

Laporan Resmi Praktikum

Pemrograman Berbasis Obyek

Projek Akhir



Nama: Alfikiyar Tirta Haidar

Kelas: 2 D4 Teknik Informatika B

NRP: 3120600044

**PROGRAM STUDI TEKNIK INFORMATIKA
DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**

2021

Daftar Isi

1. Daftar Isi.....	i
2. Projek Akhir	
a. Tujuan.....	1
b. Dasar Teori.....	1
c. UML Class Diagram	1
d. Listing Program.....	2
e. Analisa Program.....	12
f. Output Program.....	13

A. Tujuan

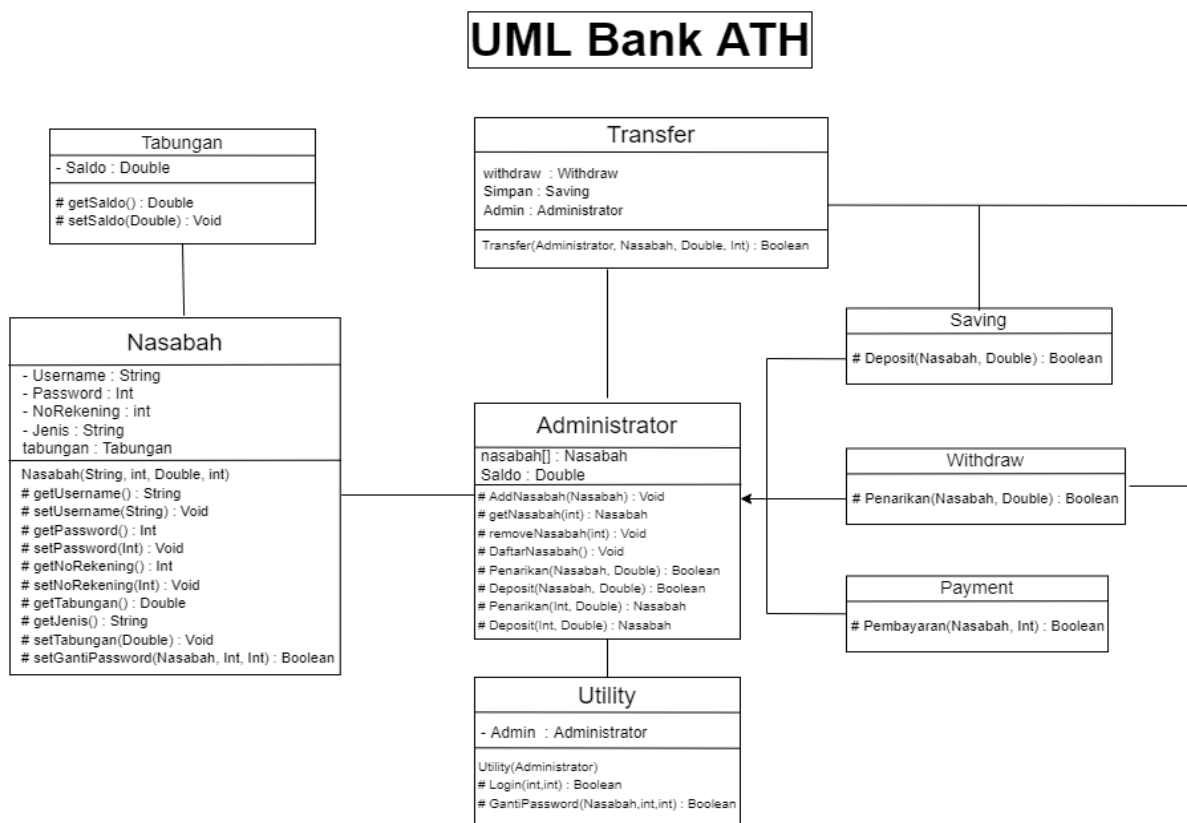
Mahasiswa mampu memecahkan permasalahan menjadi sebuah desain program berorientasi obyek dan kemudian mengimplementasikannya menjadi sebuah program komputer menggunakan bahasa Java, baik secara individu maupun berkelompok/kerjasama tim. Intruksi khususnya adalah :

1. Memahami dan menerapkan konsep *enkapsulasi* dalam simulasi transaksi perbankan.
2. Memahami dan menerapkan konsep *inheritance* dalam simulasi transaksi perbankan.
3. Memahami dan menerapkan konsep *polymorphism* dalam simulasi transaksi perbankan.
4. Memahami dan menerapkan konsep pemrograman Java yang lain dalam simulasi transaksi perbankan.

B. Dasar Teori

Pada mulanya penyediaan ATM (Automatic Teller Machine) ialah untuk memudahkan layanan pengambilan uang dari tabungan nasabah, akan tetapi seiring dengan perkembangan teknologi dan kebutuhan akan peningkatan layanan kepada para nasabah penggunaan ATM telah meluas tidak hanya sebatas pengambilan uang saja. Saat ini sudah memungkinkan bagi para nasabah untuk melakukan transfer (pemindah bukuan) uang, pembayaran, pengecekan saldo, dan transaksi keuangan lainnya cukup dengan menggunakan ATM. Selain itu transaksi di perbankan juga dapat dilakukan di teller. Perbedaannya adalah jika di ATM biasanya diberlakukan limit sesuai dengan jenis kartunya akan tetapi jika di teller tidak dibatasi transaksinya.

C. UML Class Diagram



D. Listing Program

(Main)

```
package FinalProjectNonGui;
import java.util.*;
public class BANK {
    public static void main(String[] args) {
        int PIN,PINBaru>Password;
        String Username>UserAdmin>Jenis;
        int NoRek>NoRekT;
        double Saldo>Nominal;
        int Opsi;
        boolean Status>looping = true;
        Administrator Admin = new Administrator();
        Utility Security = new Utility(Admin);
        Administrator Tarik = new Withdraw();
        Administrator Simpan = new Saving();
        Transfer transfer = new Transfer();
        Payment Bayar = new Payment();
        Nasabah User;
        char Jawab;

        do{
            System.out.println("----- Bank ATH -----");
            System.out.println("1. Login Admin");
            System.out.println("2. Login User");
            System.out.println("3. Keluar");
            System.out.print("Masukkan Pilihan Anda : ");
            Scanner input = new Scanner(System.in);
            Opsi = input.nextInt();
            System.out.println("");
            if(Opsi == 1){
                System.out.print("Masukkan User : ");
                input = new Scanner(System.in);
                UserAdmin = input.next();
                if(UserAdmin.endsWith("HAIDAR")){
                    System.out.print("Masukkan sandi : ");
                    input = new Scanner(System.in);
                    Password = input.nextInt();
                    if(Password == 101101){
                        System.out.println("----- Login Sukses -----\\n\\n");
                        do{
                            System.out.println("----- Menu Admin -----");
                            System.out.println("1. Tambah Akun");
                            System.out.println("2. Hapus Akun");
                            System.out.println("3. Daftar Nasabah");
                            System.out.println("4. Deposit");
                            System.out.println("5. Withdraw");
                            System.out.println("6. Transfer");
                            System.out.println("7. Keluar");
                            System.out.print("Masukkan Pilihan Anda : ");
                            input = new Scanner(System.in);
                            Opsi = input.nextInt();
```

```

switch (Ops) {
    case 1 -> {
        System.out.println("Masukkan Identitas");
        System.out.print("- Nama      : ");
        input = new Scanner(System.in);
        Username = input.nextLine();
        System.out.print("- No. Rekening : ");
        input = new Scanner(System.in);
        NoRek = input.nextInt();
        System.out.print("- Saldo      : Rp. ");
        input = new Scanner(System.in);
        Saldo = input.nextDouble();
        System.out.print("- PIN       : ");
        input = new Scanner(System.in);
        PIN = input.nextInt();
        System.out.println("- Jenis");
        System.out.println("1. Pelajar");
        System.out.println("2. Umum ");
        System.out.print("Pilihan    : ");
        input = new Scanner(System.in);
        if(input.nextInt()==1){
            Jenis = "PELAJAR";
        }else
            Jenis = "UMUM";
        Nasabah Akun = new Nasabah(Username,NoRek,Saldo,PIN,Jenis);
        Admin.addNasabah(Akun);
        System.out.println("--== Tambah Akun Berhasil ==--");
        break;
    }
    case 2 -> {
        System.out.print("No. Rekening yang dihapus : ");
        input = new Scanner(System.in);
        NoRek = input.nextInt();
        User = Admin.getNasabah(NoRek);
        if(User != null){
            Admin.RemoveNasabah(User);
            System.out.println("--== Hapus Akun Berhasil ==--");
        }
        break;
    }
    case 3 -> {
        System.out.println("Nama\tNoRek\ttSaldo\tPIN\tJenis");
        Admin.DaftarNasabah();
        break;
    }
    case 4 ->{
        System.out.print("No. Rekening : ");
        input = new Scanner(System.in);
        NoRek = input.nextInt();
        System.out.print("Nominal    : ");
        input = new Scanner(System.in);
        Nominal = input.nextDouble();
    }
}

```

```

        User = Admin.Deposit(NoRek, Nominal);
        if(User != null){
            System.out.println("---== Deposit Berhasil ===");
        }else{
            System.out.println("Maaf Operasi Invalid");
        }
        break;
    }
    case 5 -> {
        System.out.print("No. Rekening : ");
        input = new Scanner(System.in);
        NoRek = input.nextInt();
        System.out.print("Nominal : ");
        input = new Scanner(System.in);
        Nominal = input.nextDouble();
        User = Admin.Penarikan(NoRek, Nominal);

        if(User != null){
            System.out.println("---== Withdraw Berhasil ===");
        }else{
            System.out.println("Maaf Operasi Invalid");
        }
        break;
    }
    case 6 -> {
        System.out.print("No. Rekening : ");
        input = new Scanner(System.in);
        NoRek = input.nextInt();
        User = Admin.getNasabah(NoRek);
        System.out.print("Nominal : ");
        input = new Scanner(System.in);
        Nominal = input.nextDouble();
        System.out.print("Rekening Tujuan : ");
        input = new Scanner(System.in);
        NoRekT = input.nextInt();
        Status = transfer.Transfer(Admin, User, Nominal, NoRekT);
        if(Status == true){
            System.out.println("---== Transfer Berhasil ===");
        }else{
            System.out.println("Maaf Operasi Invalid");
        }
        break;
    }
    default -> {
        looping = false;
        break;
    }
}
System.out.println("");
}while(looping);
}else{
    System.out.println("Maaf Sandi Invalid");
}

```

```

    }

    }else{
        System.out.println("Maaf User tidak terdaftar");
    }
}

else if(Opsi == 2){
    System.out.print("Masukkan No Rekening : ");
    input = new Scanner(System.in);
    NoRek = input.nextInt();
    System.out.print("Masukkan PIN      : ");
    input = new Scanner(System.in);
    PIN = input.nextInt();
    Status = Security.Login(NoRek, PIN);
    if(Status == true){
        User = Admin.getNasabah(NoRek);
        System.out.println("---- Login Sukses ----\n\n");
        do{
            System.out.println("----- Menu Nasabah -----");
            System.out.println("1. Info Saldo");
            System.out.println("2. Deposit");
            System.out.println("3. Withdraw");
            System.out.println("4. Transfer");
            System.out.println("5. Pembayaran");
            System.out.println("6. Ubah PIN");
            System.out.println("7. Keluar");
            System.out.print("Masukkan Pilihan Anda : ");
            input = new Scanner(System.in);
            Opsi = input.nextInt();
            switch (Opsi) {
                case 1 -> {
                    System.out.println("----- Info Saldo -----");
                    System.out.println("Nama   : "+User.getUsername());
                    System.out.println("No. Rek : "+User.getNoRekening());
                    System.out.println("Saldo  : RP. "+User.getTabungan());
                    break;
                }
                case 2 -> {
                    System.out.println("----- Deposit -----");
                    System.out.print("Masukkan Nominal : ");
                    input = new Scanner(System.in);
                    Nominal = input.nextDouble();
                    if("PELAJAR".equals(User.getJenis())){
                        Status = Admin.Deposit(User, Nominal);
                    }else
                        Status = Simpan.Deposit(User, Nominal);

                    if(Status == true){
                        System.out.println("----- Deposit Berhasil -----");
                    }else{
                        System.out.println("Maaf Operasi Invalid");
                    }
                }
            }
        }while(true);
    }
}

```

```

        break;
    }
    case 3 -> {
        System.out.println("----- Withdraw -----");
        System.out.print("Masukkan Nominal : ");
        input = new Scanner(System.in);
        Nominal = input.nextDouble();
        if("PELAJAR".equals(User.getJenis())){
            Status = Admin.Penarikan(User, Nominal);
        }else
            Status = Tarik.Penarikan(User, Nominal);

        if(Status == true){
            System.out.println("----- Withdraw Berhasil -----");
        }else{
            System.out.println("Maaf Operasi Invalid");
        }
        break;
    }
    case 4 -> {
        System.out.println("----- Transfer -----");
        System.out.print("Masukkan Nominal : ");
        input = new Scanner(System.in);
        Nominal = input.nextDouble();
        System.out.print("Masukkan Rekening Tujuan : ");
        input = new Scanner(System.in);
        int NoRekTujuan = input.nextInt();
        Status = transfer.Transfer(Admin, User, Nominal, NoRekTujuan);
        if(Status == true){
            System.out.println("----- Transfer Berhasil -----");
        }else{
            System.out.println("Maaf Operasi Invalid");
        }
        break;
    }
    case 5 -> {
        System.out.println("----- Pembayaran -----");
        System.out.println("1. Listrik (Rp. 23.000)");
        System.out.println("2. Air (Rp. 16.000)");
        System.out.println("3. Gas (Rp. 13.000)");
        System.out.println("4. Lainnya (Rp. 11.000)");
        System.out.print("Masukkan Pilihan Anda : ");
        input = new Scanner(System.in);
        Opsi = input.nextInt();
        Status = Bayar.Pembayaran(User, Opsi);
        if(Status == true){
            System.out.println("----- Pembayaran Berhasil -----");
        }else{
            System.out.println("Maaf Operasi Invalid");
        }
        break;
    }
}

```



```

        case 6 -> {
            System.out.println("---=== Ubah PIN ===---");
            System.out.print("Masukkan PIN Baru : ");
            input = new Scanner(System.in);
            PINBaru = input.nextInt();
            PIN = User.getPassword();
            Status = User.setGantiPassword(User, PINBaru,PIN);
            if(Status == true){
                System.out.println("----= PIN Update =---");
            }else{
                System.out.println("Maaf PIN Tidak Berubah");
            }
            break;
        }
        default -> {
            looping = false;
            break;
        }
    }
    System.out.println("");
}while(looping);
}else{
    System.out.println("Maaf No Rekening atau PIN tidak valid");
}
}
else
    break;
System.out.println("");
looping = true;
}while(true);
System.out.println("--== Terima Kasih atas Kepercayaan Anda ==--");
}
}

```

(Class Administrator)

```

package FinalProjectNonGui;
import java.util.ArrayList;
public class Administrator {
    ArrayList<Nasabah> nasabah = new ArrayList<>();
    double Saldo;

    protected void addNasabah(Nasabah member){
        nasabah.add(member);
    }
    protected Nasabah getNasabah(int NoRek){
        Nasabah User = null;
        for(Nasabah C : nasabah){
            if(C.getNoRekening() == NoRek)
                User = C;
        }
        return User;
    }
}

```

```

protected void RemoveNasabah(Nasabah member){
    nasabah.remove(member);
}
protected void DaftarNasabah(){
    nasabah.forEach(C -> {

System.out.println(C.getUsername()+"\t"+C.getNoRekening()+"\t"+C.getTabungan()+"\t"+C.getPassw
ord()+"\t"+C.getJenis());
    });
}
protected boolean Penarikan(Nasabah Member,double Nominal){
    Nasabah User = Member;
    Saldo = User.getTabungan();
    if(Saldo - Nominal > 0){
        User.setTabungan(Saldo - Nominal);
        return true;
    }
    else
        return false;
}
protected boolean Deposit(Nasabah Member, double Nominal){
    Nasabah User = Member;
    Saldo = User.getTabungan();
    Saldo += Nominal;
    User.setTabungan(Saldo);
    return true;
}
protected Nasabah Deposit(int Norek,double Nominal){
    Nasabah User = null;
    User = getNasabah(Norek);
    Saldo = User.getTabungan();
    Saldo += Nominal;
    User.setTabungan(Saldo);
    return User;
}
protected Nasabah Penarikan(int Norek,double Nominal){
    Nasabah User = null;
    User = getNasabah(Norek);
    Saldo = User.getTabungan();
    if(User != null){
        if(Saldo - Nominal > 0){
            User.setTabungan(Saldo - Nominal);
            return User;
        }
        else
            return null;
    }else
        return null;
}
}

```

(Class Nasabah)

```
package FinalProjectNonGui;
public class Nasabah {
    private String Username;
    private int Password;
    private int NoRekening;
    private String Jenis;
    Tabungan tabungan = new Tabungan();

    Nasabah(String User,int NoRek,double Saldo,int PIN,String Jenis){
        this.Username = User;
        this.NoRekening = NoRek;
        this.tabungan.setSaldo(Saldo);
        this.Password = PIN;
        this.Jenis = Jenis;
    }
    Nasabah(String User){
        this.Username = User;
    }
    protected String getUsername(){
        return Username;
    }
    protected void setUsername(String Username){
        this.Username = Username;
    }
    protected int getPassword(){
        return Password;
    }
    protected void setPassword(int PIN){
        this.Password = PIN;
    }
    protected int getNoRekening(){
        return NoRekening;
    }
    protected double getTabungan(){
        return tabungan.getSaldo();
    }
    protected String getJenis(){
        return Jenis;
    }
    protected void setTabungan(double Saldo){
        tabungan.setSaldo(Saldo);
    }
    protected boolean setGantiPassword(Nasabah member,int PINBaru,int PINLama){
        boolean Status;
        Nasabah User = member;

        if(PINBaru == PINLama)
            Status = false;
        else{
            User.Password = PINBaru;
            Status = true;
        }
    }
}
```

```

    }
    return Status;
}
}

```

(Class Payment)

```

package FinalProjectNonGui;
public class Payment extends Administrator{
    protected boolean Pembayaran(Nasabah Member,int Opsi){
        Nasabah User = Member;
        Withdraw withdraw = new Withdraw();
        boolean status;
        super.Saldo = User.getTabungan();
        status = switch (Opsi) {
            case 1 -> withdraw.Penarikan(User,23000);
            case 2 -> withdraw.Penarikan(User,16000);
            case 3 -> withdraw.Penarikan(User,13000);
            case 4 -> withdraw.Penarikan(User,11000);
            default -> false;
        };
        return status;
    }
}

```

(Class Saving)

```

package FinalProjectNonGui;
public class Saving extends Administrator{
    protected boolean Deposit(Nasabah Member, double Nominal){
        Nasabah User = Member;
        super.Saldo = User.getTabungan();
        super.Saldo = super.Saldo + Nominal*1.1;
        User.setTabungan(super.Saldo);
        return true;
    }
}

```

(Class Tabungan)

```

package FinalProjectNonGui;
public class Tabungan{
    private double Saldo;

    protected double getSaldo(){
        return Saldo;
    }
    protected void setSaldo(double saldo){
        this.Saldo = saldo;
    }
}

```

(Class Transfer)

```

package FinalProjectNonGui;

```

```

public class Transfer{
    Withdraw withdraw = new Withdraw();
    Saving Simpan = new Saving();
    Administrator Admin;
    boolean Transfer(Administrator Admin,Nasabah Member, double Nominal, int NoRekTujuan){
        this.Admin = Admin;
        Nasabah User = Member;
        int NoRek = User.getNoRekening();
        boolean Status;
        Status = withdraw.Penarikan(User, Nominal);
        if(Status == true){
            User = Admin.getNasabah(NoRekTujuan);
            if(User == null || NoRek == NoRekTujuan){
                Status = false;
            }else{
                Status = Admin.Deposit(User, Nominal);
            }
        }
        return Status;
    }
}

```

(Class Utility)

```

package FinalProjectNonGui;
public class Utility{
    Administrator Admin;
    Utility(Administrator Admin){
        this.Admin = Admin;
    }

    protected boolean Login(int NoRek,int PIN){
        boolean Kondisi;
        Nasabah User;
        User = Admin.getNasabah(NoRek);
        if(User != null){
            if(User.getPassword() == PIN)
                Kondisi = true;
            else
                Kondisi = false;
        }else
            Kondisi = false;
        return Kondisi;
    }

    protected boolean GantiPassword(Nasabah member,int PINLama,int PINBaru){
        Nasabah User;
        User = member;
        boolean Kondisi;
        if(User.getPassword() == PINLama){
            Kondisi = false;
        }else{
            User.setPassword(PINBaru);
            Kondisi = true;
        }
    }
}

```

```

    }
    return Kondisi;
}
}

(Class Withdraw)
package FinalProjectNonGui;
public class Withdraw extends Administrator{
    protected boolean Penarikan(Nasabah Member,double Nominal){
        Nasabah User = Member;
        super.Saldo = User.getTabungan();
        if(super.Saldo - Nominal - 1500 > 0){
            User.setTabungan(super.Saldo - Nominal - 1500);
            return true;
        }
        else
            return false;
    }
}
}

```

E. Analisa Program

1. Class Administrator

Class Administrator berfungsi sebagai inti dari class lainnya. Contohnya adalah untuk menyimpan daftar Nasabah yang terdaftar, menambahkan dan menghapus Nasabah, serta didalamnya terdapat dua method seperti penarikan dan deposit dengan penerapan **polimorfisme static**. Hal itu karena method yang pertama digunakan untuk operasi pada teller (admin) dan satunya untuk operasi ATM dengan parameter dan return value yang berbeda pula tentunya. Selain itu, class administrator adalah parent class dari class Payment, Withdraw, dan Saving yang menggunakan **polimorfisme dinamis**. Hal itu karena ada method di class Administrator yang dioverride oleh child classnya.

2. Class Nasabah

Class Nasabah memiliki fungsi utama untuk mengelola informasi yang ada di tiap nasabahnya, seperti Username, PIN, No Rekening, Jenis Bank, dan Saldo. Didalamnya menerapkan proses enkapsulasi dimana semua atribut beripe private dan bisa diakses melalui perantara sebuah method.

3. Class Payment

Cass Payment adalah class yang khusus untuk menangani proses pembayaran dimana pada class ini **mengextends** class Administrator (menjadi child class) agar dapat mengakses atribut Saldo dan method yang ada di parentnya.

4. Class Saving

Class Saving adalah class yang khusus untuk menangani proses deposit untuk jenis akun “UMUM”, dimana class ini menjadi **child class** dari class Administrator dan melakukan **overriding** pada method Deposit yang ada di class parentnya. Sifat dari override yang

dilakukan class Saving adalah mengembangkan argument yang ada pada method parentnya, dimana proses deposit pada akun jenis “UMUM” akan dikenakan biaya tertentu dalam proses transaksi apapun.

5. Class Withdraw

Class Withdraw adalah class yang berfungsi untuk proses penarikan pada jenis akun “UMUM”, sama halnya dengan class Saving yang mana merupakan **child class** dari class Administrator dan melakukan **overriding** pada method Penarikan yang ada di class parentnya.

6. Class Tabungan

Class Tabungan adalah class yang khusus untuk menangani Saldo dari tiap nasabah, baik itu untuk melakukan set Saldo maupun get Saldo. Didalamnya class ini memanfaatkan proses **enkapsulasi** karena atribut Saldo di class ini bertipe private.

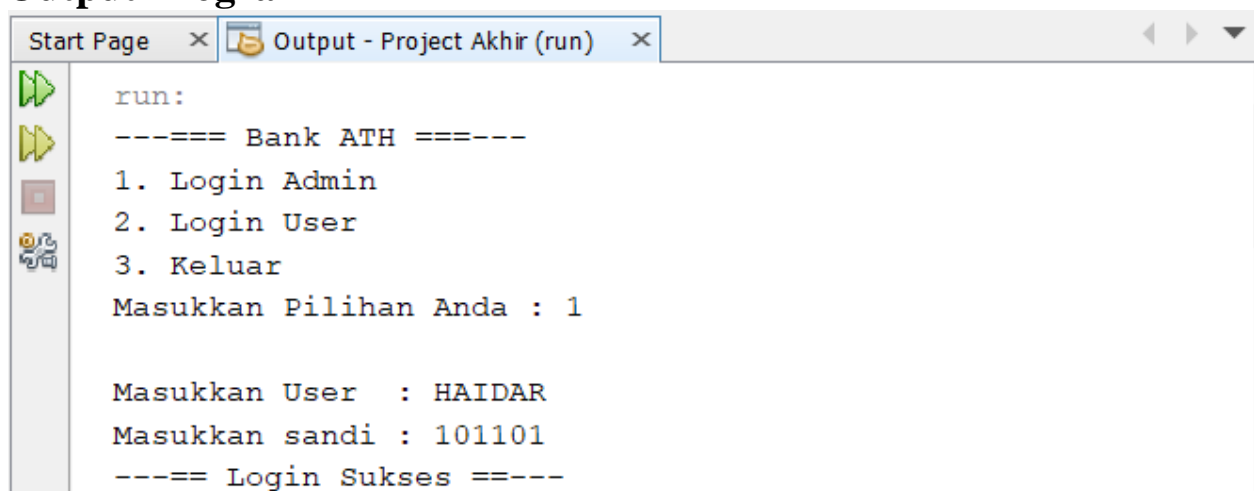
7. Class Transfer

Class Transfer adalah class yang khusus untuk menangani proses transfer antar nasabah ke nasabah, sehingga dibutuhkan method yang berfungsi menyimpan semua daftar tabungan untuk dilakukan filter dan menangkap atribut yang bertipe Administrator. Dimana algoritma yang diterapkan di class ini adalah menggunakan proses Withdraw seperti pada class Withdraw dan proses Saving seperti pada method deposit yang ada di class Administrator.

8. Utility

Class Utility adalah class yang berfungsi khusus untuk mengelola proses Login pada ATM dan penanganan masalah PIN, seperti ubah PIN.

F. Output Program

The image shows a screenshot of an IDE's output window. The window has two tabs: 'Start Page' and 'Output - Project Akhir (run)'. The 'Output' tab is active, displaying the following text:

```
run:
----- Bank ATH -----
1. Login Admin
2. Login User
3. Keluar
Masukkan Pilihan Anda : 1

Masukkan User   : HAIDAR
Masukkan sandi  : 101101
----- Login Sukses -----
```

```
----- Menu Admin -----  
1. Tambah Akun  
2. Hapus Akun  
3. Daftar Nasabah  
4. Deposit  
5. Withdraw  
6. Transfer  
7. Keluar  
Masukkan Pilihan Anda : 1  
Masukkan Identitas  
- Nama : Tirta  
- No. Rekening : 10110111  
- Saldo : Rp. 100000  
- PIN : 111111  
- Jenis  
1. Pelajar  
2. Umum  
Pilihan : 1  
----- Tambah Akun Berhasil -----
```

```
----- Menu Admin -----  
1. Tambah Akun  
2. Hapus Akun  
3. Daftar Nasabah  
4. Deposit  
5. Withdraw  
6. Transfer  
7. Keluar  
Masukkan Pilihan Anda : 1  
Masukkan Identitas  
- Nama : Agus  
- No. Rekening : 10110122  
- Saldo : Rp. 75000  
- PIN : 222222  
- Jenis  
1. Pelajar  
2. Umum  
Pilihan : 2
```


--== Tambah Akun Berhasil ==--

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Tirta	10110111	100000.0		111111 PELAJAR
Agus	10110122	75000.0	222222	UMUM

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 2

No. Rekening yang dihapus : 10110111

--== Hapus Akun Berhasil ==--

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	75000.0	222222	UMUM

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 1

Masukkan Identitas

- Nama : A'at
 - No. Rekening : 10110133
 - Saldo : Rp. 60000
 - PIN : 333333
 - Jenis
1. Pelajar
 2. Umum

Pilihan : 1

==== Tambah Akun Berhasil =====

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	75000.0	222222	UMUM
A'at	10110133	60000.0	333333	PELAJAR

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit

5. Withdraw

6. Transfer

7. Keluar

Masukkan Pilihan Anda : 4

No. Rekening : 10110122

Nominal : 15000

----- Deposit Berhasil -----

----- Menu Admin -----

1. Tambah Akun

2. Hapus Akun

3. Daftar Nasabah

4. Deposit

5. Withdraw

6. Transfer

7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	90000.0	222222	UMUM
A'at	10110133	60000.0	333333	PELAJAR

----- Menu Admin -----

1. Tambah Akun

2. Hapus Akun

3. Daftar Nasabah

4. Deposit

5. Withdraw

6. Transfer

7. Keluar

Masukkan Pilihan Anda : 5

No. Rekening : 10110133

Nominal : 13000

----- Withdraw Berhasil -----

----- Menu Admin -----

1. Tambah Akun

2. Hapus Akun

3. Daftar Nasabah

4. Deposit

5. Withdraw

6. Transfer

7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	90000.0	222222	UMUM
A'at	10110133	47000.0	333333	PELAJAR

----- Menu Admin -----

1. Tambah Akun

2. Hapus Akun

3. Daftar Nasabah

4. Deposit

5. Withdraw

6. Transfer

7. Keluar

Masukkan Pilihan Anda : 6

No. Rekening : 10110122

Nominal : 27000

Rekening Tujuan : 10110133

----- Transfer Berhasil -----

----- Menu Admin -----

1. Tambah Akun

2. Hapus Akun

3. Daftar Nasabah

4. Deposit

5. Withdraw

6. Transfer

7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	61500.0	222222	UMUM
A'at	10110133	74000.0	333333	PELAJAR

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 6

No. Rekening : 10110122

Nominal : 27000

Rekening Tujuan : 10110133

----- Transfer Berhasil -----

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 3

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	61500.0	222222	UMUM
A'at	10110133	74000.0	333333	PELAJAR

----- Menu Admin -----

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

Masukkan Pilihan Anda : 7

----- Bank ATH -----

1. Login Admin
2. Login User
3. Keluar

Masukkan Pilihan Anda : 1

Masukkan User : 10110122

Maaf User tidak terdaftar

----- Bank ATH -----

1. Login Admin
2. Login User
3. Keluar

Masukkan Pilihan Anda : 2

Masukkan No Rekening : 10110122

Masukkan PIN : 222222

----- Login Sukses -----

----- Menu Nasabah -----

1. Info Saldo
2. Deposit
3. Withdraw
4. Transfer
5. Pembayaran
6. Ubah PIN
7. Keluar

Masukkan Pilihan Anda : 1

----- Info Saldo -----

Nama : Agus

No. Rek : 10110122

Saldo : RP. 61500.0

----- Menu Nasabah -----

1. Info Saldo
2. Deposit

```

3. Withdraw
4. Transfer
5. Pembayaran
6. Ubah PIN
7. Keluar
Masukkan Pilihan Anda : 2
----- Deposit -----
Masukkan Nominal : 30000
----- Deposit Berhasil -----

----- Menu Nasabah -----
1. Info Saldo
2. Deposit
3. Withdraw
4. Transfer
5. Pembayaran
6. Ubah PIN
7. Keluar
Masukkan Pilihan Anda : 3
----- Withdraw -----
Masukkan Nominal : 15000
----- Withdraw Berhasil -----

----- Menu Nasabah -----
1. Info Saldo
2. Deposit
3. Withdraw
4. Transfer
5. Pembayaran
6. Ubah PIN
7. Keluar
Masukkan Pilihan Anda : 1
----- Info Saldo -----
Nama      : Agus
No. Rek   : 10110122
Saldo     : RP. 78000.0

```

```
----- Menu Nasabah -----  
1. Info Saldo  
2. Deposit  
3. Withdraw  
4. Transfer  
5. Pembayaran  
6. Ubah PIN  
7. Keluar  
Masukkan Pilihan Anda : 4  
----- Transfer -----  
Masukkan Nominal : 20000  
Masukkan Rekening Tujuan : 10110133  
----- Transfer Berhasil -----  
  
----- Menu Nasabah -----  
1. Info Saldo  
2. Deposit  
3. Withdraw  
4. Transfer  
5. Pembayaran  
6. Ubah PIN  
7. Keluar  
Masukkan Pilihan Anda : 5  
----- Pembayaran -----  
1. Listrik (Rp. 23.000)  
2. Air (Rp. 16.000)  
3. Gas (Rp. 13.000)  
4. Lainnya (Rp. 11.000)  
Masukkan Pilihan Anda : 2  
----- Pembayaran Berhasil -----  
  
----- Menu Nasabah -----  
1. Info Saldo  
2. Deposit  
3. Withdraw  
4. Transfer  
5. Pembayaran  
6. Ubah PIN
```



```
7. Keluar
Masukkan Pilihan Anda : 1
----- Info Saldo -----
Nama      : Agus
No. Rek   : 10110122
Saldo     : RP. 39000.0

----- Menu Nasabah -----
1. Info Saldo
2. Deposit
3. Withdraw
4. Transfer
5. Pembayaran
6. Ubah PIN
7. Keluar
Masukkan Pilihan Anda : 6
----- Ubah PIN -----
Masukkan PIN Baru : 123456
---- PIN Update ----

----- Menu Nasabah -----
1. Info Saldo
2. Deposit
3. Withdraw
4. Transfer
5. Pembayaran
6. Ubah PIN
7. Keluar
Masukkan Pilihan Anda : 7

----- Bank ATH -----
1. Login Admin
2. Login User
3. Keluar
Masukkan Pilihan Anda : 1
```

```
Masukkan User : HAIDAR
Masukkan sandi : 101101
----== Login Sukses ==----
```

```
-----== Menu Admin ==-----
```

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

```
Masukkan Pilihan Anda : 3
```

Nama	NoRek	Saldo	PIN	Jenis
Agus	10110122	39000.0	123456	UMUM
A'at	10110133	94000.0	333333	PELAJAR

```
-----== Menu Admin ==-----
```

1. Tambah Akun
2. Hapus Akun
3. Daftar Nasabah
4. Deposit
5. Withdraw
6. Transfer
7. Keluar

```
Masukkan Pilihan Anda : 7
```

```
-----== Bank ATH ==-----
```

1. Login Admin
2. Login User
3. Keluar

```
Masukkan Pilihan Anda : 3
```

```
---== Terima Kasih atas Kepercayaan Anda ===--
```

```
BUILD SUCCESSFUL (total time: 6 minutes 33 seconds)
```