      
                      break;
                }
           }
```

CHECKING ACCOUNT :

```
public void getChecking()
                {
                    System.out.println();
                    System.out.println("Checking Account");
                    System.out.println("Type 1. View Balance");
                    System.out.println("Type 2. Withdraw Funds");
                    System.out.println("Type 3. Deposit Funds");
                    System.out.println("Type 4. Exit \n");
                    System.out.println("Your Choice please");
                    int selection = menuInput.nextInt();

                    switch(selection)
                    {
                    case 1:
                         System.out.println("Checking Account 
Balance: " +moneyFormat.format(getCheckingBalance()));
                         getAccountType();
                         System.out.println();
                         break;
                    case 2:
                         getCheckingWithdrawInput();
                         getAccountType();
                         System.out.println();
                         break;
                    case 3:
                         getCheckingDepositInput();
                         getAccountType();
                         System.out.println();
                         break;
                    case 4:
                         System.out.println("Thank you for using 
this ATM, Bye. \n");

                         System.out.println();
                         break;
                    default:
                         System.out.println("\n Invalid Choice 
\n");

                         getChecking();
                    }
                }
```

SAVING ACCOUNT:

```
public void getSaving()
        {
            System.out.println("Saving Account : ");
            System.out.println("Type 1 : View Balance");
            System.out.println("Type 2 : Withdraw Funds");
            System.out.println("Type 3 : Deposit Funds");
            System.out.println("Type 4 : Exit \n");
            System.out.println("Your Choice please");

            int selection = menuInput.nextInt();
            System.out.println("\n \n");

            switch(selection)
            {
            case 1:
                System.out.println("Saving Account Balance: "
+moneyFormat.format(getSavingBalance()));
                getAccountType();
                System.out.println();
                break;
            case 2:
                getSavingWithdrawInput();
                getAccountType();
                System.out.println();
                break;
            case 3:
                getSavingDepositInput();
                getAccountType();
                System.out.println();
                break;
            case 4:
                System.out.println("Thank you for using this
ATM, Bye. \n");
                System.out.println();
                break;
            default:
                System.out.println("\n Invalid Choice \n");
                getChecking();
        }
    }
}
```

ADVANTAGES

1. Less effort to complete transaction.

2. Quick access to information.

3. Ease of Communication.

4. No need to maintain the bulk of papers.

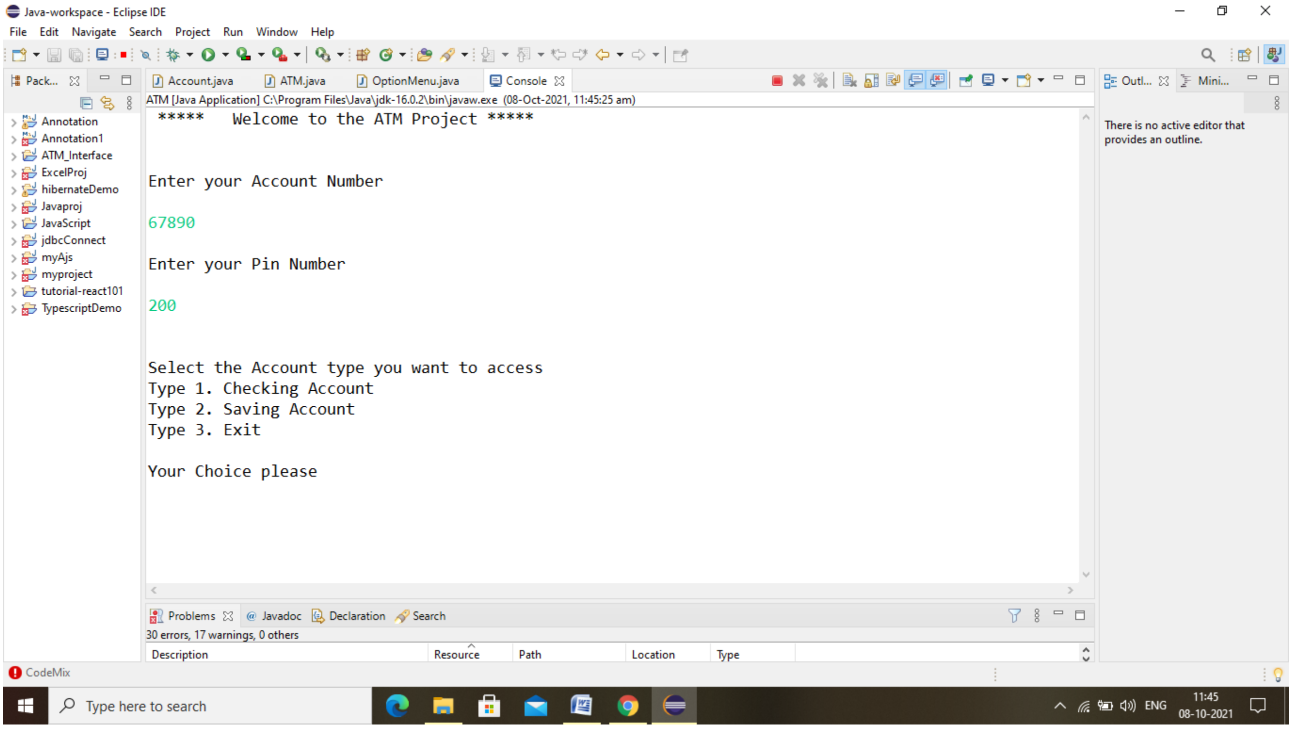
5. Transparency and Control.

Chapter 2

PROJECT IMPLEMENTATION

SCREENS

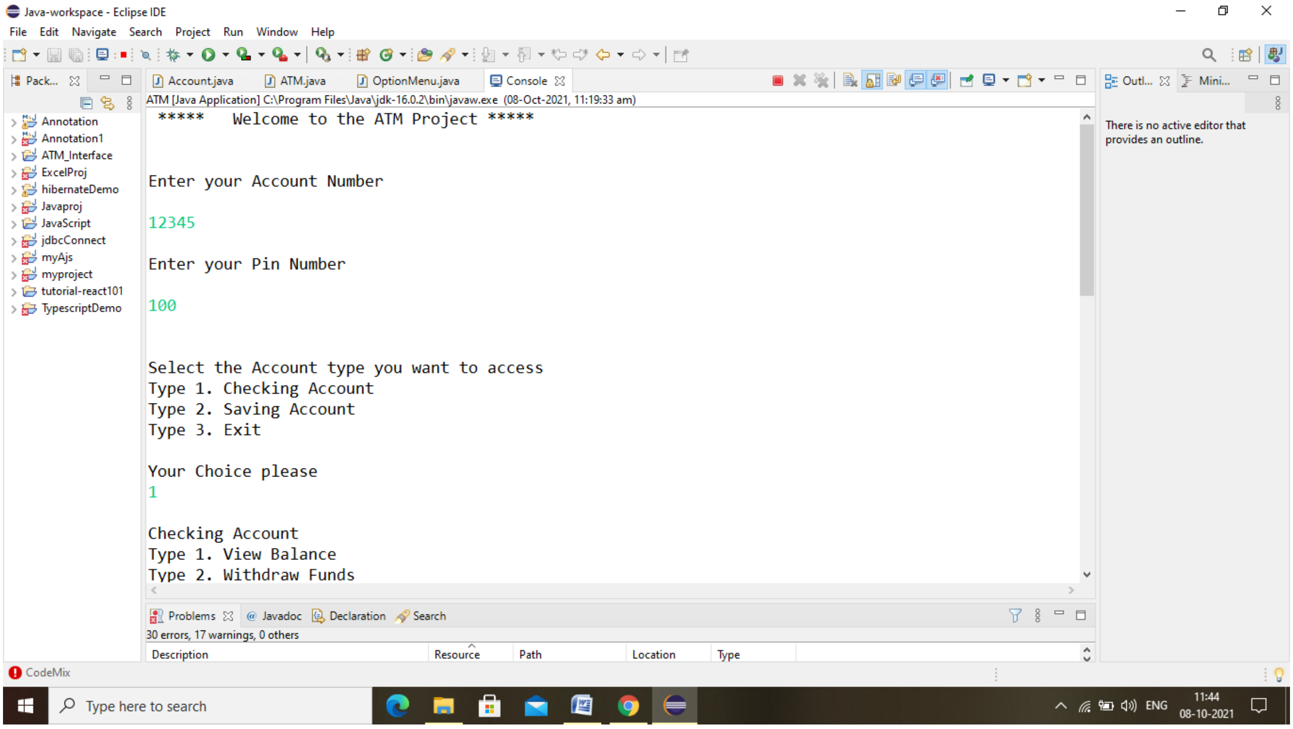
INTRODUCTION



ATM [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe (08-Oct-2021, 11:45:25 am)

```
 *****   Welcome to the ATM Project *****


Enter your Account Number

67890

Enter your Pin Number

200


Select the Account type you want to access
Type 1. Checking Account
Type 2. Saving Account
Type 3. Exit

Your Choice please
```

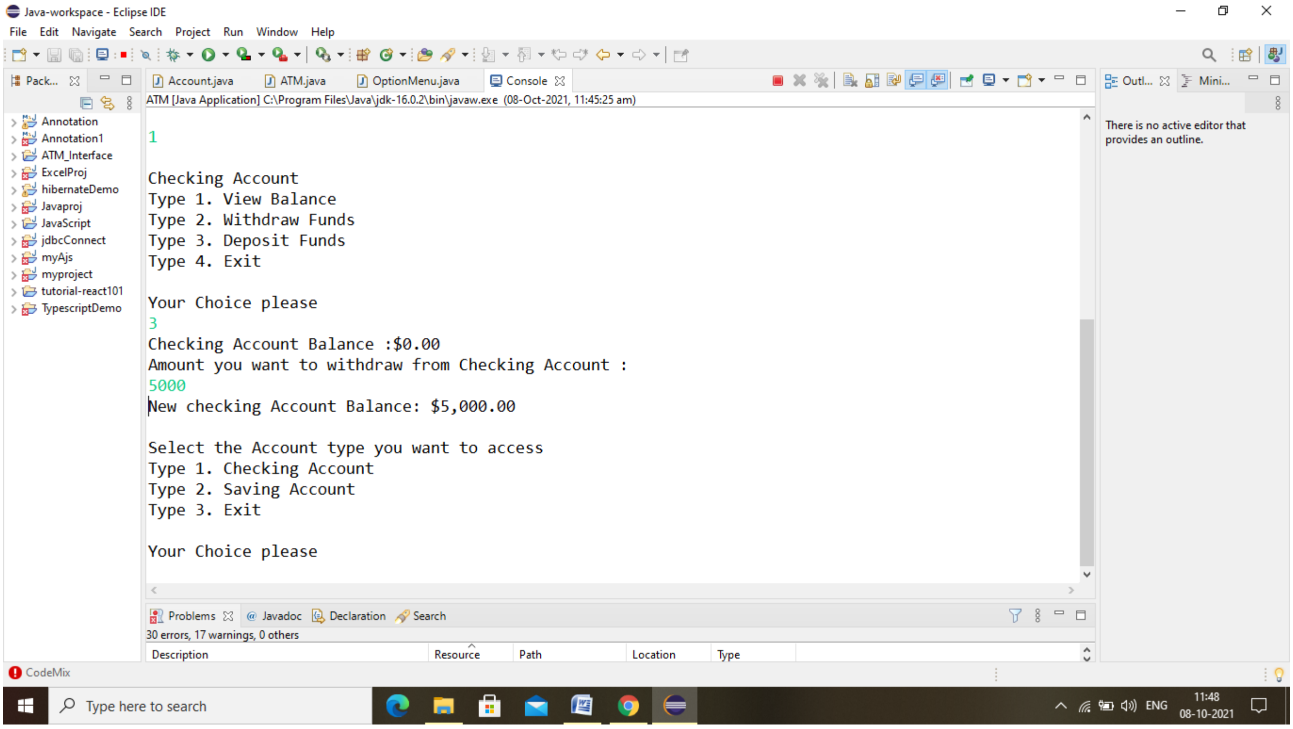
CHECKING ACCOUNT:



ATM [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe (08-Oct-2021, 11:19:33 am)

```
 *****   Welcome to the ATM Project *****


Enter your Account Number

12345

Enter your Pin Number

100


Select the Account type you want to access
Type 1. Checking Account
Type 2. Saving Account
Type 3. Exit

Your Choice please
1

Checking Account
Type 1. View Balance
Type 2. Withdraw Funds
```

30 errors, 17 warnings, 0 others

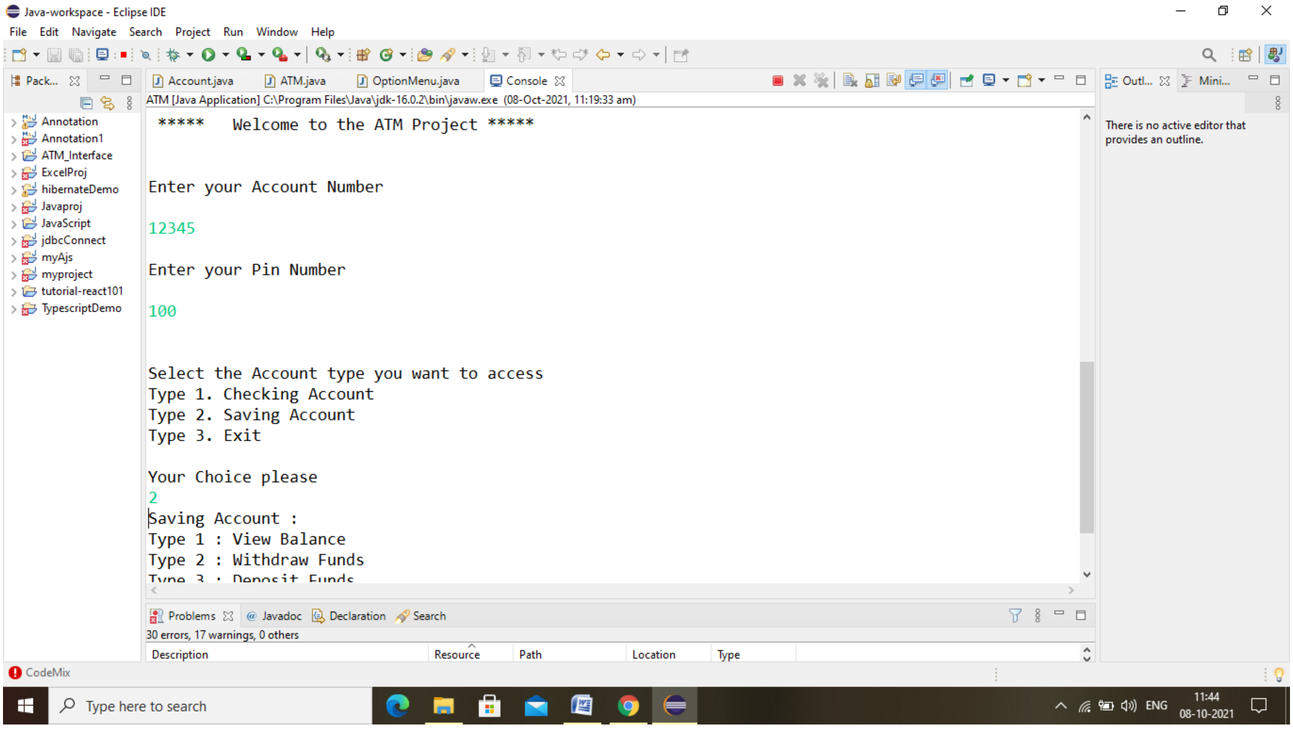


ATM [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe (08-Oct-2021, 11:45:25 am)

```

1

Checking Account
Type 1. View Balance
Type 2. Withdraw Funds
Type 3. Deposit Funds
Type 4. Exit

Your Choice please
3
Checking Account Balance :$0.00
Amount you want to withdraw from Checking Account :
5000
New checking Account Balance: $5,000.00

Select the Account type you want to access
Type 1. Checking Account
Type 2. Saving Account
Type 3. Exit

Your Choice please
```

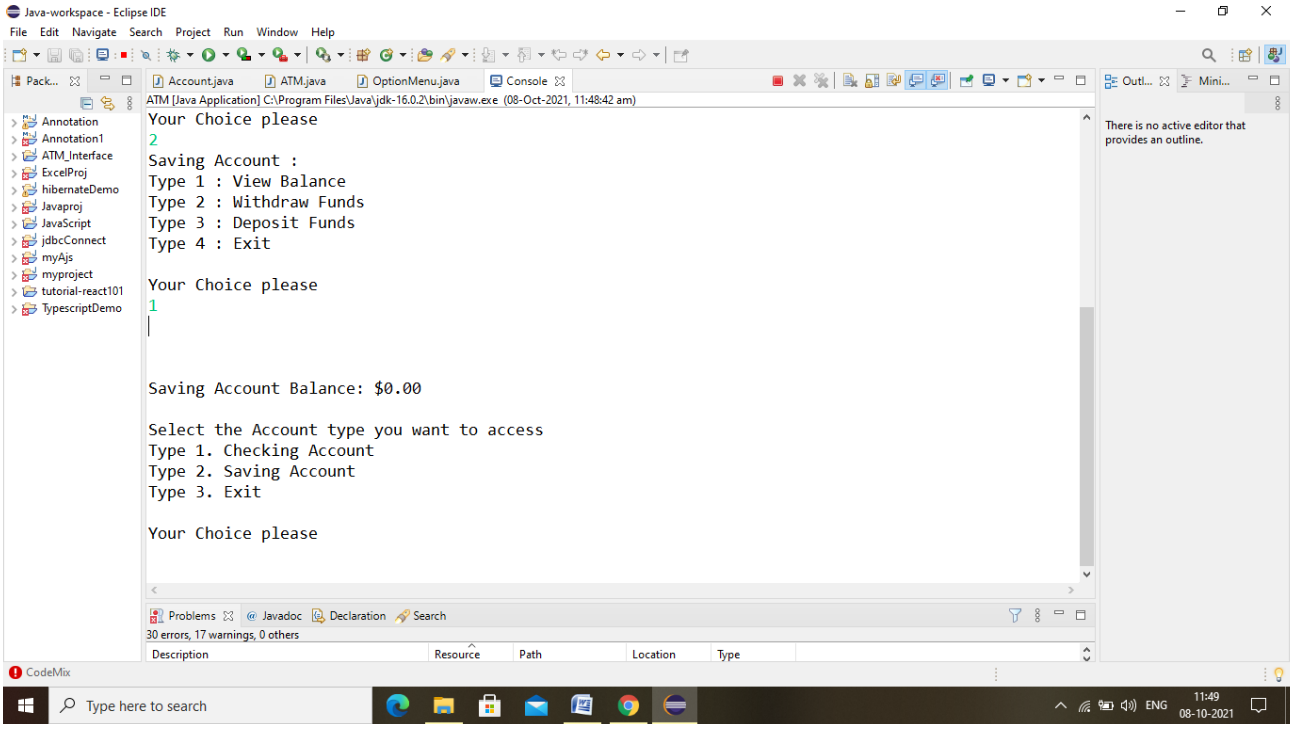
30 errors, 17 warnings, 0 others

SAVING ACCOUNT:



```
ATM [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (08-Oct-2021, 11:19:33 am)
 *****   Welcome to the ATM Project *****


Enter your Account Number

12345

Enter your Pin Number

100


Select the Account type you want to access
Type 1. Checking Account
Type 2. Saving Account
Type 3. Exit

Your Choice please
2
Saving Account :
Type 1 : View Balance
Type 2 : Withdraw Funds
Type 3 : Deposit Funds
```

30 errors, 17 warnings, 0 others



```
ATM [Java Application] C:\Program Files\Java\jdk-16.0.2\bin\javaw.exe  (08-Oct-2021, 11:48:42 am)
Your Choice please
2
Saving Account :
Type 1 : View Balance
Type 2 : Withdraw Funds
Type 3 : Deposit Funds
Type 4 : Exit

Your Choice please
1



Saving Account Balance: $0.00

Select the Account type you want to access
Type 1. Checking Account
Type 2. Saving Account
Type 3. Exit

Your Choice please
```

30 errors, 17 warnings, 0 others

Chapter 3

CONCLUSIONS

3.1 CONCLUSIONS

ATM Interface enables the client of the Bank to access his account without the need of bank and cash operations can be done anytime depending upon the need of the user. Everything in ATM interface is an electronic operation thus it greatly reduces the need of paper record maintenance. Any user knowing the simple language can access cash transactions easily. Thus, all these problems are overcome in “ATM Interface”.