

LAPORAN PRAKTIKUM

METODOLOGI PENGEMBANGAN PERANGKAT LUNAK

MODUL 9



Versi 3.1

Penyusun : Tim Dosen MPPL

Nama : 1. Aisyah Nur Fadhlia

2. Jovita Amanda Putri

3. Putri Syabillah

NIM : 1. 064002200020

2. 064002200033

3. 064002200015

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.

N o	Kemampuan Akhir Yang Diharapkan	Kriteria Penilaian dan Indikator	Jml Menit	Bobot Nilai (%)
----------------	--	---	----------------------	--------------------------------

1	– Mahasiswa mampu mengoperasikan IDE dengan benar (Sub CPMK 2.4).	Kriteria penilaian : Rubrik deskriptif. Indikator : Ketepatan waktu dalam menyelesaikan Laporan Praktikum 9 .	85	1,5
TOTAL			85	1,5

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

2

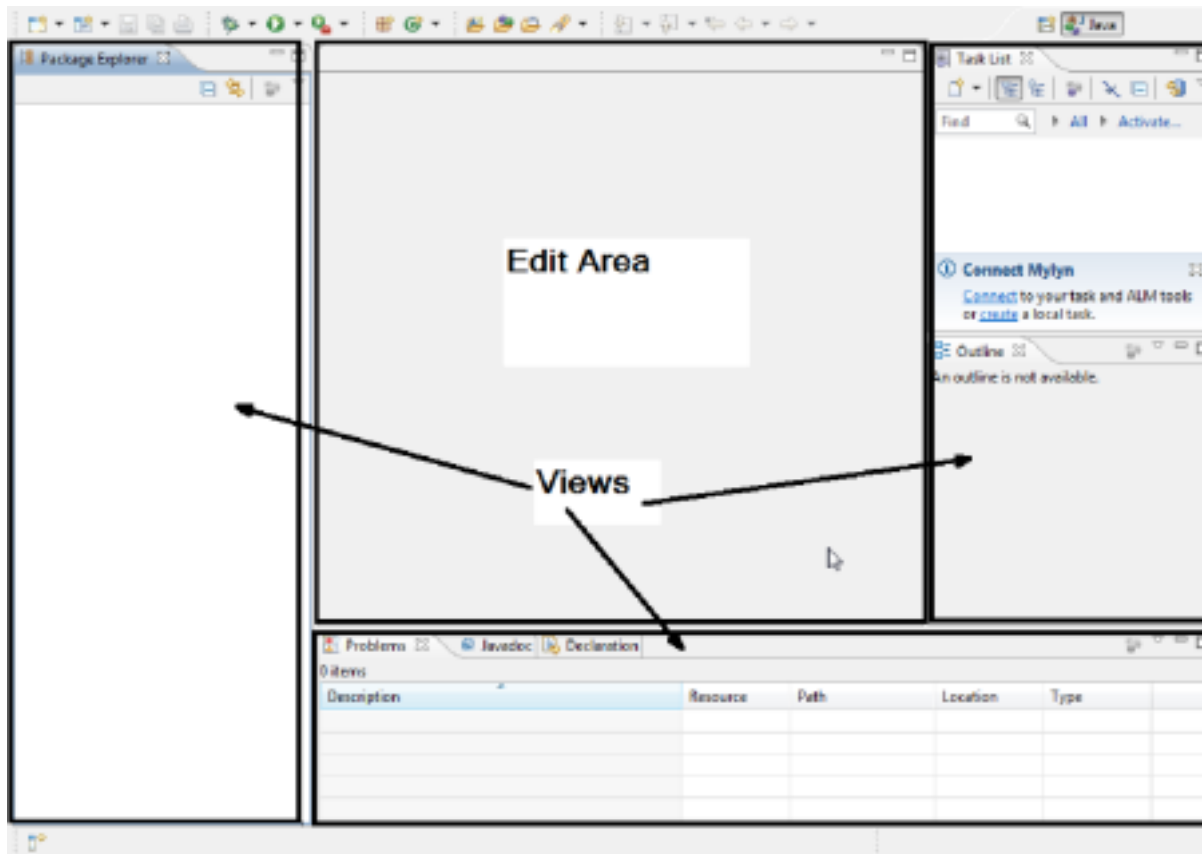
Eclipse merupakan sebuah *development environment* yang sangat *user-friendly* untuk membuat program JAVA. Eclipse sudah banyak dikenal dengan baik oleh banyak programmer JAVA. dan digunakan untuk berbagai pengembangan aplikasi. Sehingga mempelajari Eclipse dapat menambah wawasan dan pengetahuan tentang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

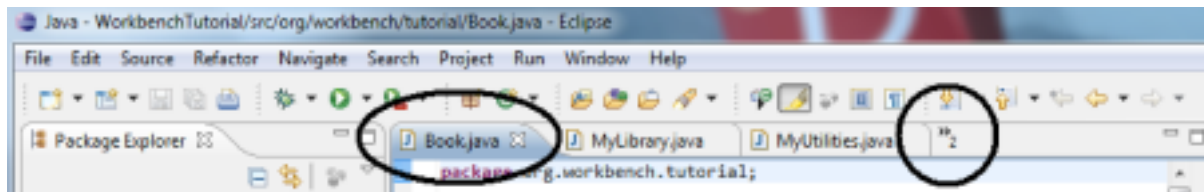
1. Pada komputer berbasis Windows, *double-click* pada *file* eclipse.exe. Sedangkan pada komputer berbasis Linux atau Mac, *double-click* pada *file* eclipse.
2. Ketika muncul *dialog box*, masukkan *pathname* untuk *workspace* (digunakan untuk menyimpan *JAVA Project*), dan tekan tombol **OK**.
3.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

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



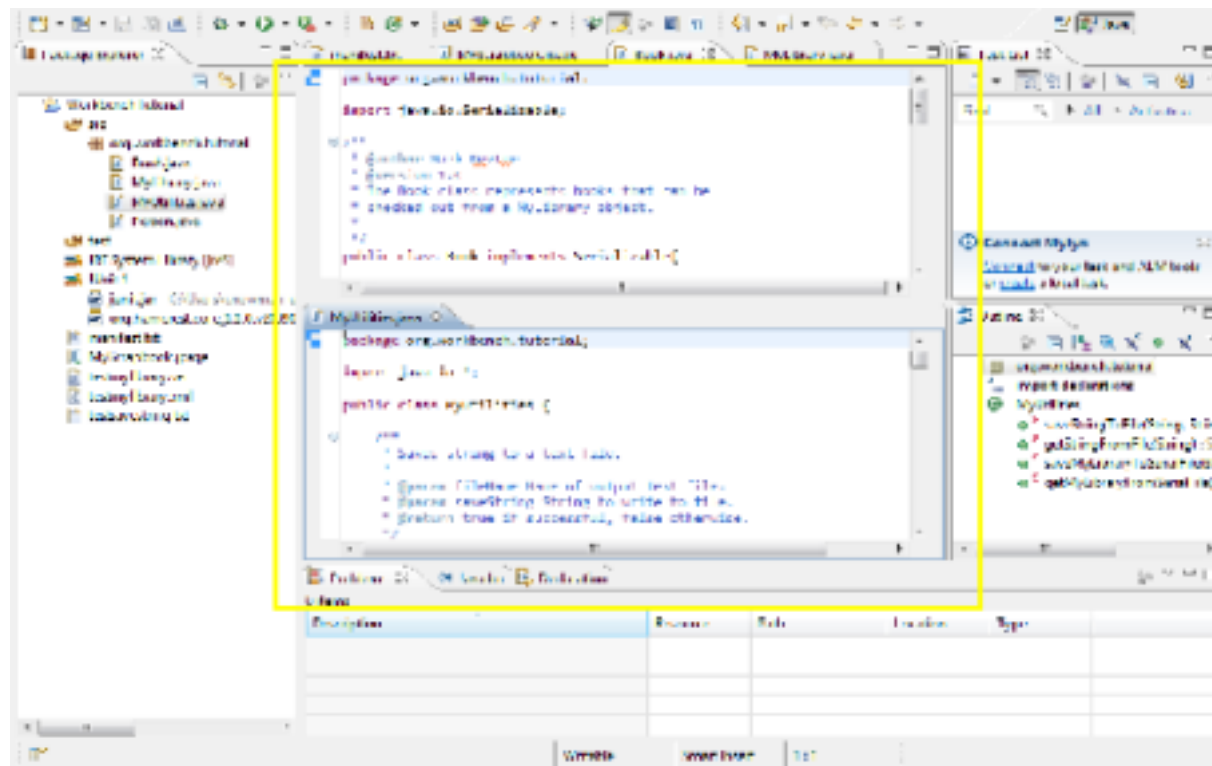
Gambar Edit Area dan View

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.



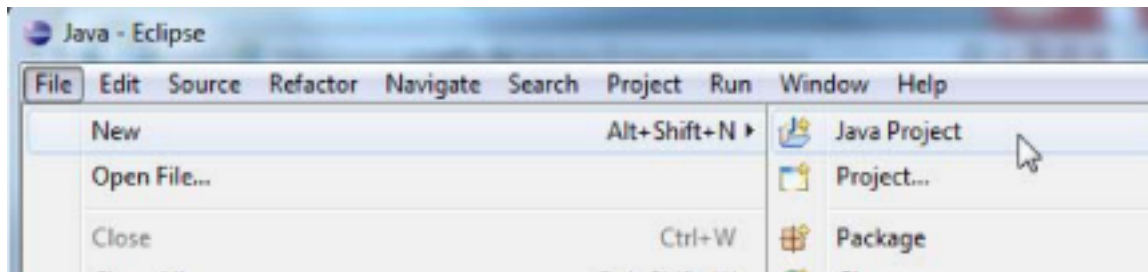
Gambar *Multiple Windows* Pada Edit Area

5

Langkah untuk membuat program JAVA :

1. Buat **Project**

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



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

New Java Project

Create a Java Project
Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location
Location:

JRE

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE (currently 'jre6') [Configure JREs...](#)

Project layout

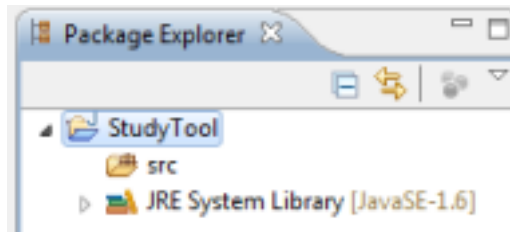
☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

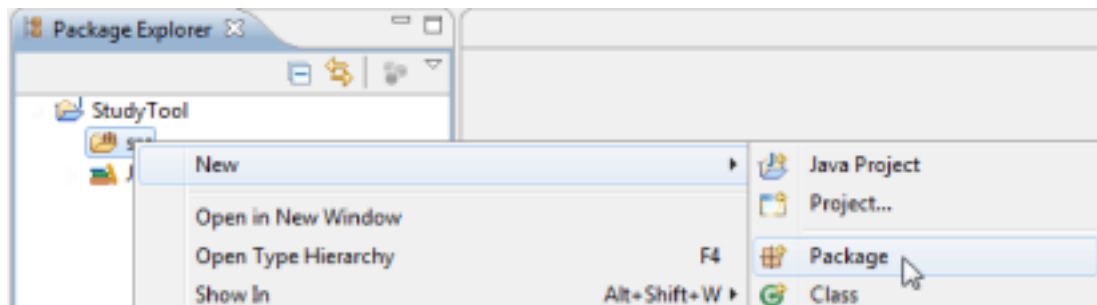
☐ Add project to working sets
Working sets:

- 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**

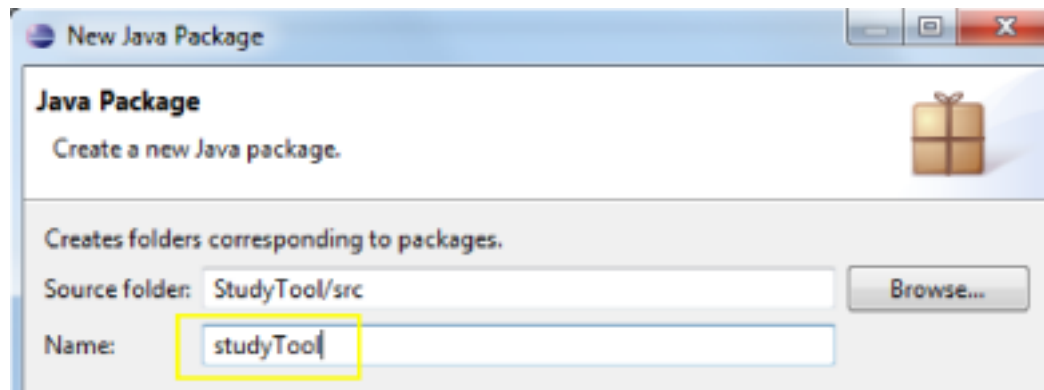


2. Buat **Package**

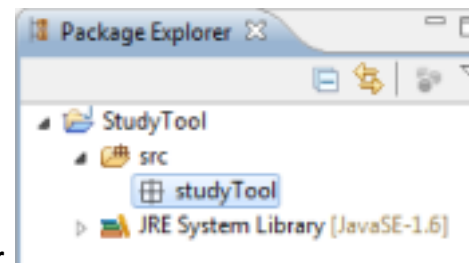
- Pada tab **Package Explorer**, pilih <Nama Project>, lalu klik kanan dan pilih **New** → **Package**



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



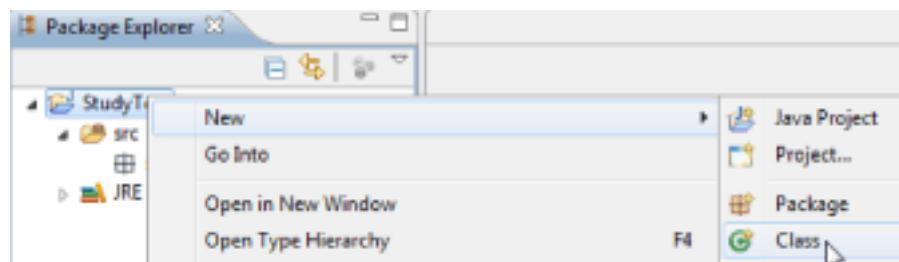
7



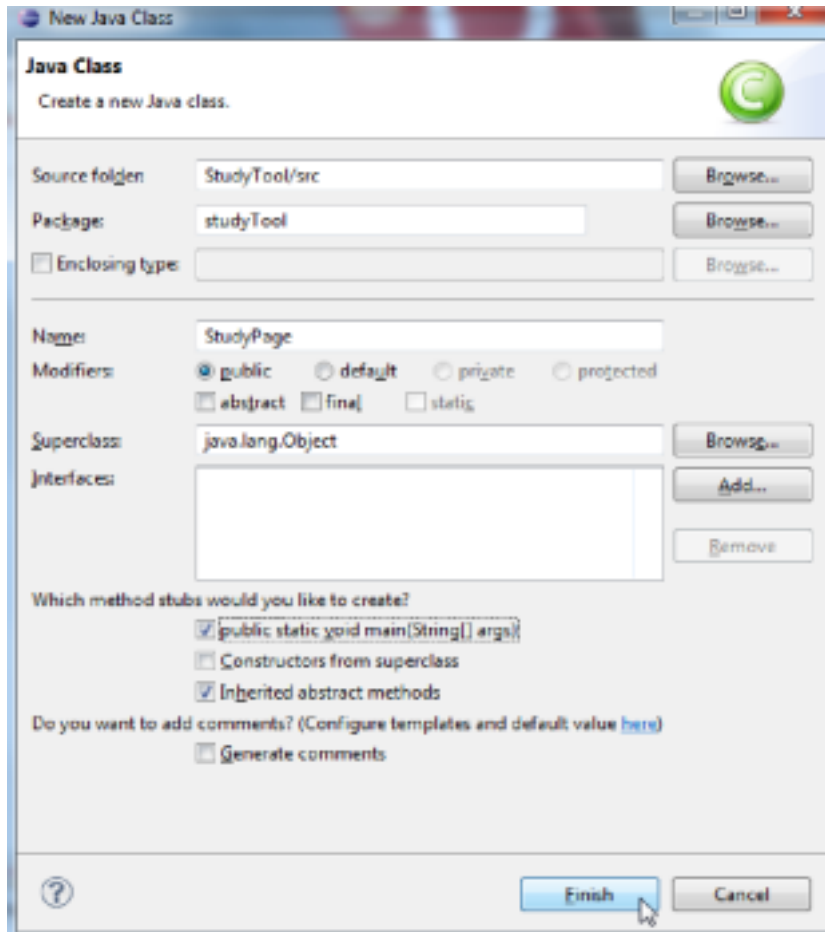
- Sehingga *package* tersebut akan tampil pada **Package Explorer**

3. Pilih **Class**

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



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

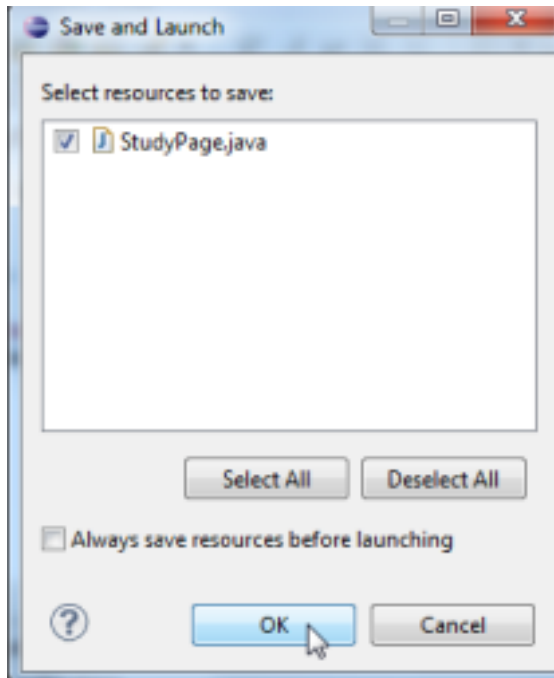


4. Create and run JAVA code

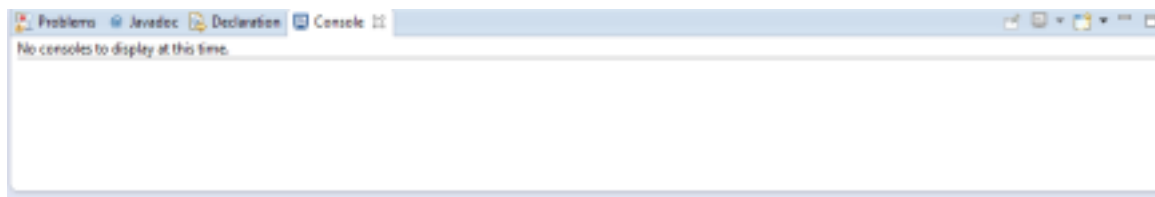
- Ketikkan *source code* pada **Main Class**

```
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**



TUGAS

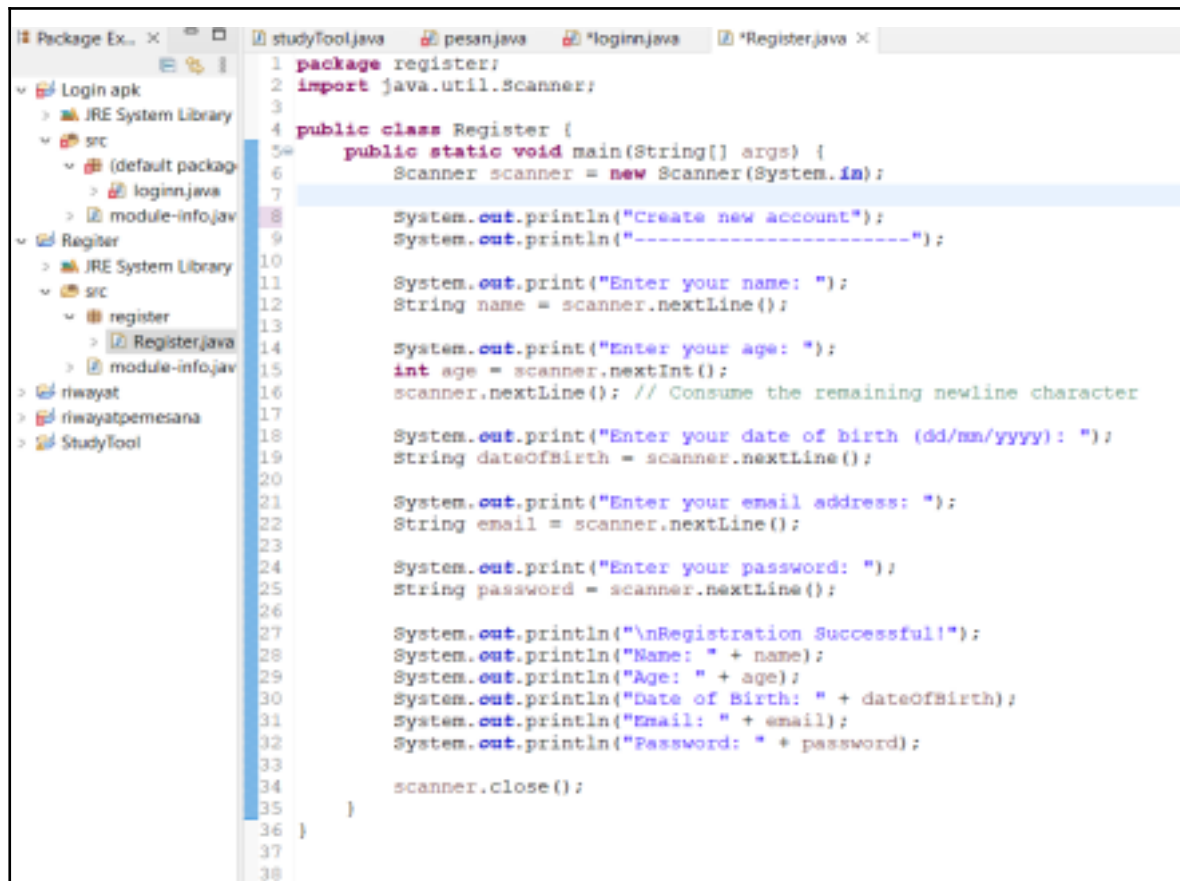
Pelajari studi kasus. Kemudian lengkapi isian berikut ini.

10

Latihan 9.1. Membuat *Class*

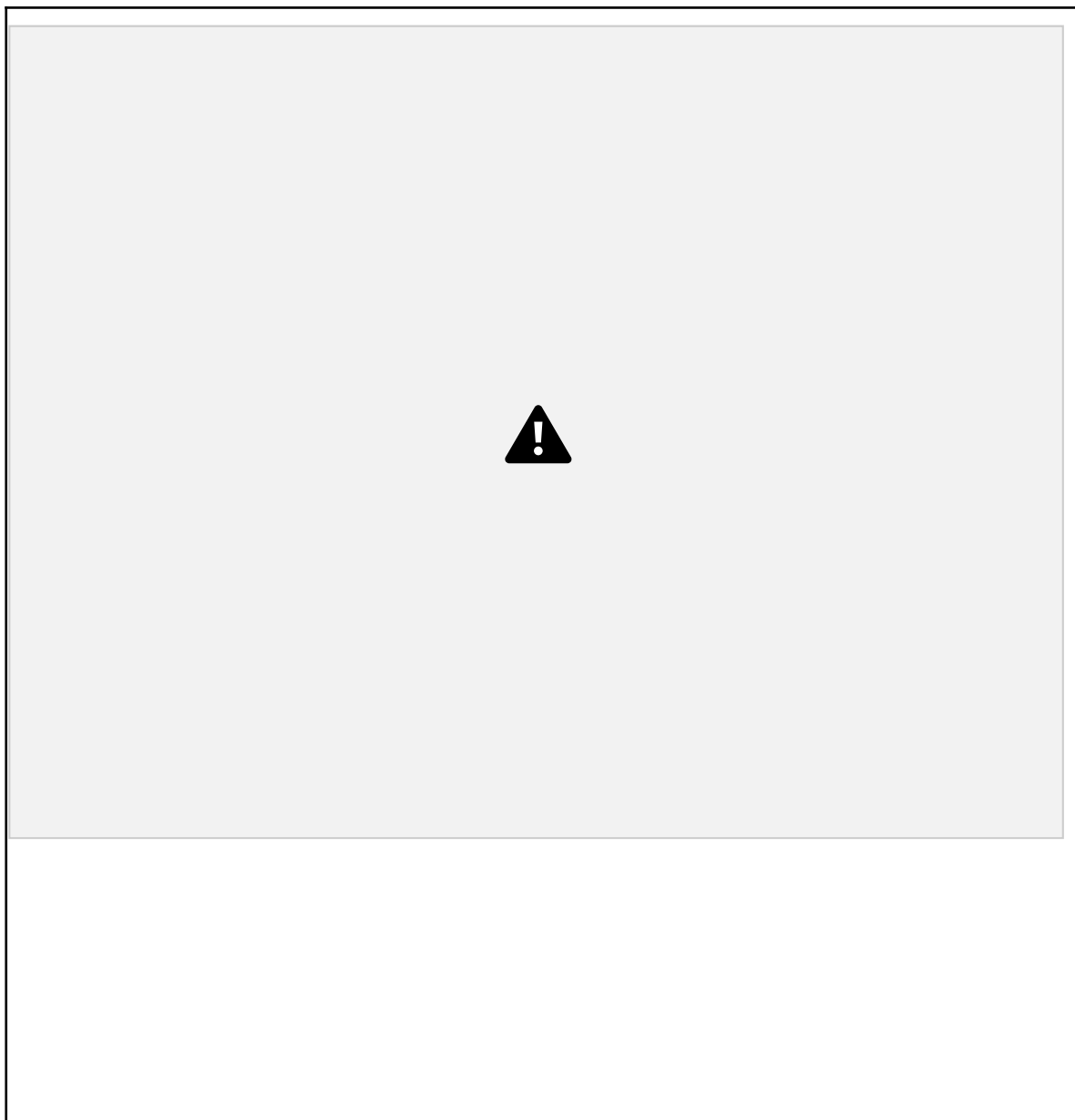
File : <Register>.<java>

NOTE : Extension diisi dengan .java



File : <StudyPage>.<java> (**fitur Log in**)

NOTE : Extension diisi dengan .java

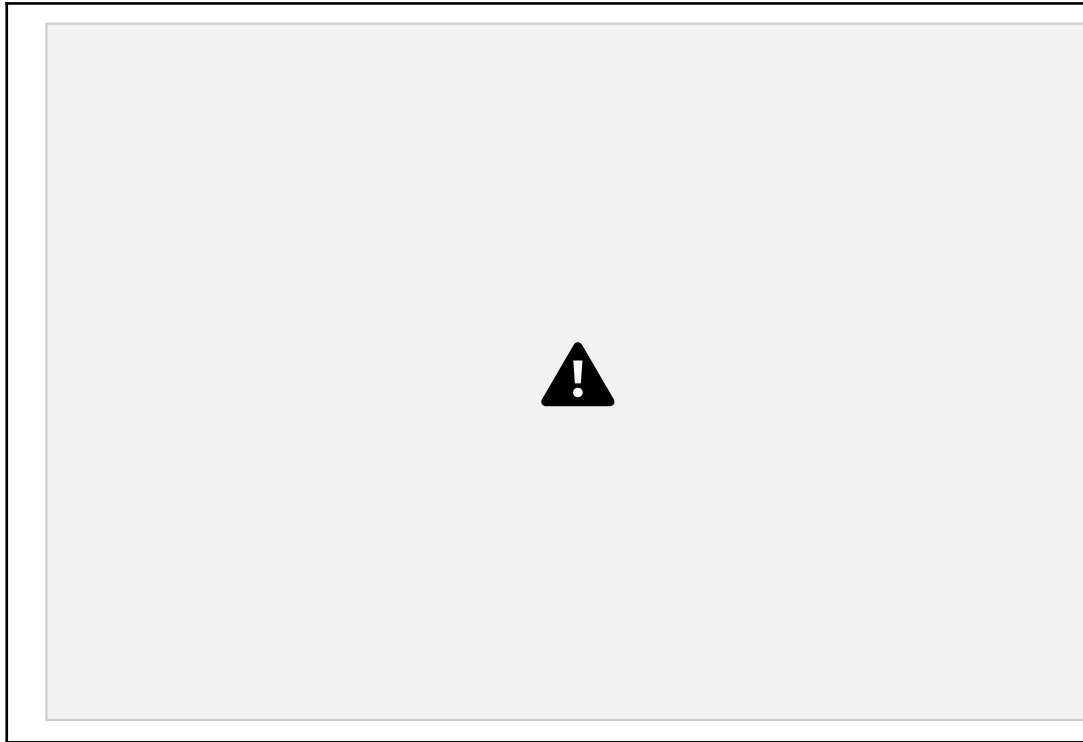


12

File : <Pemesanan>.<java> (**fitur riwayat pemesana**)

NOTE : Extension diisi dengan .java



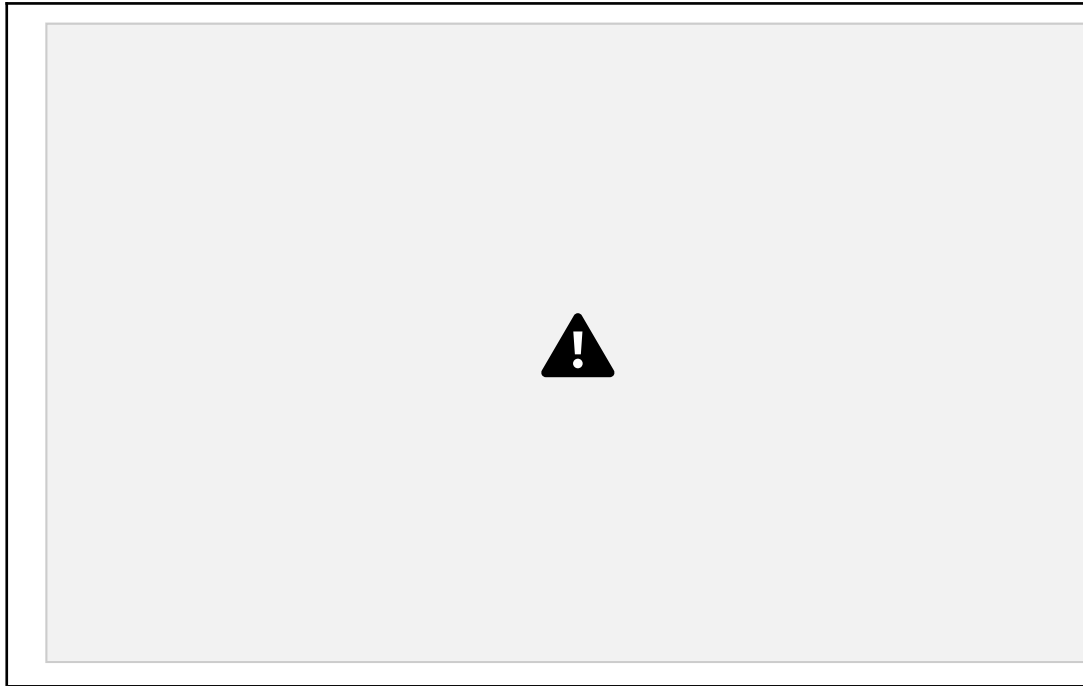


13

File : <Beranda>.<java>

NOTE : Extension diisi dengan .java



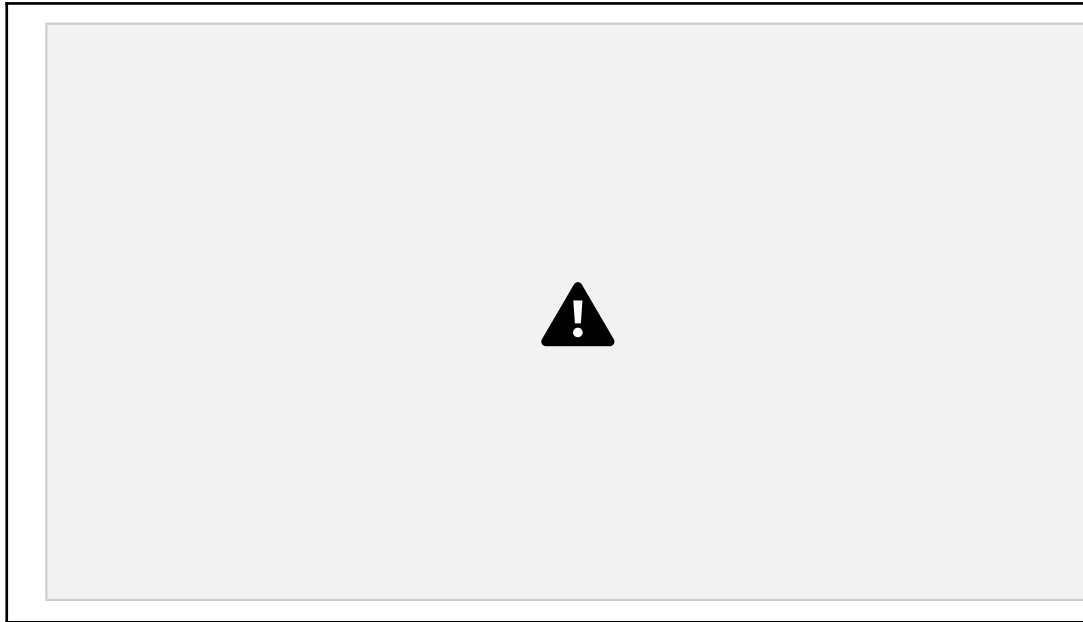


14

File : <Pengaturan>.<java>

NOTE : Extension diisi dengan .java





15

File : <Pemesanan1>.<java> (**fitur pemesanan**)

NOTE : Extension diisi dengan .java



Latihan 9.2. Membuat *Test Case Scenario***Project Name :** ShooutIT!!**Sub Module Name :** <Register>**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	Actual Result	Pass or Fail
Register.java	Input : Name, Age, tanggal/bulan/tahun lahir, email, dan password Action : 1. Buka aplikasi.	pengguna berhasil mendaftarkan diri	pengguna berhasil mendaftarkan diri dengan menginput data- data yang diminta.	Pass

	<ol style="list-style-type: none"> 2. Beralih ke halaman atau tampilan registrasi. 3. Masukkan informasi pengguna yang valid pada kolom formulir registrasi, yang berisikan Name, Age, tanggal/bulan/tahun lahir, email, dan password 4. Klik tombol "Registrasi" 5. Verifikasi bahwa proses registrasi berhasil dan pengguna diarahkan ke halaman atau tampilan yang sesuai (misalnya, halaman utama, halaman login). 6. Uji keamanan dari proses registrasi dengan mencoba untuk menghindari langkah-langkah validasi atau otentikasi. 7. Verifikasi bahwa aplikasi mencegah akses atau manipulasi yang tidak sah terhadap proses registrasi. 8. Verifikasi bahwa aplikasi dapat menangani beban tersebut dan mendaftarkan banyak pengguna tanpa ada masalah kinerja. 9. Uji fitur atau fungsionalitas tambahan yang spesifik untuk proses registrasi, seperti verifikasi email atau captcha. 10. Dokumentasikan masalah atau bug yang ditemui selama proses 			
--	---	--	--	--

	pengujian dan			
--	---------------	--	--	--

	laporkan kepada tim pengembangan untuk diperbaiki			
--	--	--	--	--

Screenshot



Sub Module Name : <login >

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	Actual Result	Pass or Fail
StudyPage.java	Input : Email,password Action : 1. Buka aplikasi. 2. Beralih ke halaman atau tampilan login. 3. Masukkan email atau username yang valid pada kolom yang sesuai. 4. Masukkan password yang valid pada kolom password. 5. Klik tombol "Login" atau submit formulir.	pengguna berhasil login kedalam aplikasi	pengguna berhasil login kedalam aplikasi menggunakan email dan password yang sudah didaftarkan	Pass

	<p>6. Verifikasi bahwa proses login berhasil dan pengguna diarahkan ke halaman atau tampilan yang sesuai.</p> <p>7. Coba melakukan login dengan email/username atau password yang tidak valid.</p> <p>8. Verifikasi bahwa pesan kesalahan yang tepat ditampilkan, menunjukkan bahwa login gagal karena kredensial yang salah.</p> <p>9. Ulangi proses login dengan informasi yang tidak valid atau tidak lengkap, seperti tidak mengisi kolom email/username atau password.</p>			
--	---	--	--	--

Screenshot :



Sub Module Name : Riwayat Pemesanan

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

(✓) Form Printout / Report

() Struktur DB

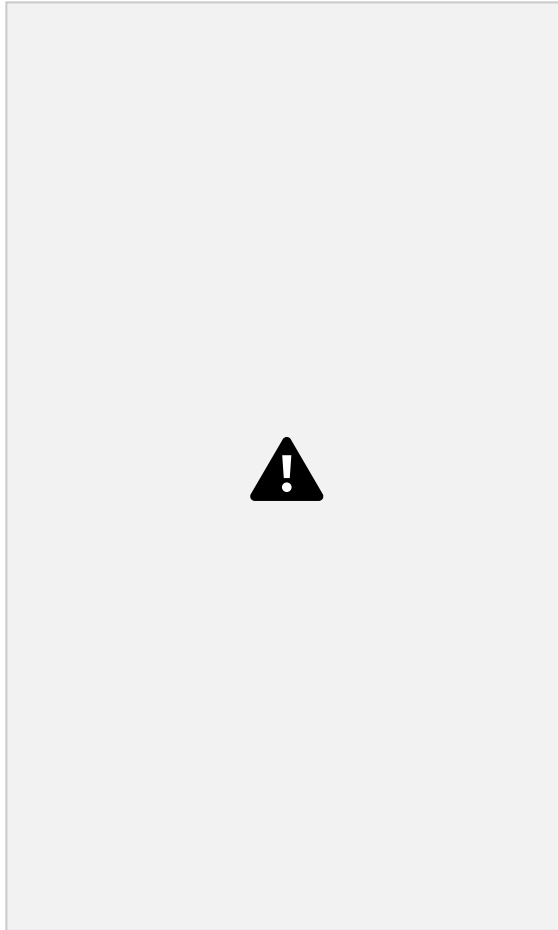
() Lainnya : _____

List of Component Inspected :

Test Case ID	Description of Test Case	Expected Result	Actual Result	Pass or Fail
Pemesanan.j av a	Input : - Action : 1. masuk ke beranda 2. pilih fitur riwayat pemesanan 3. pada riwayat pemesanan terdapat informasi riwayat pemesanan yang telah ditambahkan ke dalam daftar daftarPemesanan. Setiap pemesanan ditampilkan dengan informasi nama, umur, jenis pemesanan, nama dokter, dan harga. data tersebut berdasarkan konsultasi yang telah dilakukan user.	pengguna dapat mengakses dan melihat daftar riwayat pemesanan	pengguna dapat mengakses dan melihat daftar riwayat pemesanan	Pass

--	--	--	--	--

Screenshot :



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	Actual Result	Pass or Fail
Beranda.java	<p>Input : pada aplikasi tidak ada perintah input di beranda, tetapi dalam pengimplementasian source code diminta menginput nomor sesuai fiturnya.</p> <p>Action :</p> <p>1. Buka aplikasi.</p>	pengguna dapat mengakses beranda	pengguna berhasil mengakses beranda aplikasi	Pass

2. Verifikasi bahwa pengguna diarahkan ke halaman beranda setelah login atau saat membuka aplikasi.
3. Periksa apakah fitur pemesanan tersedia

- dan terlihat di halaman beranda.
4. Klik atau pilih fitur pemesanan untuk melihat detail atau memulai proses pemesanan.
 5. Verifikasi bahwa pengguna diarahkan ke halaman atau tampilan yang sesuai untuk melanjutkan proses pemesanan.

Screenshot :



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	Actual Result	Pass or Fail
Pengaturan.j av a	<p>Input : pada aplikasi tidak ada perintah input di beranda, tetapi dalam pengimplementasian source code diminta menginput nomor sesuai fitur yang ada di dalam pengaturan.</p> <p>Action :</p> <p>1. Buka aplikasi.</p>	pengguna dapat mengakses menu setting	pengguna berhasil mengakses menu setting	Pass

	<p>2.Verifikasi bahwa pengguna telah login dan berada di halaman beranda sebelum mengakses fitur pengaturan.</p> <p>3.Periksa apakah fitur pengaturan tersedia dan terlihat di halaman beranda.</p> <p>4. Klik atau pilih fitur pengaturan untuk mengakses pengaturan aplikasi.</p> <p>5.Verifikasi bahwa pengguna diarahkan ke halaman atau tampilan yang sesuai untuk mengatur preferensi atau konfigurasi aplikasi.</p> <p>6.Periksa apakah fitur profil tersedia dan terlihat di halaman pengaturan.</p> <p>7. Klik atau pilih fitur profil untuk melihat atau mengedit informasi pengguna.</p> <p>8.Verifikasi bahwa pengguna diarahkan ke halaman atau tampilan yang sesuai untuk melihat atau mengedit profil.</p> <p>9.Kembali ke halaman pengaturan.</p> <p>10.Periksa apakah fitur catatan medis tersedia dan terlihat di halaman pengaturan.</p> <p>11. Klik atau pilih fitur catatan medis untuk melihat atau mengedit catatan medis pengguna.</p>			
--	--	--	--	--

	<p>12. Verifikasi bahwa pengguna diarahkan ke halaman atau tampilan yang sesuai untuk melihat atau mengedit catatan medis.</p> <p>13. Kembali ke halaman pengaturan.</p> <p>14. Periksa apakah fitur keamanan tersedia dan terlihat di halaman pengaturan.</p> <p>15. Klik atau pilih fitur keamanan untuk mengatur pengaturan keamanan akun pengguna.</p> <p>16. Verifikasi bahwa pengguna diarahkan ke halaman atau tampilan yang sesuai untuk mengatur pengaturan keamanan.</p> <p>17. Kembali ke halaman pengaturan.</p> <p>18. Periksa apakah fitur umum (general) tersedia dan terlihat di halaman pengaturan.</p> <p>19. Klik atau pilih fitur umum untuk mengatur pengaturan umum aplikasi.</p> <p>20. Verifikasi bahwa pengguna diarahkan ke halaman atau tampilan yang sesuai untuk mengatur pengaturan umum.</p> <p>21. Lakukan pengujian pada setiap fitur pengaturan untuk memastikan bahwa perubahan yang dilakukan oleh</p>			
--	--	--	--	--

	pengguna diterapkan dan berfungsi dengan baik			
--	---	--	--	--

30

Screenshot :				

31



Sub Module Name : Pemesanan

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	Actual Result	Pass or Fail
--------------	--------------------------	-----------------	---------------	--------------

Pemesanan1.ja va

Input : -

Action :

1. masuk ke menu

utama(beranda) 2. masuk ke fitur pemesanan

pengguna dapat

mengakses menu pemesanan

pengguna berhasil

mengakses menu pemesanan

Pass

	<ul style="list-style-type: none">3. pilih metode/ jenis pemesanan secara online atau offline4. melakukan proses pemesanan5. terakhir melakukan pembayaran dari pemesanan			
--	---	--	--	--

Screenshot :

