

**DOKUMEN UJI PERANGKAT LUNAK**

**<**KOST KITA**>**

untuk:

<Orang Perantauan>

Dipersiapkan oleh:

<Andika Rizki Putra Pamungkas (2211102181)>

<Zahra Salsabila (2211102206)>

<Rezki Nur Vidhiyansyah(2211102207)>

<Alvi Anggraeni(2211102209)>

<El Vares Hadni Hameed(2211102220)>

Prodi Teknik Informatika - Universitas Telkom Purwokerto

|  | Prodi Teknik Informatika  Universitas Telkom  Purwokerto | **Nomor Dokumen** | | **Halaman** |
| --- | --- | --- | --- | --- |
| ***DUPL-****001* ***<x: id\_proyek>*** | | *<#>/<jml #* |
| **Revisi** | *<nomor revisi>* | *Tgl: <isi tanggal>* |

**Daftar Isi**

[1 Pendahuluan 4](#_heading=h.30j0zll)

[1.1 Tujuan Pembuatan Dokumen 4](#_heading=)

[1.2 Ruang Lingkup Pengujian 4](#_heading=h.3znysh7)

[1.3 Referensi 4](#_heading=)

[1.4 Overview Sistem & Fitur Utamanya 4](#_heading=)

[1.5 Overview Pengujian 5](#_heading=h.2et92p0)

[1.5.1 Perangkat Keras Pengujian 5](#_heading=)

[1.5.2 Sumber Daya Manusia 5](#_heading=)

[1.5.3 Perangkat Lunak Pengujian 5](#_heading=h.3dy6vkm)

[1.5.4 Material Pengujian 6](#_heading=)

[1.5.5 Strategi dan Metode Pengujian 6](#_heading=)

[1.5.6 Jadwal Pengujian 6](#_heading=)

[**2 Pelaksanaan Pengujian 6**](#_heading=)

[2.1 Pengujian UNIT 7](#_heading=)

[2.1.1 Pengujian White Box Method 7](#_heading=)

[2.1.2 Pengujian Class dengan JUnit/PhPUnit 15](#_heading=)

[2.2 Pengujian INTEGRASI 16](#_heading=)

[2.2.1 Pengujian DUPL-01 Login User 16](#_heading=h.b583yjbyo4dy)

[2.2.3 Pengujian DUPL-01 Pencarian Kost 17](#_heading=h.z595m9shm1ut)

[2.2.4 Pengujian DUPL-01 Pemesanan Kost 18](#_heading=)

[2.2.5 Pengujian DUPL-01 Pembayaran Kost 18](#_heading=h.oozq9a8suuar)

[2.3 SYSTEM TEST (Optional, jika software embedded di system IOT dll) 19](#_heading=)

[2.4 USER ACCEPTANCE TEST (Berjuang mencari user potensial) 19](#_heading=)

[2.5 Summary Pengujian Yang Masih Gagal 20](#_heading=)

[**3 Lampiran 20**](#_heading=)

**Daftar Gambar**

Hanya dicantumkan dan diisi jika ada gambar pada badan dokumen

**Daftar Tabel**

Hanya dicantumkan dan diisi jika ada tabel pada badan dokumen

**Daftar Lampiran**

Hanya dicantumkan dan diisi jika ada lampiran setelah badan dokumen

# **Pendahuluan**

## ***Tujuan Pembuatan Dokumen***

Tujuan dari pembuatan dokumen ini adalah sebagai panduan untuk melakukan pengujian terhadap perkembangan proyek perangkat lunak website Kost Kita. Dokumen ini digunakan untuk melihat hasil uji dari perangkat lunak website Kost Kita apakah sudah sesuai dengan rancangan yang telah dibuat atau tidak. Dokumen ini ditujukan untuk programmer sebagai bahan pertimbangan pengembangan perangkat lunak lebih lanjut agar memenuhi target perancangan.

## ***Ruang Lingkup Pengujian***

Lingkungan Pengujian Perangkat Lunak ini meliputi Perangkat Lunak Pengujian, Perangkat Keras Pengujian, Sumber Daya Manusia, Perangkat Lunak Pengujian, Material Pengujian, Strategi dan Metode Pengujian, dan Jadwal Pengujian.

## ***Referensi***

Berikut adalah referensi yang digunakan dalam membuat dokumen DUPL ini:

1. SKPL-001 - SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK, Kost Kita
2. DPPL-001 – DOKUMEN PERANCANGAN PERANGKAT LUNAK, Kost Kita
3. Template DUPL, S1 Teknik Informatika Universitas Telkom

## ***Overview Sistem & Fitur Utamanya***

Website Kost Kita merupakan sebuah platform digital yang dirancang untuk memfasilitasi pencarian dan penyewaan kost secara praktis. Sistem menghubungkan penyewa dan pemilik kost dalam satu ekosistem yang terintegrasi, memudahkan pengguna untuk menemukan tempat tinggal yang sesuai kebutuhan dengan cepat. Menggunakan teknologi berbasis web responsif, Kost Kita mendukung akses dari berbagai perangkat, termasuk desktop, tablet, dan smartphone.

Fitur utama website Kost Kita meliputi:

1. Pengguna dapat melakukan pencarian berdasarkan lokasi sekitar Purwokerto, harga, fasilitas, dan tipe kamar.
2. Dilengkapi dengan galeri foto setiap tipe kamarnya, dan deskripsi lengkap
3. Penyewa dapat memesan atau menghubungi pemilik langsung melalui sistem pesan terintegrasi.
4. Admin dapat mengelola properti dari setiap pemilik kost, mengunggah informasi baru, dan memantau penyewaan.
5. Data pengguna dilindungi dengan sistem autentikasi dan enkripsi.

## ***Overview Pengujian***

### **Perangkat Keras Pengujian**

Perangkat keras yang digunakan untuk menguji website *Kost Kita* mencakup berbagai perangkat dengan spesifikasi sebagai berikut:

1. Server WAP

* Intel Xeon atau AMD EPYC dengan frekuensi minimum 2.4 GHz.
* Harddisk 500 GB SSD/Ram 16 GB.
* Windows, macOS, atau Linux.

1. User

* Intel Core i3 generasi ke-8 atau setara.
* Harddisk 2 GB/Ram 4 GB.
* Google Chrome, Microsoft Edge, Safari (perangkat macOS).

### **Sumber Daya Manusia**

Sumber daya manusia yang diperlukan dalam pengujian perangkat lunak ini adalah pengembang dari perangkat lunak website Kost Kita dimana pengembang tersebut memahami :

* Konsep sistem keseluruhan dari perangkat lunak website Kost Kita
* Konsep pemrograman Bahasa PHP
* Konsep *database* menggunakan XAMPP

### **Perangkat Lunak Pengujian**

Perangkat lunak yang digunakan untuk menguji website Kost Kita mencakup berbagai perangkat dengan spesifikasi sebagai berikut:

1. Google Chrome, Microsoft Edge, Safari (yang biasa digunakan untuk Web Browser).
2. MySQL (Database digunakan untuk menyimpan, mengelola, dan mengambil data dalam berbagai aplikasi, terutama di web).
3. Min. Windows 7, macOS (Sistem Operasi)
4. PHP (Web Server)

### **Material Pengujian**

Material pengujian mencakup berbagai modul utama yang membentuk sistem website Kost Kita. Modul-modul yang akan diuji meliputi:

* Modul Pencarian Kost, menguji kemampuan pencarian lokasi.
* Modul Pemesanan Online, memastikan proses pemesanan berjalan.
* Modul Pengelolaan Akun, menguji fungsionalitas registrasi, login, dan pengelolaan data pengguna.
* Modul Menambahkan, mengupdate, dan menghapus Kost, memastikan admin dapat mengunggah informasi, dan memperbarui data.
* Modul Notifikasi dan Komunikasi, menguji pengiriman notifikasi kepada pengguna.

### **Strategi dan Metode Pengujian**

Pengujian *Black Box* : merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program. Dengan kata lain pada pengujian *Black Box* dilakukan dengan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak.

### **Jadwal Pengujian**

(Sebutkan kasus data pengujiannya)

Tabel 1. .............

| **Use Case** | **PIC** | **Jadwal pengujian** |
| --- | --- | --- |
| *Login* |  | *28 Maret 2016* |
| Use Case lainnya.. |  |  |

# **Pelaksanaan Pengujian**

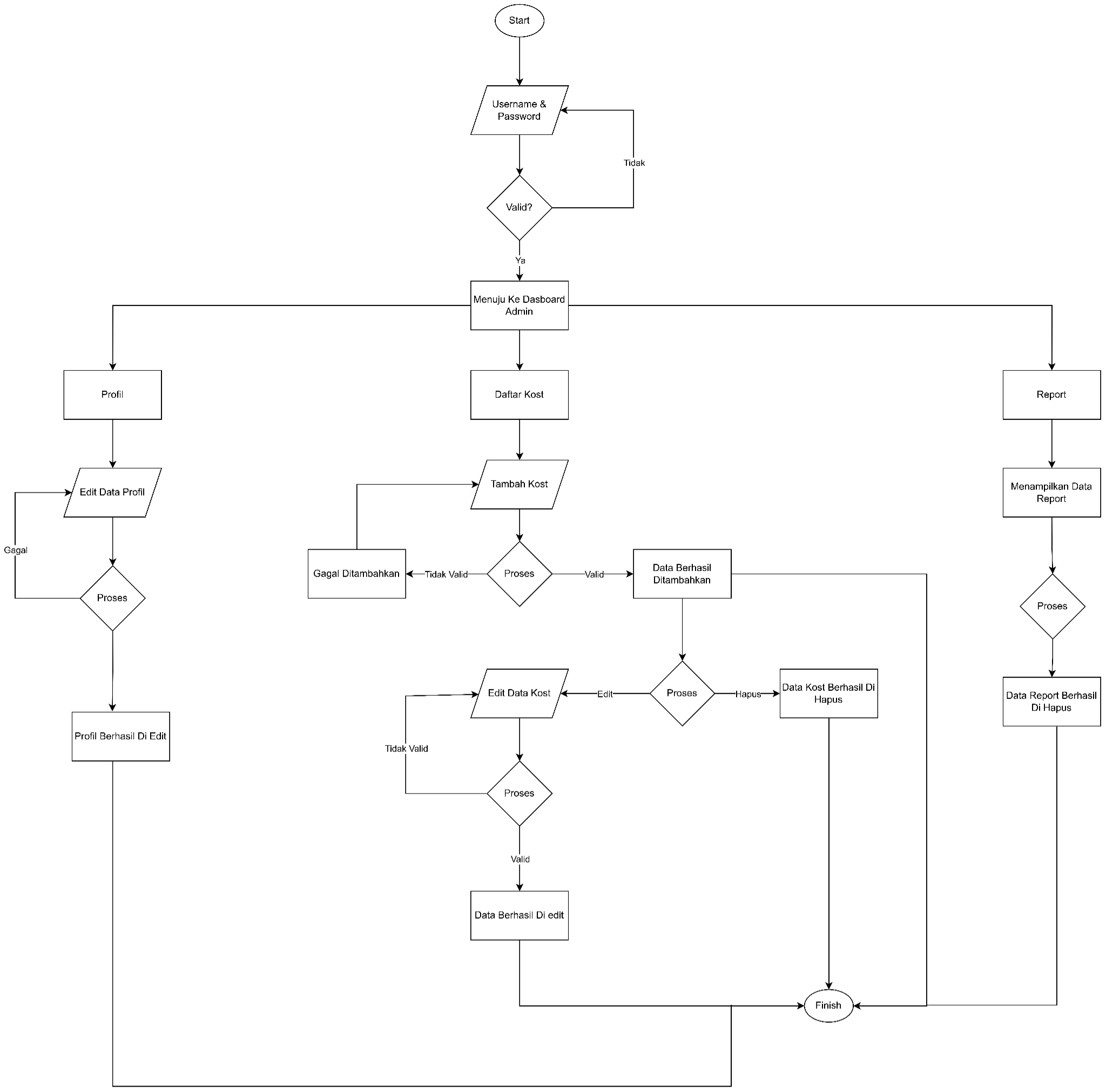
Pengujian yang dilakukan pada website Kost Kita diantaranya adalah login user sebagai calon penyewa dan admin, fitur pencarian kost, proses pemesanan kost, dan pengujian pada modul laporan.

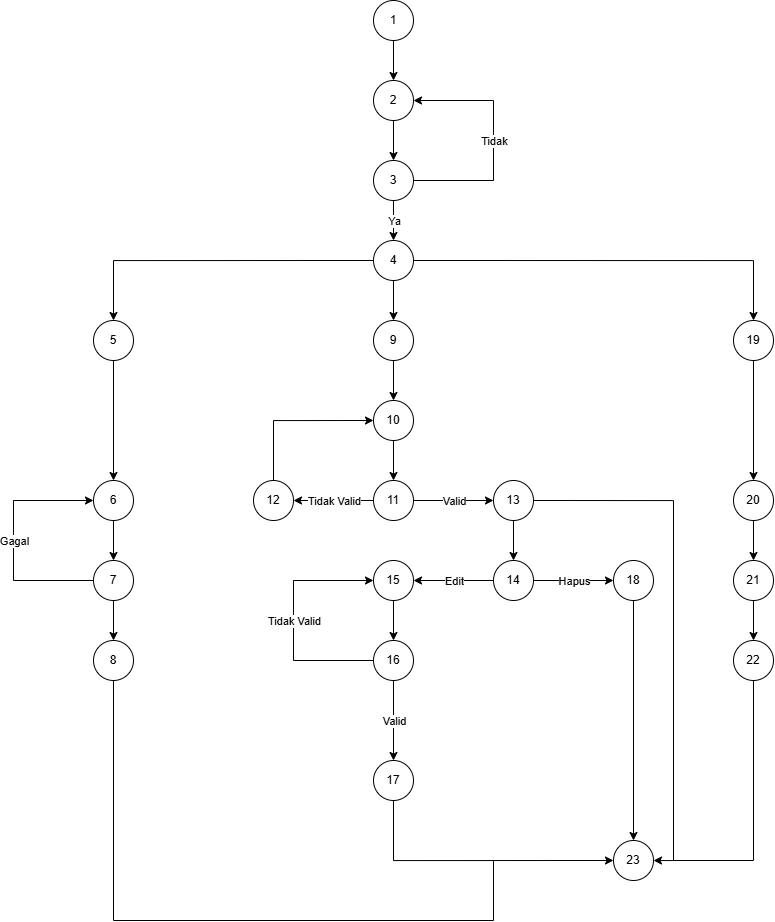
## ***Pengujian UNIT***

### **Pengujian White Box Method**

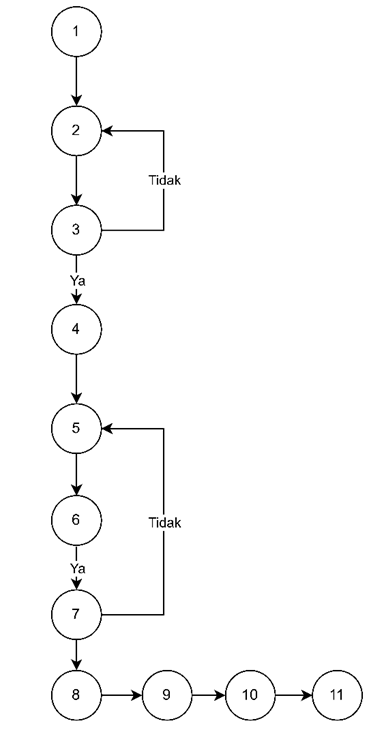
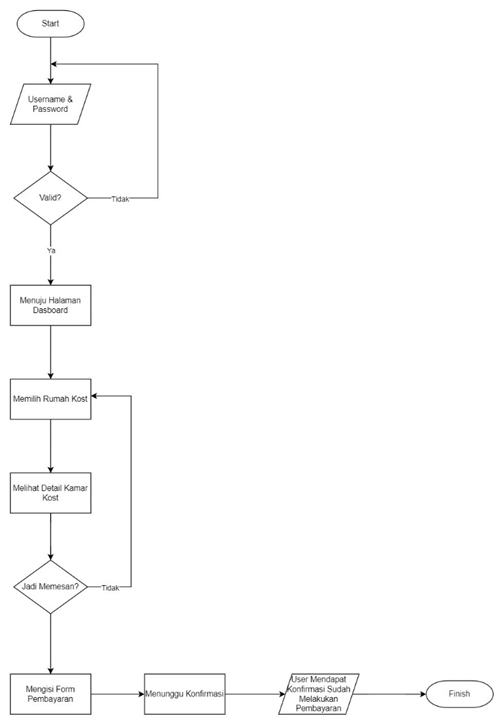
1. Pilih Method paling rumit dari class paling rumit
2. Buat Flowchart/Flowgraph Nya

* Admin





* User



1. Hitung cyclomatic complexity nya

| **Admin** | **User** |
| --- | --- |
| V (G) = E - N + 2  V(G) = 30 - 23 + 2  V(G) = 9 | V (G) = E - N + 2  V(G) = 12 - 11 + 2  V(G) = 3 |

1. Daftar Path yang perlu diuji

* Admin

| No | Independent Path |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9. | 1-2-3-2 (Admin gagal login)  1-2-3-4-5-6-7-6 (Admin gagal mengedit data profil)  1-2-3-4-5-6-7-8-23 (Admin berhasil mengedit data profil)  1-2-3-4-9-10-11-12-10 (Admin gagal menambahkan data kost)  1-2-3-4-9-10-11-13-23 (Admin berhasil menambahkan data kost)  1-2-3-4-9-10-11-13-14-15-16-15 (Admin gagal mengedit data kost)  1-2-3-4-9-10-11-13-14-15-16-17-23 (Admin berhasil mengedit data kost)  1-2-3-4-9-10-11-13-14-18-23 (Admin berhasil menghapus data kost)  1-2-3-4-19-20-21-22-23 (Admin berhasil menampilkan laporan data kost dan berhasil menghapusnya) |

* User

| No | Independent Path |
| --- | --- |
| 1.  2.  3. | 1-2-3-2 (User gagal login)  1-2-3-4-5-6-7-5 (User tidak jadi melakukan pemesanan kamar kost)  1-2-3-4-5-6-7-8-9-10-11 (User berhasil melakukan pemesanan kamar kost hingga ke pembayaran) |

1. Siapkan data uji untuk setiap path

* Admin

| Path | 1 |
| --- | --- |
| Jalur | 1-2-3-2 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  2. Jika tidak valid, Admin akan diminta kembali untuk memasukkan username dan password. |
| Hasil Pengujian | Gagal |

| Path | 2 |
| --- | --- |
| Jalur | 1-2-3-4-5-6-7-6 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  5.Admin memilih menu profil.  6.Admin melakukan edit data profil  7.Setelah selesai, data disimpan. Jika penyimpanan gagal  6.Admin akan diarahkan kembali ke halaman edit profil |
| Hasil Pengujian | Gagal |

| Path | 3 |
| --- | --- |
| Jalur | 1-2-3-4-5-6-7-8-23 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  5.Admin memilih menu profil.  6.Admin melakukan edit data profil  7.Setelah selesai, data disimpan  8.Sistem menampilkan notifikasi bahwa profil berhasil diedit.  23.Finish |
| Hasil Pengujian | Berhasil |

| Path | 4 |
| --- | --- |
| Jalur | 1-2-3-4-9-10-11-12-10 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  9.Admin memilih menu daftar kost  10.Admin memilih opsi tambah kost  11.Sistem memproses data yang dimasukkan.  12. Jika data tidak valid, sistem menampilkan notifikasi bahwa penambahan data kost gagal.  10. Admin diarahkan kembali ke halaman tambah kost. |
| Hasil Pengujian | Gagal |

| Path | 5 |
| --- | --- |
| Jalur | 1-2-3-4-9-10-11-13-23 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  9.Admin memilih menu daftar kost  10.Admin memilih opsi tambah kost  11.Sistem memproses data yang dimasukkan.  13.Jika data valid, sistem menampilkan notifikasi bahwa data berhasil ditambahkan.  23.Finish |
| Hasil Pengujian | Berhasil |

| Path | 6 |
| --- | --- |
| Jalur | 1-2-3-4-9-10-11-13-14-15-16-15 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  9.Admin memilih menu daftar kost  10.Admin memilih opsi tambah kost  11.Sistem memproses data yang dimasukkan.  13.Jika data valid, sistem menampilkan notifikasi bahwa data berhasil ditambahkan.  14.Admin memilih opsi edit data kost  15. Admin melakukan perubahan pada data kost.  16.Sistem memproses data yang diubah.  15.Jika data tidak valid, sistem akan mengarahkan Admin kembali ke halaman edit data kost. |
| Hasil Pengujian | Gagal |

| Path | 7 |
| --- | --- |
| Jalur | 1-2-3-4-9-10-11-13-14-15-16-17-23 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  9.Admin memilih menu daftar kost  10.Admin memilih opsi tambah kost  11.Sistem memproses data yang dimasukkan.  13.Jika data valid, sistem menampilkan notifikasi bahwa data berhasil ditambahkan.  14.Admin memilih opsi edit data kost  15. Admin melakukan perubahan pada data kost.  16.Sistem memproses data yang diubah.  17.Jika data valid, sistem menampilkan notifikasi bahwa data berhasil diedit.  23.Finish |
| Hasil Pengujian | Berhasil |

| Path | 8 |
| --- | --- |
| Jalur | 1-2-3-4-9-10-11-13-14-18-23 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  9.Admin memilih menu daftar kost  10.Admin memilih opsi tambah kost  11.Sistem memproses data yang dimasukkan.  13.Jika data valid, sistem menampilkan notifikasi bahwa data berhasil ditambahkan.  14.Admin memilih opsi hapus data kost  18. Data kost berhasil dihapus, dan sistem menampilkan notifikasi konfirmasi penghapusan.  23.Finish. |
| Hasil Pengujian | Berhasil |

| Path | 9 |
| --- | --- |
| Jalur | 1-2-3-4-19-20-21-22-23 |
| Skenario | 1.Start  2.Admin memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, Admin diarahkan ke halaman dashboard.  19.Admin memilih menu report  20.Sistem menampilkan data report.  21.Admin melakukan proses penghapusan data report.  22.Data report berhasil dihapus, dan sistem menampilkan notifikasi konfirmasi penghapusan.  23.Finish |
| Hasil Pengujian | Berhasil |

* User

| Path | 1 |
| --- | --- |
| Jalur | 1-2-3-2 |
| Skenario | 1.Start  2.User memasukan username dan password  3. Sistem memeriksa validitas username dan password.  2. Jika tidak valid, user akan diminta kembali untuk memasukkan username dan password. |
| Hasil Pengujian | Gagal |

| Path | 2 |
| --- | --- |
| Jalur | 1-2-3-4-5-6-7-5 |
| Skenario | 1.Start  2.User memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, user diarahkan ke halaman dashboard.  5.User memilih rumah kost.  6. User melihat detail kamar kost yang dipilih.  7. Apakah ingin memesan? Jika tidak  5. User kembali ke langkah memilih rumah kost. |
| Hasil Pengujian | Berhasil |

| Path | 3 |
| --- | --- |
| Jalur | 1-2-3-4-5-6-7-8-9-10-11 |
| Skenario | 1.Start  2.User memasukan username dan password  3. Sistem memeriksa validitas username dan password.  4. Jika valid, user diarahkan ke halaman dashboard.  5.User memilih rumah kost.  6. User melihat detail kamar kost yang dipilih.  7. Apakah ingin memesan? Jika iya, lanjut ke langkah berikutnya.  8. User diminta mengisi formulir pembayaran.  9.Menunggu Konfirmasi dari sistem.  10. User menerima konfirmasi bahwa pembayaran telah berhasil.  11.Finish |
| Hasil Pengujian | Berhasil |

1. Tampilkan screenshot hasilnya.

### **Pengujian Class dengan JUnit/PhPUnit**

Jelaskan di sini contoh pengujian sebuah class.

Setiap methodnya perlu diuji, dengan data uji yang membuat VALID atau yang membuat FAIL.

**Tabel 2 Pengujian Class**

| **CLASS** | **Method** |  | **Kasus dan Hasil Uji (Data normal)** | |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan\*** |  | **Kesimpulan** |
|  |  | User ID: Universitas  Nama : xyz | Form menampilkan data user baru untuk *user* | Dapat melakukan | [X | ] diterima |
|  |  | password: rahasia | peneliti dan responden | pengisian data *user* | [ | ] ditolak |
|  |  |  |  | baru |  |  |
|  |  | konfirmasi password:rahasia(hasil pada lampiran B(Gambar B.2 dan B.4)) |  | Sesuai yang diharapkan |  |  |
|  |  | Klik tombol simpan | Data tersimpan di file *User* peneliti dan *user* | Data pengisian *user* | [X | ] diterima |
|  |  | (hasil pada lampiran | Responden | Responden dan peneliti | [ | ] ditolak |
|  |  | B(Gambar B.3 dan B.6)) |  | tersimpan |  |  |
|  |  |  |  | Sesuai yang diharapkan |  |  |
|  |  | Klik tombol Reset | Data yang telah terisi telah dihapus | Data telah terhapus | [X | ] diterima |
|  |  | (hasil pada lampiran |  | sesuai yang diharapkan | [ | ] ditolak |
|  |  | B(Gambar B.7 dan B.8 ) |  |  |  |  |

\* **CONTOH PENGUJIAN DENGAN JUNIT/PHPUNIT DILAMPIRKAN**

1. Contoh Code JUnit/PhpUnit untuk pengujian Class : (Sebutkan NamaClassnya)



1. Screenshoot hasil pengujian JUnit



## ***Pengujian INTEGRASI***

### Pengujian DUPL-01 Login User

Pengujian *login* terbagi menjadi dua bagian diantaranya pendaftaran *user* baru dan pengecekan user yang telah terdaftar sebagai berikut :

**Tabel 1 Pengujian Login *User***

| **PROSES(DFD-1)/**  **USE CASE** |  | **Kasus dan Hasil Uji (Data normal)** | |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** |  | **Kesimpulan** |
|  | Username: bernadya1@gmail.com  password: bernadya123! | Pengguna berhasil login dan diarahkan ke halaman dashboard. | Kredensial diterima dan sistem mengizinkan akses. | [X | ] diterima |
|  | Klik tombol masuk | Dapat menampilkan notifikasi berhasil *login* | Berhasil melakukan *login* | [X | ] diterima |
|  |  |  |  |  |  |

#### Pengujian DUPL-01 Pendaftaran *User* Baru

Berikut ini adalah tabel pengujian *login* untuk pendaftaran *user* baru :

**Tabel 2 Pengujian Pendaftaran *User* Baru**

| **PROSES(DFD-1)/**  **USE CASE** |  | **Kasus dan Hasil Uji (Data normal)** | |  |
| --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
|  | Username: Bernadya  Email:bernadya1@gmail.com  Password: bernadya123!  Konfirm password: bernadya123! | Akun berhasil dibuat, muncul pesan sukses, pengguna diarahkan ke halaman login. | Semua masukan diterima sesuai validasi. | [X] diterima |
|  | Klik tombol daftar | Dapat menampilkan notifikasi berhasil *register* | Berhasil melakukan *register* | [X] diterima |
|  |  |  |  |  |

| Use CASE |  | **Kasus dan Hasil Uji (Data salah:)** | |  |
| --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
|  | Data Login *User* dan Responden tidak terdaftar.  Username : aaa  Password : aaa | Tidak dapat *login* dan Menampilkan pesan “Login Gagal” | *User* tidak dapat *login* dan memberikan pesan ”login gagal” sesuai yang diharapkan | [X] diterima |
|  | Data Login *User* dan Responden telah terdaftar :  Username: universitas | Tidak dapat melakukan pendaftaran karena username telah terdaftar | Mengeluarkan pesan User Id telah terdaftar | [X] diterima |
|  | Nama : xyz  password : rahasia |  |  |  |
|  |  |  |  |  |

### Pengujian DUPL-01 Pencarian Kost

Berikut ini adalah tabel pengujian *pencarian kost* untuk mencari kamar:

**Tabel 3 Pengujian Pencarian Kost**

| **PROSES(DFD-1)/**  **USE CASE** |  | **Kasus dan Hasil Uji (Data normal)** | |  |
| --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
|  | Search: “Karanglewas” | Sistem mencari lokasi dari kata kunci yang diberikan | Sistem menampilkan hasil relevan sesuai kata kunci. | [X] diterima |
|  | Klik tombol search | Dapat menampilkan daftar kost yang sesuai dengan lokasi atau nama yang ditampilkan. | Berhasil menampilkan daftar kost | [X] diterima |
|  |  |  |  |  |

| **USE CASE** |  | **Kasus dan Hasil Uji (Data salah)** | |  |
| --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
|  | Search: Kota | Tidak akan menampilkan halaman pencarian dengan kata kunci yang diberikan | Berhasil memberikan pesan ”Data yang dicari tidak terdaftar” | [X] diterima |
|  | Klik tombol search | Dapat menampilkan pesan “Data yang dicari tidak terdaftar” | Berhasil memberikan pesan ”Data yang dicari tidak terdaftar” | [X] diterima |
|  |  |  |  |  |

### 

### **Pengujian DUPL-01** Pemesanan Kost

Berikut ini adalah tabel pengujian *pemesanan kost* untuk memesan kamar:

**Tabel 4 Pengujian Pemesanan Kost**

| **PROSES(DFD-1)/**  **USE CASE** |  | **Kasus dan Hasil Uji (Data normal)** | |  |
| --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
|  | Dropdown no kamar: No. 5 | Dropdown menampilkan semua opsi nomor kamar yang tersedia. | Nomor kamar dapat dipilih dengan  benar tanpa error. | [X] diterima |
|  | Dropdown tipe kamar: Single Bed | Dropdown menampilkan semua tipe kamar yang tersedia. | Tipe kamar dapat dipilih dengan benar tanpa error. | [X] diterima |
|  | Dropdown pembayaran: Transfer bank | Dropdown menampilkan semua metode pembayaran yang tersedia. | Metode pembayaran dapat dipilih dengan benar tanpa error. | [X] diterima |
|  |  |  |  |  |

### 

### Pengujian DUPL-01 Pembayaran Kost

Berikut ini adalah tabel pengujian *pembayaran kost:*

**Tabel 4 Pengujian Pembayaran Kost**

| **PROSES(DFD-1)/**  **USE CASE** |  | **Kasus dan Hasil Uji (Data normal)** | |  |
| --- | --- | --- | --- | --- |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
|  | Klik opsi bank yang tersedia (Mandiri, BCA, BNI, BRI, Bank Lainnya).  Pilih: Mandiri | Pengguna dapat memilih salah satu bank tujuan tanpa error. | Bank yang dipilih ditandai dengan benar. | [X] diterima |
|  | Klik tombol "Bayar dengan Transfer Mandiri" | Sistem melanjutkan proses ke langkah berikutnya (konfirmasi pembayaran). | Tombol berfungsi tanpa error dan melanjutkan proses pembayaran.. | [X] diterima |
|  | Klik tombol "Ya, saya sudah bayar" | Sistem menampilkan pesan konfirmasi dan melanjutkan proses ke halaman verifikasi. | Tombol bekerja tanpa error dan memproses konfirmasi pembayaran. | [X] diterima |
|  |  |  |  |  |

## ***SYSTEM TEST (Optional, jika software embedded di system IOT dll)***

PENGUJIAN SISTEM OLEH : PETUGAS PELAKSANA PENGUJIAN

TANGGAL : ......

| Komponen Sistem | Fitur Komponen Yang Diuji | Rencana pengujiannya | Hasil Yang Diharapkan | Hasil Pengujian | Kesimpulan |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## ***USER ACCEPTANCE TEST (Berjuang mencari user potensial)***

USER ACCEPTANCE TEST OLEH : PETUGAS PELAKSANA PENGUJIAN

TANGGAL : ......

| Use Case Yang Diuji | Rencana Pengujian | Hasil Yang Diharapkan | Hasil Aktual | Kesimpulan |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## ***Summary Pengujian Yang Masih Gagal***

(Berisi laporan dari pengujian yang telah dilakukan, dengan menyampaikan informasi status dari setiap fungsional yang masih gagal)

| **Kelas Uji** | **Butir Uji** | **Kesimpulan pengujian** |
| --- | --- | --- |
| Uji sensor ketinggian air | Tetesan air dianggap banjir | Ditolak |
|  |  |  |
|  |  |  |
|  |  |  |

# **Lampiran**

1. Capture /screenshot hasil pengujian modul-modul penting
2. Hasil pengukuran OOMetric aplikasi yang telah berhasil dibangun dengan software (tool) pengukuran OOMetric (lihat [http://www.virtualmachinery.com](http://www.virtualmachinery.com/))