

**LAPORAN PRAKTIKUM**  
**METODOLOGI PENGEMBANGAN**  
**PERANGKAT LUNAK**  
**MODUL 9**



**Versi 3.1**

**Penyusun : Tim Dosen MPPL**

**Nama : Kharisma Maulida Saara**

**Tarum Widyasti Pertiwi**

**Vania Rahma Dewi**

**NIM : 064002200024**

**064002200027**

**064002200030**

**Jurusan Teknik Informatika**

**Fakultas Teknologi Informasi**

**Universitas Trisakti**

**2022**

**MODUL 9 : Pelaksanaan Tahapan Implementasi  
dan Pengujian Perangkat Lunak**

**Pokok Bahasan:**

Pelaksanaan tahapan implementasi dan pengujian perangkat lunak.

**Kode Pokok Bahasan:**

IKS324.PRAK.2019.05

Melaksanakan tahapan implementasi perangkat lunak dengan melakukan instalasi dan menggunakan Eclipse IDE.

IKS324.PRAK.2019.06

Melaksanakan tahapan pengujian perangkat lunak.

No	Kemampuan Akhir Yang Diharapkan	Kriteria Penilaian dan Indikator	Jml Menit	Bobot Nilai (%)
1	– Mahasiswa mampu mengoperasikan IDE dengan benar ( <b>Sub CPMK 2.4</b> ).	<b>Kriteria penilaian :</b> Rubrik deskriptif.  <b>Indikator :</b> Ketepatan waktu dalam menyelesaikan <b>Laporan Praktikum 9</b> .	85	1,5
<b>TOTAL</b>			<b>85</b>	<b>1,5</b>

**TUGAS PENDAHULUAN**

Untuk dapat menjalankan modul praktikum ini silahkan membaca artikel berikut :

1. Eclipse
2. *Software testing*

**LAB SETUP**

Untuk dapat menjalankan praktikum ini maka harus disiapkan peralatan dan tempat pengambilan data sebagai berikut :

1. Studi kasus pengembangan perangkat lunak pada suatu organisasi / perusahaan / institusi
2. Eclipse

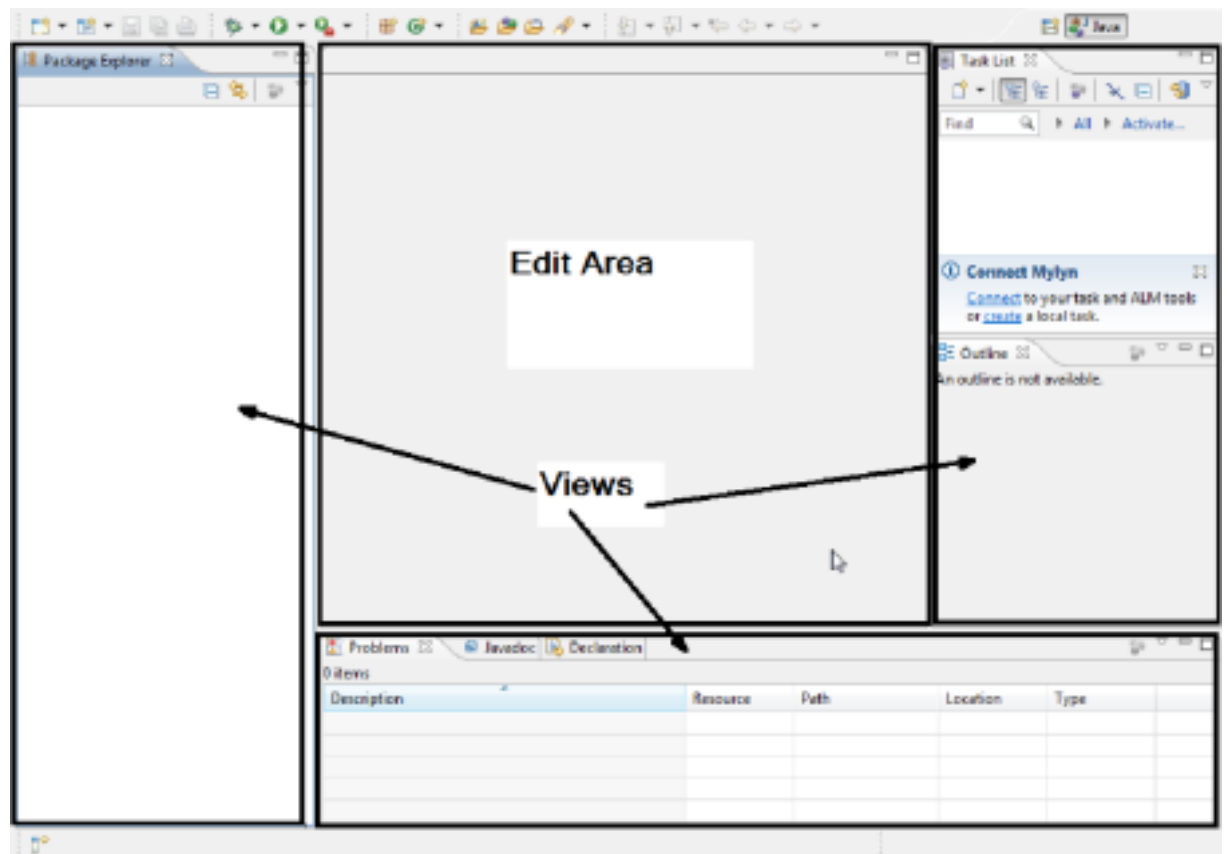
Eclipse merupakan sebuah *development environment* yang sangat *user-friendly* untuk membuat program JAVA. Eclipse sudah banyak dikenal dengan baik oleh banyak programmer JAVA. dan digunakan untuk berbagai pengembangan aplikasi. Sehingga mempelajari Eclipse dapat menambah wawasan dan pengetahuan tentang pemrograman JAVA.

Eclipse dikembangkan oleh Open Source Community. Proyek Eclipse dikelola oleh Eclipse Foundation. Untuk mengoperasikan Eclipse, dibutuhkan Java Runtime Environment (JRE). Eclipse berisikan berbagai macam komponen dan *plugin*, termasuk *JAVA compiler*.

Berikut tahapan untuk mengoperasikan Eclipse :

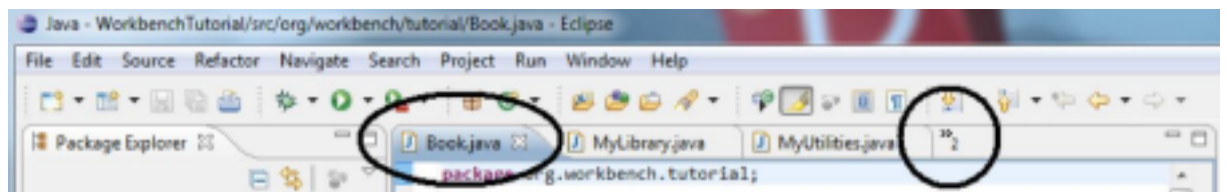
1. Pada komputer berbasisan Windows, *double-click* pada *file* eclipse.exe.  
Sedangkan pada komputer berbaskisan Linux atau Mac, *double-click* pada *file* eclipse.
2. Ketika muncul *dialog box*, masukkan *pathname* untuk *workspace* (digunakan untuk menyimpan *JAVA Project*), dan tekan tombol **OK**.
3. Eclipse akan muncul dan menampilkan **Display Page**.
4. Tutup **Display Page**,

Eclipse menyediakan sebua **Edit Area** dan **View**. **Edit Area** digunakan untuk memasukkan *JAVA source code*. Sedangkan **View** merupakan *sub-window* yang menampilkan informasi tentang *project*.



Gambar Edit Area dan View

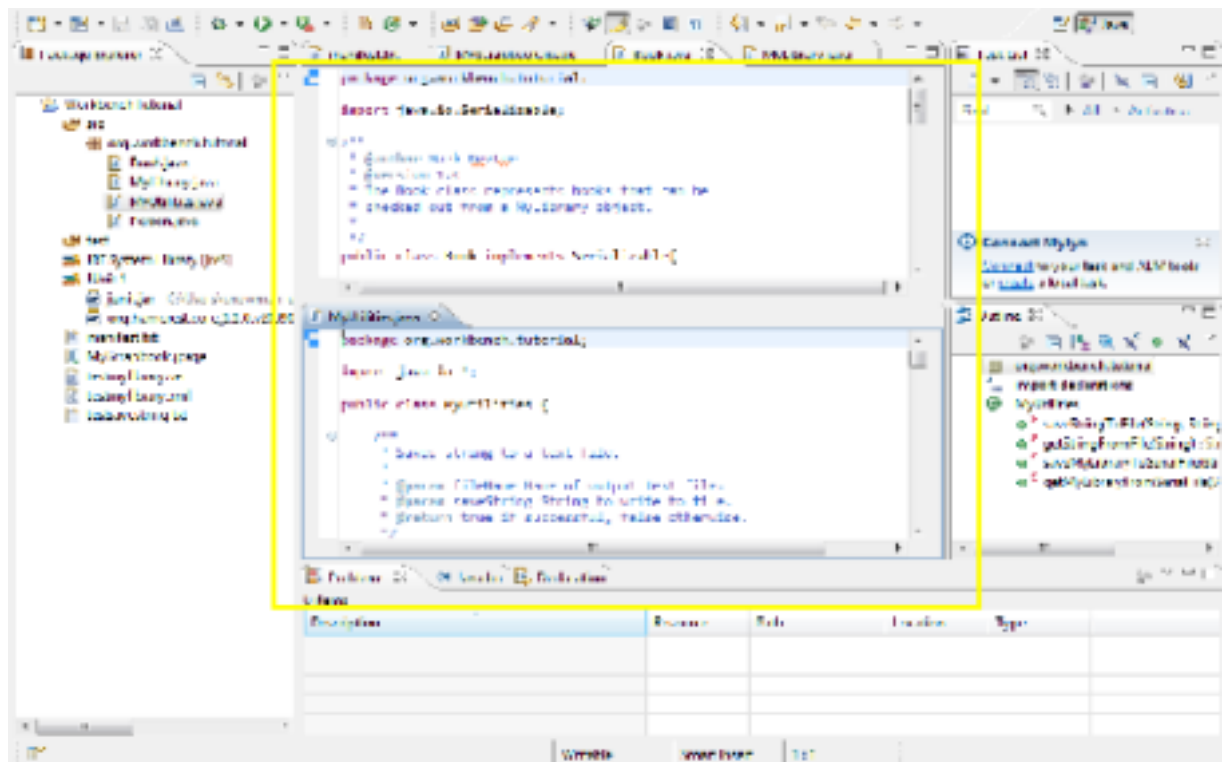
**Edit Area** menggunakan tab jika Anda membuka lebih dari 1 *file*.



**Gambar Tab Pada Edit Area**

**Edit Area** dapat memiliki *multiple windows* pada *space* yang tersedia.

4

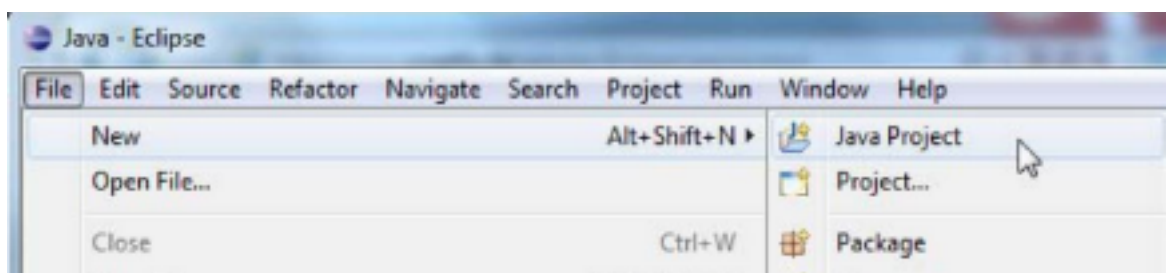


**Gambar Multiple Windows Pada Edit Area**

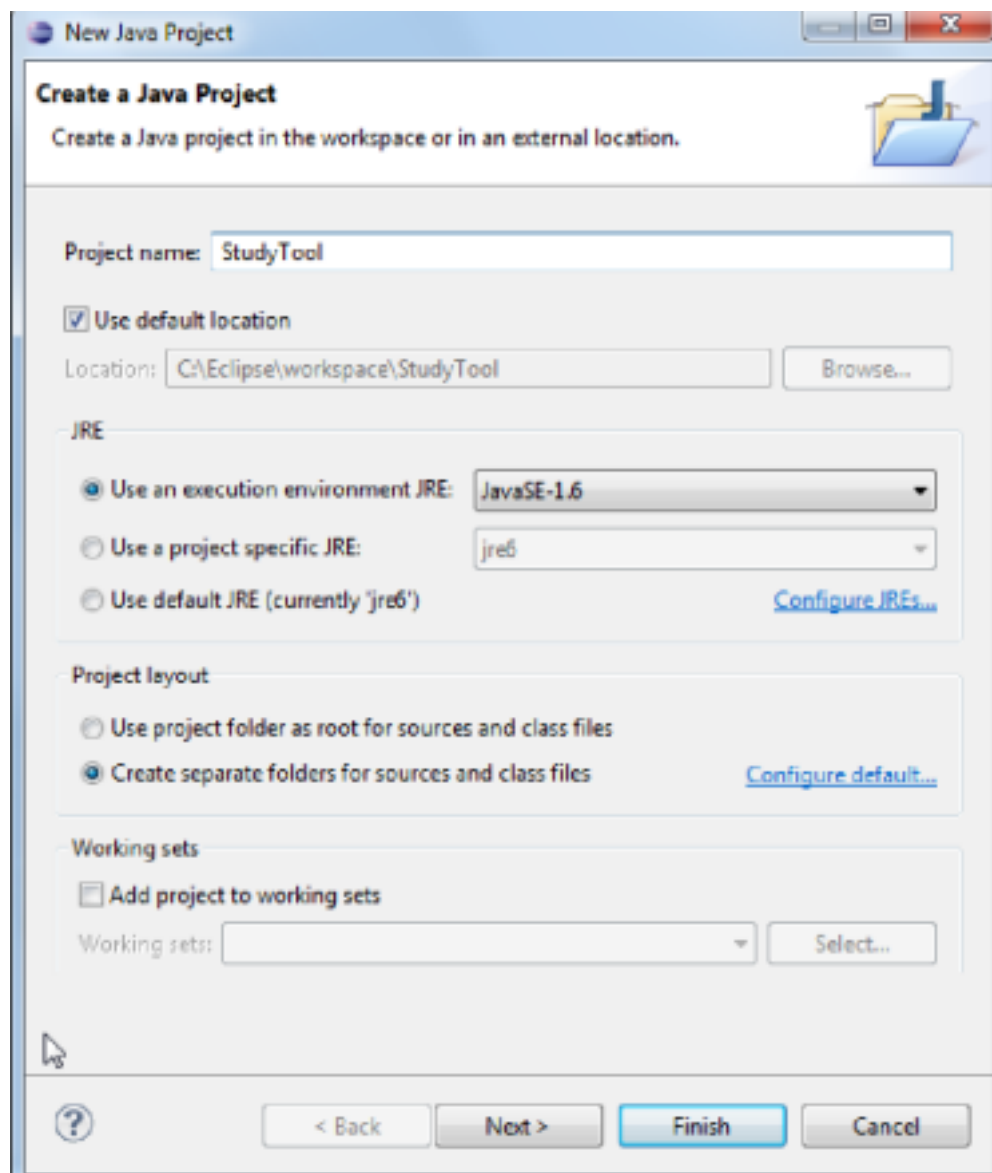
Langkah untuk membuat program JAVA :

1. Buat **Project**

- Pilih **File** → **New** → **Java Project**

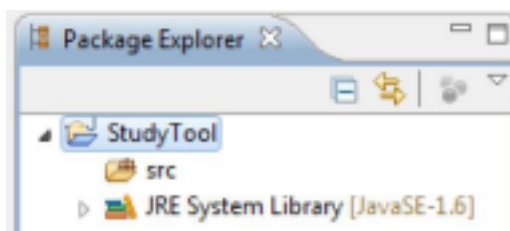


- Masukkan **Project Name** dan klik **Finish**



6

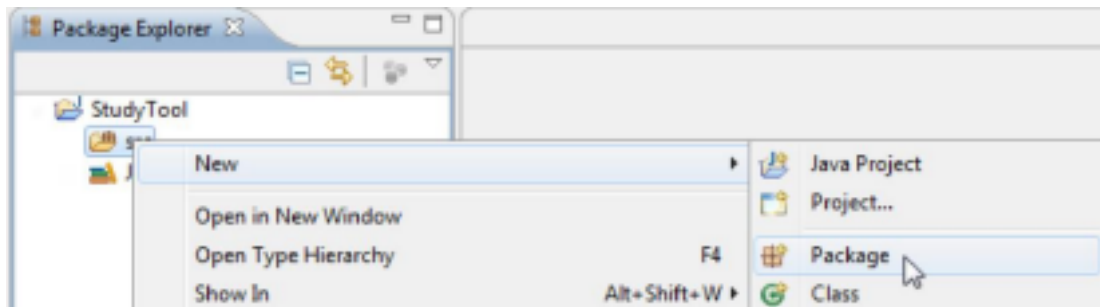
- Seluruh informasi tentang *project* di-set ke **Default** dan klik **Finish**
- Setelah itu, *project* sudah selesai dibuat dan ditampilkan sebagai folder
- **Package view** dapat dilihat di sebelah kiri dari **Edit Area**



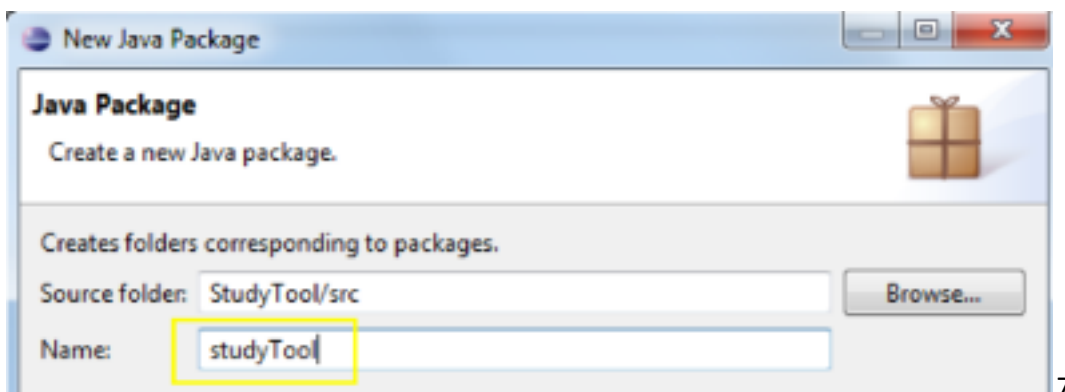
## 2. Buat **Package**

- Pada tab **Package Explorer**, pilih **<Nama Project>**, lalu klik kanan dan pilih **New** →

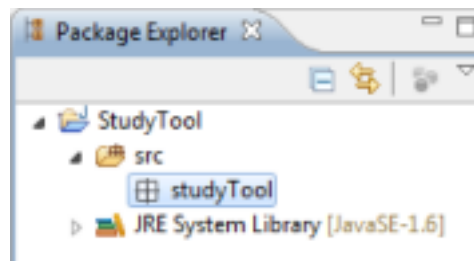
## Package



- Ketikkan nama *package* dan klik **Finish**

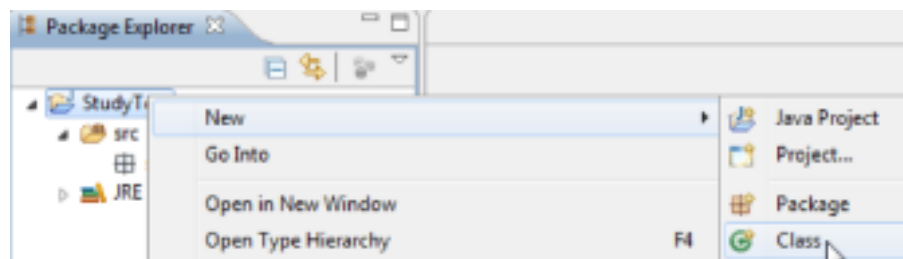


- Sehingga *package* tersebut akan tampil pada **Package Explorer**



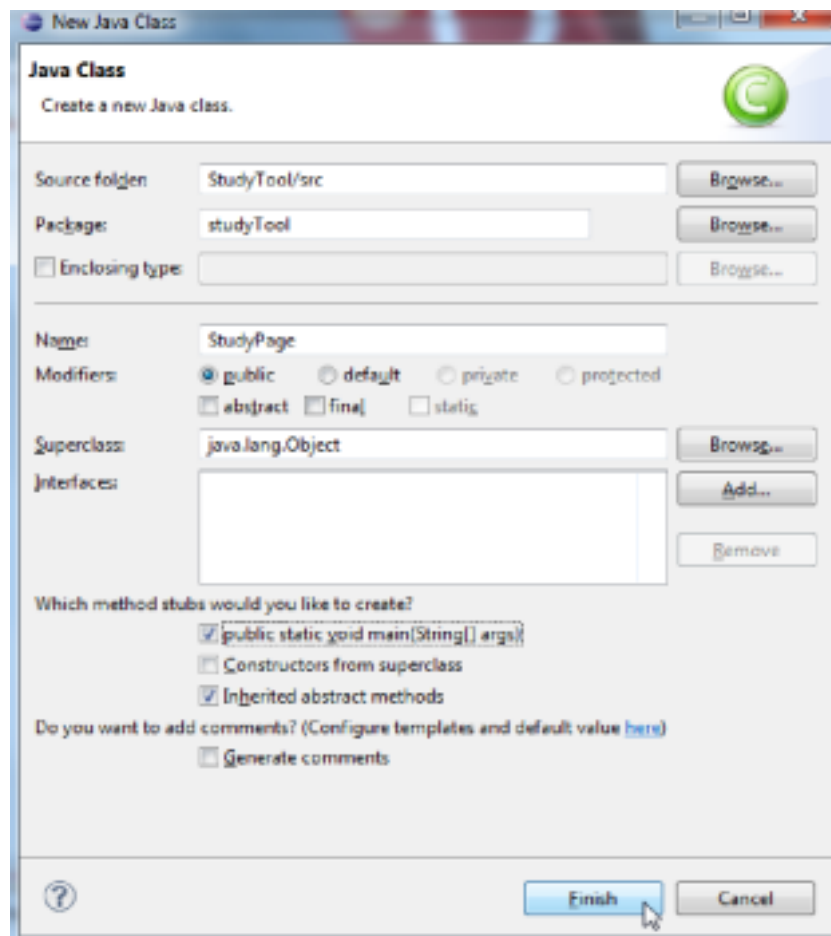
### 3. Pilih **Class**

- Klik kanan pada **<Nama Project>** dan pilih **New → Class**



- Ketikkan nama **Class** dan klik **Finish**





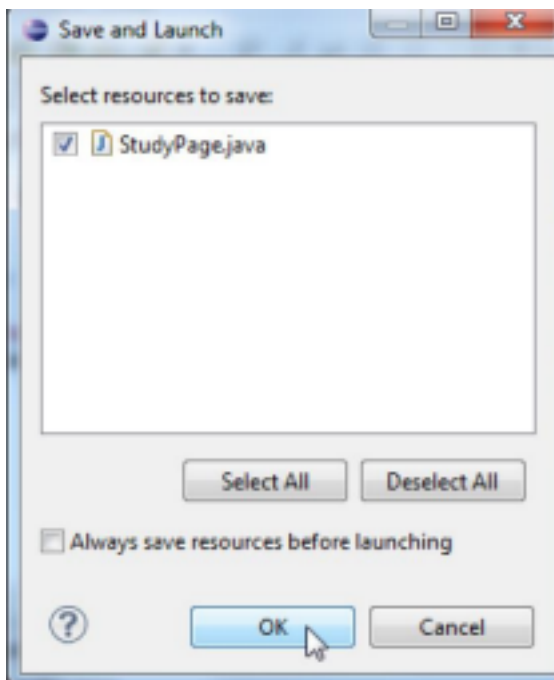
8

#### 4. Create and run JAVA code

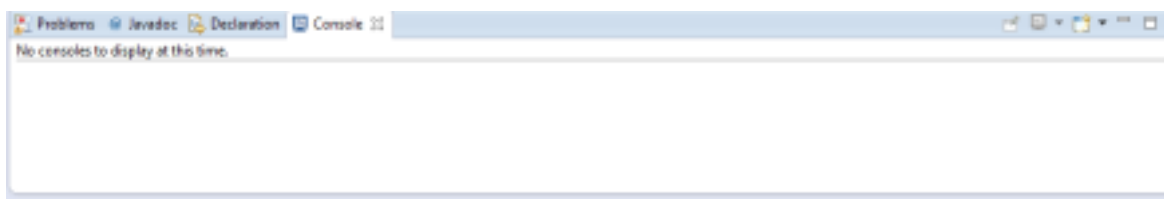
- Ketikkan *source code* pada **Main Class**

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println("Test Java Code");    }  
}
```

- Klik kanan pad *file* .java
- Pilih **Run As** → **Java Application**
- Simpan **Class** jika ditanyakan dengan mengklik **OK**



- Hasil akan ditampilkan pada **Console View**



9

## TUGAS

Pelajari studi kasus. Kemudian lengkapi isian berikut ini.

### Latihan 9.1. Membuat *Class*

File : registktv.java

**NOTE** : Extension diisi dengan .java

```
import java.util.Scanner;
```

```
public class registktv {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Selamat datang! Silakan lakukan
registrasi akun.");
        System.out.print("Masukkan Nama lengkap: "); String
namalengkap = scanner.nextLine();

        System.out.print("Masukkan Alamat email: "); String
email = scanner.nextLine();

        System.out.print("Masukkan No telepon: ");
        String notelepon = scanner.nextLine();
    }
}
```

```

    System.out.print("Masukkan Nama pengguna: "); String
username = scanner.nextLine();

    System.out.print("Masukkan Password: ");
    String password = scanner.nextLine();

    System.out.print("Masukkan Ulang password: "); String
confirmPassword = scanner.nextLine();

    boolean isRegistered = registerAccount(namalengkap, email,
notelepon, username, password, confirmPassword);

    if (isRegistered) {
        System.out.println("Registrasi akun berhasil!"); } else {
        System.out.println("Registrasi akun gagal. Silakan periksa
kembali data yang dimasukkan.");
    }
    scanner.close();
}

private static boolean registerAccount(String namalengkap, String
email, String notelepon, String username, String password, String
confirmPassword) {
    // Lakukan validasi terhadap data yang dimasukkan if
(username.isEmpty() || email.isEmpty() ||
notelepon.isEmpty() || password.isEmpty() ||
confirmPassword.isEmpty()) {
        System.out.println("Maaf, semua kolom harus diisi!"); return
false;
    } else if (!password.equals(confirmPassword)) {

```

```

System.out.println("Maaf, password tidak cocok!"); return
false;
} else if (password.length() < 8) {
System.out.println("Maaf, password harus memiliki minimal 8
karakter!");
return false;
}

// Lakukan proses registrasi akun
// Kode di sini akan mencakup logika untuk menyimpan data ke
database atau melakukan tindakan lain yang diperlukan

return true;
}
}

```

Output :

```

Selamat datang! Silakan lakukan registrasi akun.
Masukkan Nama lengkap: Kharisma Tarum Vania
Masukkan Alamat email: KTV123@gmail.com
Masukkan No telepon: 089876543212
Masukkan Nama pengguna: Khatava123
Masukkan Password: Halloiniaku123
Masukkan Ulang password: Halloiniaku123
Registrasi akun berhasil!
|

```



**NOTE :** Extension diisi dengan .java

```
import java.util.Scanner;

public class LoginLaundryKTV {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Data akun yang telah terdaftar
        String username = "user123";
        String password = "password123";

        System.out.print("Masukkan nama pengguna: ");
        String inputUsername = scanner.nextLine();

        System.out.print("Masukkan password: ");
        String inputPassword = scanner.nextLine();

        boolean isLoggedIn = login(inputUsername,
            inputPassword, username, password);

        if (isLoggedIn) {
            System.out.println("Login berhasil! Selamat datang, "
+ inputUsername + "!");
        } else {
            System.out.println("Login gagal. Nama pengguna atau
password tidak valid.");
        }

        scanner.close();
    }

    private static boolean login(String inputUsername,
        String inputPassword, String username, String password) {
        // Lakukan validasi login
        if (inputUsername.equals(username) &&
            inputPassword.equals(password)) {
            return true;
        } else {
            return false;
        }
    }
}
```

Output:



File : PemesananlaundryKTV.java

**NOTE:** Extension diisi dengan .java

```
import java.util.Scanner;

public class PemesananlaundryKTV {
    private String laundryType;
    private String laundryMethod;
    private String pickupLocation;
    private String paymentMethod;

    public void setLaundryType(String type) {
        this.laundryType = type;
    }

    public void setLaundryMethod(String method) {
        this.laundryMethod = method;
    }

    public void setPickupLocation(String location) {
        this.pickupLocation = location;
    }

    public void setPaymentMethod(String method) {
        this.paymentMethod = method;
    }

    public void placeOrder() {
        System.out.println("Detail Pemesanan laundry:");
        System.out.println("Jenis Laundry: " + laundryType);
        System.out.println("Metode Laundry: " + laundryMethod);
        System.out.println("Lokasi Antarkan: " + pickupLocation);
        System.out.println("Metode Pembayaran: " + paymentMethod);
        System.out.println("Pesanan telah diterima. Terima kasih!");
    }

    public static void main(String[] args) {
        PemesananlaundryKTV order = new PemesananlaundryKTV();
        Scanner scanner = new Scanner(System.in);

        System.out.println("Pemesanan Laundry");
        System.out.println("=====");
        System.out.println("Jenis Laundry: ");
        System.out.println("1. Cuci Lipat");
        System.out.println("2. Cuci Kering");
        System.out.println("3. Cuci Setrika");
        int laundryTypeChoice = scanner.nextInt();
        scanner.nextLine();

        switch (laundryTypeChoice) {
            case 1:
                order.setLaundryType("Cuci Lipat"); break;
            case 2:
                order.setLaundryType("Cuci Kering"); break;
            case 3:
                order.setLaundryType("Cuci Setrika");
        }
    }
}
```

```

    break;
    default:
        System.out.println("Pilihan tidak valid. Pemesanan
dibatalkan.");
        scanner.close();
        return;
    }

    System.out.println("Metode Laundry: ");
    System.out.println("1. Express");
    System.out.println("2. Regular");
    int laundryMethodChoice = scanner.nextInt();
    scanner.nextLine();

    switch (laundryMethodChoice) {
        case 1:
            order.setLaundryMethod("Express");
            break;
        case 2:
            order.setLaundryMethod("Regular");
            break;
        default:
            System.out.println("Pilihan tidak valid. Pemesanan
dibatalkan.");
            scanner.close();
            return;
    }

    System.out.print("Lokasi Antarkan: ");
    String pickupLocation = scanner.nextLine();
    order.setPickupLocation(pickupLocation);

    System.out.println("Metode Pembayaran: ");
    System.out.println("1. Cash");
    System.out.println("2. L-pay");
    System.out.println("3. M-Bank");
    System.out.println("4. QRIS");
    System.out.println("5. Dana");
    int paymentMethodChoice = scanner.nextInt();
    scanner.nextLine();

    switch (paymentMethodChoice) {
        case 1:
            order.setPaymentMethod("Cash");
            break;
        case 2:
            order.setPaymentMethod("L-pay");
            break;
        case 3:
            order.setPaymentMethod("M-Bank");
            break;
        case 4:
            order.setPaymentMethod("QRIS");
            break;
        case 5:
            order.setPaymentMethod("Dana");
            break;
    }

```

**default:**

14

```
System.out.println("Pilihan tidak valid. Pemesanan dibatalkan.");
scanner.close();
return;
}

order.placeOrder();
scanner.close();
}
}
```

Output:



15

File : antarjemputlaundryKTV.java

**NOTE** : Extension diisi dengan .java

```
import java.util.Scanner;
```

```
public class antarjemputlaundryKTV {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        String namaPelanggan = "Vania rahma";
```



```

String lokasi = "Jl. Bumyagara";
String metodePembayaran = "M-banking";

System.out.println("Nama pelanggan: " + namaPelanggan);
System.out.println("Lokasi: " + lokasi);
System.out.println("Pembayaran: " + metodePembayaran);
System.out.print("Apakah Anda ingin menerima pesanan?
(ya/tidak): ");
String response = scanner.nextLine();

if (response.equalsIgnoreCase("ya")) {
    System.out.println("Pesanan kamu diterima! Driver menuju ke
lokasimu");
    // Tambahkan logika untuk menghubungi tim antar jemput laundry
} else if (response.equalsIgnoreCase("tidak")) {
    System.out.println("Driver menolak pesananmu. Kami akan mencari
driver baru. Terima kasih.");
} else {
    System.out.println("Respon tidak valid. Pesanan ditolak.");
}
}
}

```

Output :



File : statuspemesananKTV.java

**NOTE :** Extension diisi dengan .java

```

public class statuspemesananKTV {
    public static void main(String[] args) {
        // Status pemesanan
        System.out.println("Status pemesanan Laundry KTV");
        // Pemesanan dilakukan oleh pelanggan
        System.out.println("-Pemesanan dibuat oleh pelanggan");
    }
}

```

```

// Konfirmasi pemesanan dari pihak laundry
System.out.println("-Konfirmasi pemesanan dari pihak laundry");

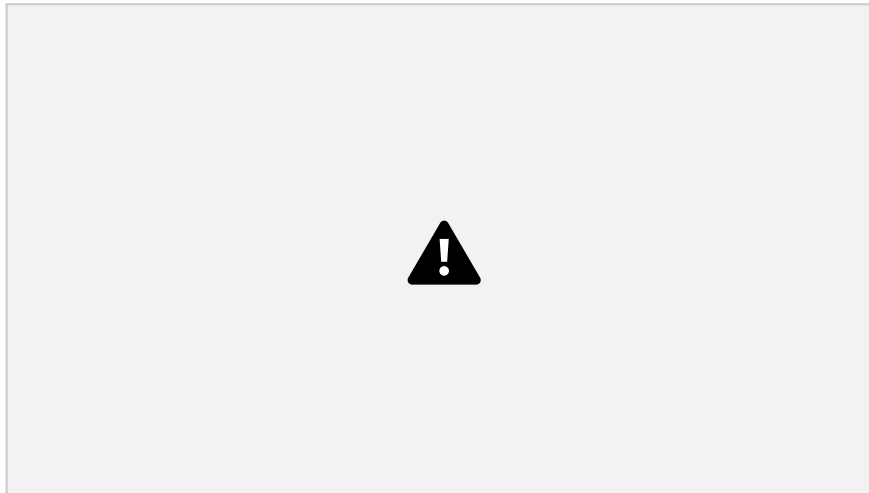
// Driver menerima pesanan
System.out.println("-Driver menerima pesanan");
// Driver menuju lokasi pengambilan laundry
System.out.println("-Driver menuju lokasimu untuk mengambil laundry");

// Laundry sedang diproses di laundry
System.out.println("-Laundry sedang diproses");
// Laundry telah selesai diproses
System.out.println("-Laundry telah siap untuk diantar");
// Driver menuju lokasi pengantaran
System.out.println("-Driver menuju lokasi pengantaran");
// Laundry telah diterima oleh pelanggan
System.out.println("-Laundry telah diterima oleh pelanggan");

// Pembayaran lunas
System.out.println("-Pembayaran lunas");
}
}

```

Output :



17

File : statuspemesananpovdriverKTV.java

**NOTE:** Extension diisi dengan .java

```

public class statuspemesananpovdriverKTV {
    public static void main(String[] args) {
        // Status pemesanan
        System.out.println("Status pemesanan driver");
        // Driver menerima pesanan
        System.out.println("-Pesanan diterima");

        // Driver menuju ke lokasi pelanggan
        System.out.println("-Menuju ke lokasi pelanggan");
        // Driver menuju ke lokasi laundry
        System.out.println("-Menuju ke lokasi laundry");
        // Konfirmasi pihak laundry untuk diproses
        System.out.println("-Pesanan di proses pihak laundry");
    }
}

```

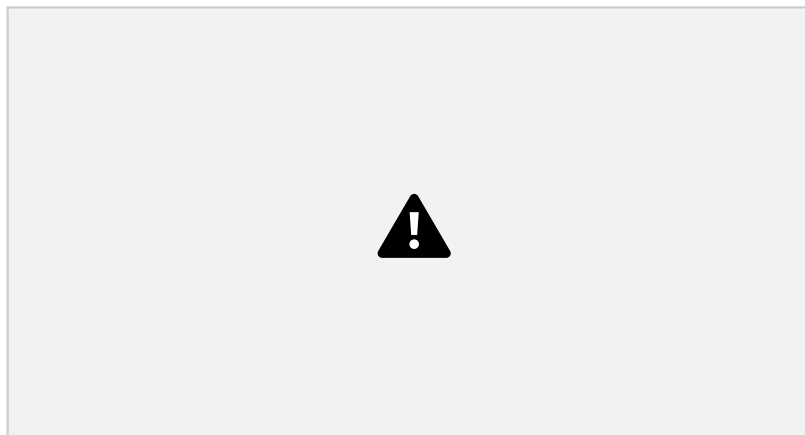
```

// Laundry sedang diproses di laundry
System.out.println("-Laundry sedang diproses");
// Laundry telah selesai diproses
System.out.println("-Laundry telah siap untuk diantar");
// Driver menuju lokasi pengantaran
System.out.println("-Driver menuju lokasi pengantaran");
// Laundry telah diterima oleh pelanggan
System.out.println("-Laundry telah diterima oleh
pelanggan");

// Driver mengupload foto
System.out.println("Silahkan untuk mengupload bukti!");
// Pembayaran lunas
System.out.println("-Pembayaran lunas");
}
}

```

Output :



18

## Latihan 9.2. Membuat *Test Case Scenario*

**Project Name :** KTV's LAUNDRY

**Sub Module Name :** Registrasi

**Type of Component :** ( ☒ ) Screen

( ☒ ) Model

( ☒ ) Controller

( ) Form Printout / Report

( ☒ ) Struktur DB



( ) Lainnya : \_\_\_\_\_

**List of Component Inspected :**

Test Case ID	Description of Test Case	Expected Result	A
--------------	--------------------------	-----------------	---

registktv	<p>Input : Nama Lengkap, Alamat Email, No Telepon, Nama Pengguna, Password serta Konfirmasi Password.</p> <p>Action :</p> <ol style="list-style-type: none"> <li>1. Buka aplikasi</li> <li>2. Muncul halaman utama untuk registrasi</li> <li>3. Menyimpan data pengguna ke sistem</li> </ol>	<p>Hasil yang diharapkan yaitu user berhasil mendaftarkan diri dengan penginputan data diri.</p>	<p>User mengunggah data</p>
-----------	--	--	-----------------------------

19

<p><b>Screenshot :</b></p> <p><b>Tampilan jika registrasi akun berhasil :</b></p> <div style="text-align: center;">  </div> <p><b>Tampilan jika registrasi akun gagal :</b></p> <div style="text-align: center;">  </div>			
<p><b>Sub Module Name : Login</b></p>			

**Type of Component :** ( ☒ ) Screen

( ☒ ) Model

( ) Controller



( ) Form Printout / Report

20

( ☒ ) Struktur DB

( ) Lainnya : \_\_\_\_\_

**List of Component Inspected :**

Test Case ID	Description of Test Case	Expected Result	A
LoginLaundryK TV	<p>Input : Nama pengguna (username) dan password yang dibuat saat registrasi akun.</p> <p>Action :</p> <ol style="list-style-type: none"> <li>1. Tampil halaman login</li> <li>2. Menyimpan data login</li> <li>3. Tampil login berhasil</li> </ol>	User dapat melakukan login dengan menginput username dan password yang sudah dibuat saat registrasi akun.	User melakukan login.
<p><b>Screenshot :</b></p> <p>Tampilan jika login berhasil :</p>  <p>Tampilan jika login gagal :</p> 			

21

**Sub Module Name :** Pemesanan (order)

**Type of Component :** ( √ ) Screen

( √ ) Model

( √ ) Controller

( √ ) Form Printout / Report

( √ ) Struktur DB

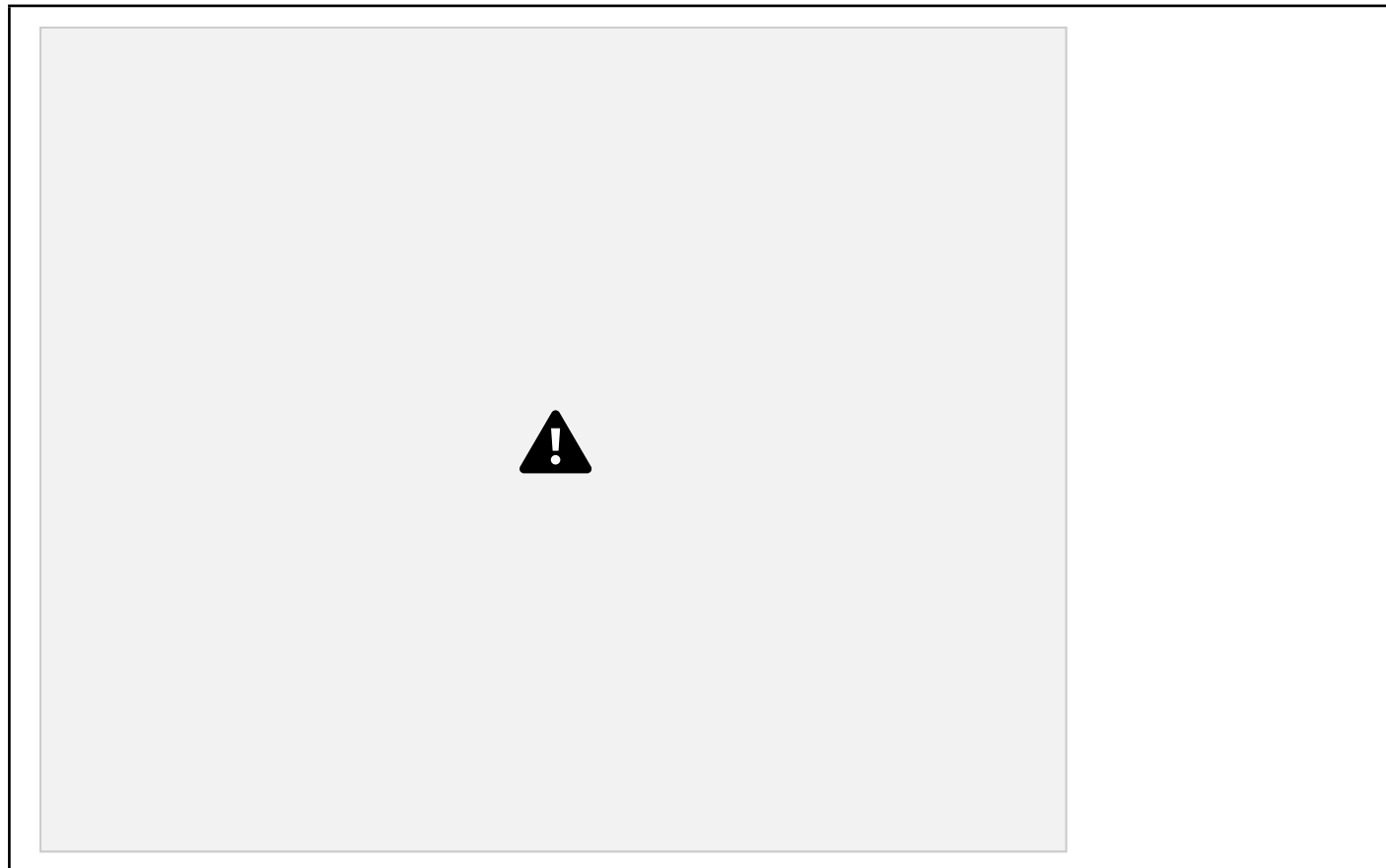
( ) Lainnya : \_\_\_\_\_

**List of Component Inspected :**

Test Case ID	Description of Test Case	Expected Result	A
--------------	--------------------------	-----------------	---

<b>LaundryOrder</b>	<p>Input : Jenis Laundry (option), Metode laundry+ry (option), Lokasi antarjemput, Metode pembayaran (option).</p> <p>Action :</p> <ol style="list-style-type: none"> <li>1. Tampil halaman utama (beranda)</li> <li>2. Menampilkan menu pemesanan</li> <li>3. Menampilkan opsi pembayaran</li> <li>4. Menampilkan detail pemesanan</li> </ol>	User dapat melakukan pemesanan, mengisi lokasi, memilih pembayaran, dan menerima detail pemesanan.	User melakukan pemesanan akhir.
<b>Screenshot :</b>			

22



23

<b>Sub Module Name :</b> Antar Jemput Pesanan
---

**Type of Component :** ( ✓ ) Screen  
( ✓ ) Model

( √ ) Controller

( √ ) Form Printout / Report

( √ ) Struktur DB

( ) Lainnya : \_\_\_\_\_

**List of Component Inspected :**

Test Case ID	Description of Test Case	Expected Result	A
antarjemputl au ndryKTV	Input : (driver) menerima atau menolak pesanan (user)  Action :  1. Tampil detail pesanan (termasuk lokasi penjemputan dan pembayaran)	Driver dapat menerima dan menolak pesanan user, dan user mendapatkan notifikasi bahwa pesanan diterima atau ditolak. Jika ditolak oleh driver, sistem akan mencari driver yang baru.	Drive mene ak pe user menc notifi pesar diteri Sistem menc yang pesar oleh user a

24

	2. Tampil di perangkat user bahwa pesanan diterima ataupun ditolak		menc notifi
--	---	--	----------------

**Screenshot :**

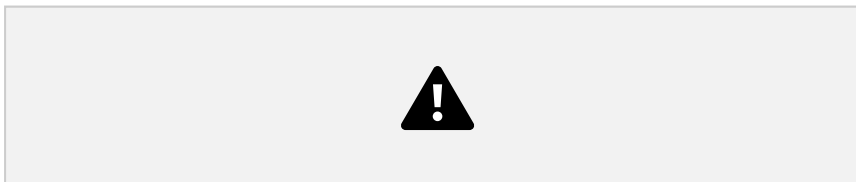
Jika pesanan diterima oleh driver :



Jika pesanan ditolak oleh driver :



Jika inputan driver tidak valid :



25

**Sub Module Name :** Status Pemesanan Laundry KTV

**Type of Component :** ( ☒ ) Screen

( ☐ ) Model

( ☐ ) Controller

( ☐ ) Form Printout / Report

( ☒ ) Struktur DB

( ☐ ) Lainnya : \_\_\_\_\_

**List of Component Inspected :**

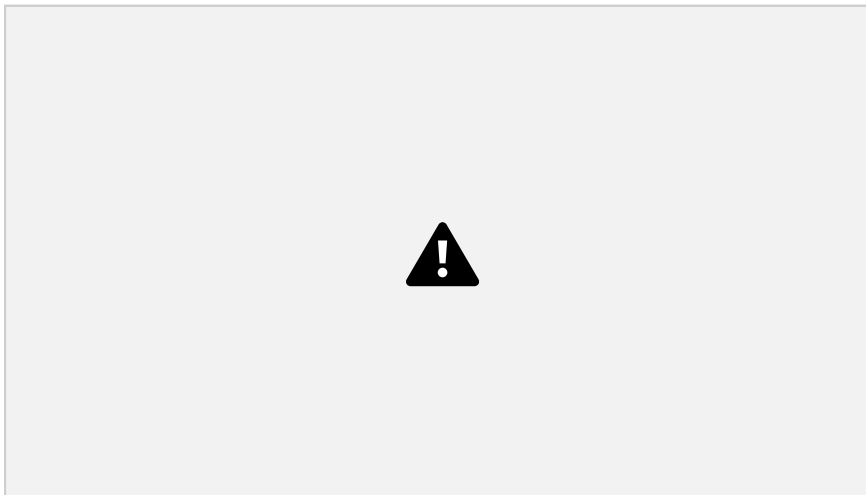
Test Case ID	Description of Test Case	Expected Result	A
--------------	--------------------------	-----------------	---



statuspemesan an anKTV	Input : -  Action :  1. Sistem menyimpan proses – proses pemesanan 2. Tampil proses pemesanan di perangkat user	User dapat melihat proses pemesanan dari awal sampai akhir.	User proses dari akhir.
------------------------	--	---	-------------------------

26

**Screenshot :**



**Sub Module Name :** Status Pemesanan pov.Driver

**Type of Component :** ( ☒ ) Screen

( ☒ ) Model

( ☒ ) Controller

( ) Form Printout / Report

( ☒ ) Struktur DB

( ) Lainnya : \_\_\_\_\_

27

**List of Component Inspected :**

Test Case ID	Description of Test Case	Expected Result	A
statuspemes an anpovdriver KT V	Input : Upload foto bukti pengantaran  Action :  1. Sistem menyimpan proses – proses pemesanan dan foto bukti pengantaran.  2. Tampil proses pemesanan di perangkat user.	Driver dapat melihat proses pemesanan dari awal sampai akhir, serta dapat mengupload foto bukti pengantaran.	Drive melih peme samp serta foto peng
Screenshot :			

28



29