

GL03

DOKUMEN UJI PERANGKAT LUNAK

<TEKOM GYM>

untuk:

Pusat Kebugaran Tekkom Gym

Purwokerto, Jawa Tengah

Dipersiapkan oleh:

SINGGIH PRAWIRO NEGORO (2211102341)


M . ALFAREZ PAHLEVI TANJUNG (2211102069)

HAYYAN NASHRULLOH (2211102100)

NIZAR QULUBI (2211102096)

HIZAM ARIPIN (2211102317)

Prodi Teknik Informatika - Universitas Telkom

	Prodi S1 Teknik Informatika Universitas Telkom	Nomor Dokumen		Halaman
		<i>DUPL-001</i>		1/
		Revisi	-	DESEMBER 2024

Daftar Isi

1	Pendahuluan	4
1.1	Tujuan Pembuatan Dokumen	4
1.2	Deskripsi Umum Sistem	4
1.3	Definisi dan Singkatan.....	Error! Bookmark not defined.
1.4	Referensi.....	4
2	Lingkungan Pengujian Perangkat Lunak.....	Error! Bookmark not defined.
2.1	Perangkat Lunak Pengujian	4
2.2	Perangkat Keras Pengujian	4
2.3	Material Pengujian.....	4
2.4	Sumber Daya Manusia.....	4
2.5	Metode Pengujian	5
2.6	Tujuan Pengujian	Error! Bookmark not defined.
2.7	Rencana Pengujian	7
2.8	Pelaksanaan Pengujian	7
2.8.1	Pengujian DUPL-01 Login User	10
2.9	Kesimpulan Pengujian.....	14
3	Lampiran	14

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

1 Pendahuluan

1.1 Tujuan Pembuatan Dokumen

- Membantu pengguna memahami fungsi dan fitur web aplikasi T-Kom Gym.
- Menyediakan langkah-langkah operasional untuk memanfaatkan aplikasi secara efektif.

1.2 Ruang Lingkup Pengujian

1.3 Referensi

Menjelaskan referensi SKPL & DPPL yang akan digunakan sebagai rujukan pembuatan dokumen ini

1.4 Overview Sistem & Fitur Utamanya

Menjelaskan deskripsi umum dari sistem/perangkat lunak yang diuji.

1.5 Overview Pengujian

1. Verilbilty testing dan penetration testing
- 2 blackbox white box testing
- 3 user acceptance testing dan usability testing

1.5.1 Perangkat Keras Pengujian

1. Server Pengujian:

- **Prosesor:** Intel Xeon E5-2620 v4 atau setara (6 core, 2.10 GHz)
- **RAM:** 16 GB
- **Penyimpanan:** SSD 500 GB
- **Sistem Operasi:** Linux Ubuntu Server 20.04 LTS
- **Jaringan:** Koneksi internet minimal 100 Mbps
- Perangkat keras yang digunakan dipilih untuk mensimulasikan berbagai kondisi pengguna akhir dan lingkungan operasional.
- Pengujian dilakukan pada konfigurasi perangkat keras minimum dan optimal untuk mengevaluasi kinerja aplikasi di berbagai skenario.

1.5.2 Sumber Daya Manusia

Menjelaskan sumber daya manusia yang terlibat dalam pengujian perangkat lunak

a.sdm untuk venebilty testing dan penetration testing

b. blackbox white box testing

c. user acceptance testing dan usability testing

1.5.3 Perangkat Lunak Pengujian

Nessus

Owasp zap

1.5.4 Material Pengujian

Menjelaskan modul – modul yang akan diuji

Prodi Informatika – Universitas Telkom	DUPL-xxx	Halaman 4 dari 16
--	----------	-------------------

1.5.5 Strategi dan Metode Pengujian

Strategi Pengujian

Pengujian dilakukan secara bertahap untuk memastikan bahwa setiap komponen sistem memenuhi persyaratan fungsional dan non-fungsional. Strategi pengujian mencakup:

1. Pengujian Unit (Unit Testing):

- Fokus pada pengujian setiap unit atau komponen individu secara terisolasi.
- Dilakukan oleh pengembang untuk memastikan setiap fungsi atau modul berjalan sesuai dengan spesifikasinya.

2. Pengujian Integrasi (Integration Testing):

- Menguji interaksi antar modul untuk memastikan data dan alur kerja dapat berjalan dengan lancar.
- Menggunakan pendekatan top-down atau bottom-up sesuai dengan kebutuhan sistem.

3. Pengujian Sistem (System Testing):

- Memastikan bahwa seluruh sistem berjalan sebagai satu kesatuan yang terpadu.
- Menguji semua fitur utama, seperti login, registrasi, pemesanan, dan manajemen admin.

4. Pengujian Penerimaan Pengguna (User Acceptance Testing):

- Melibatkan pengguna potensial untuk memastikan sistem sesuai dengan kebutuhan dan harapan mereka.
- Dilakukan dalam lingkungan yang menyerupai penggunaan sistem sebenarnya.

5. Pengujian Non-Fungsional:

- **Kinerja (Performance Testing):** Menguji waktu respon dan stabilitas sistem di bawah beban tinggi.
- **Keamanan (Security Testing):** Memastikan data pengguna aman dari ancaman eksternal.
- **Kompatibilitas (Compatibility Testing):** Memastikan sistem berjalan dengan baik di berbagai perangkat dan browser.

Metode Pengujian

1. Black-Box Testing:

- Metode ini digunakan untuk menguji fungsi-fungsi sistem tanpa memeriksa kode sumber.
- Fokus pada input dan output sesuai spesifikasi.
- Contoh: Validasi login, proses registrasi, dan pemesanan.

2. White-Box Testing:

- Menguji logika internal, alur program, dan struktur kode sumber.
- Digunakan dalam pengujian unit dan integrasi.

3. Manual Testing:

- Dilakukan oleh penguji manusia dengan mengikuti skenario yang telah ditentukan.

- Contoh: Memeriksa UI/UX dan memastikan tombol bekerja dengan baik.

4. **Automated Testing:**

- Menggunakan alat otomatisasi seperti Selenium untuk menguji skenario yang berulang dan kompleks.
- Cocok untuk pengujian regresi dan beban.

5. **Exploratory Testing:**

- Pengujian tanpa skenario yang terdefinisi sebelumnya untuk menemukan bug yang tidak terduga.
- Dilakukan oleh penguji dengan memahami tujuan sistem.

Alur Pengujian

1. **Persiapan:**

- Menyusun rencana pengujian berdasarkan kebutuhan sistem.
- Menentukan alat dan lingkungan pengujian.

2. **Pelaksanaan Pengujian:**

- Melakukan pengujian unit, integrasi, sistem, dan penerimaan pengguna secara bertahap.
- Mencatat hasil pengujian dan bug yang ditemukan.

3. **Evaluasi dan Perbaikan:**

- Meninjau hasil pengujian dan mengidentifikasi area perbaikan.
- Melakukan iterasi pengujian hingga semua fungsi berjalan dengan baik.

4. **Laporan:**

- Menyusun laporan akhir yang mencakup hasil pengujian, temuan, dan rekomendasi perbaikan.

Strategi dan metode ini dirancang untuk memastikan bahwa sistem memenuhi standar kualitas dan memberikan pengalaman terbaik kepada pengguna.

1.5.6 Jadwal Pengujian

(Sebutkan kasus data pengujiannya)

Tabel 1.

Use Case	PIC	Jadwal pengujian
Login		28 Maret 2016
Use Case lainnya..		

2 Pelaksanaan Pengujian

(Uraikan pengujian sesuai dengan butir uji yang sudah didefinisikan dalam rencana pengujian)

2.1 Pengujian UNIT

2.1.1 Pengujian White Box Method

- Pilih Method paling rumit dari class paling rumit
- Buat Flowchart/Flowgraphnya
- Hitung cyclomatic complexitynya
- Daftar Path yang perlu diuji
- Siapkan data uji untuk setiap path
- Tampilkan screenshot hasilnya.

2.1.2 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
LoginRegister	login(String username, String password)	Username: user123 Password: ValidPass123	Pengguna berhasil login ke dashboard user	Pengguna berhasil masuk tanpa pesan kesalahan	Sesuai
	register(String username, String password)	Username: newuser	Akun berhasil	Akun baru dibuat dan	

CLASS	Method	Kasus dan Hasil Uji (Data normal)			
		Data Masukan	Yang diharapkan	Pengamatan*	Kesimpulan
	password, String email	Password: StrongPass123 Email: newuser@mail.com	dibuat dan data disimpan di Sesuai database	diarahkan ke halaman login	
Admin	addProduct(String name, double price)	Nama Produk: Barbell Harga: 500000	Produk baru berhasil ditambahkan	Produk berhasil ditambahkan ke database dan dapat terlihat di daftar produk	Sesuai
	deleteProduct(int productId)	ID Produk: 101	Produk berhasil dihapus	Produk dengan ID 101 berhasil dihapus dari database	Sesuai

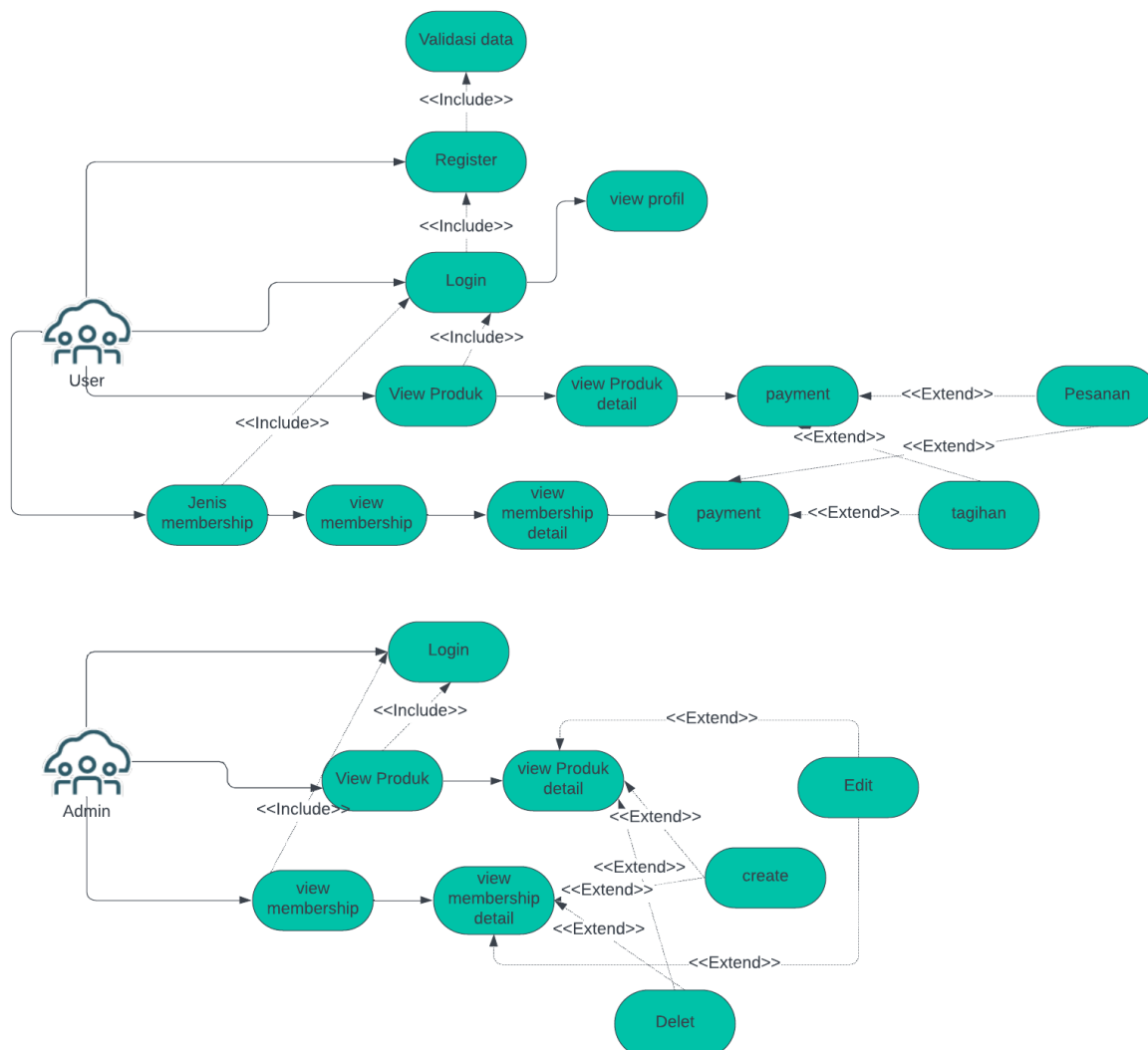
*** CONTOH PENGUJIAN DENGAN JUNIT/PHPUNIT DILAMPIRKAN**

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

B. Screenshoot hasil pengujian JUnit

	Port	Protocol	State	Service
	6001	tcp	open	X11:1
	6000	tcp	open	X11
	5357	tcp	open	wsdapi
	1755	tcp	open	wms
	5800	tcp	open	vnc-http
	5900	tcp	open	vnc
	5000	tcp	open	upnp
	1900	tcp	open	upnp
	37	tcp	open	time
	23	tcp	open	telnet
	427	tcp	open	svrloc
	8888	tcp	open	sun-answerbook
	587	tcp	open	submission
	22	tcp	open	ssh
	3128	tcp	open	squid-http
	444	tcp	open	snpp
	10000	tcp	open	snet-sensor-mgmt
	199	tcp	open	smux
	465	tcp	open	smtps
	25	tcp	closed	smtp
	5060	tcp	open	sip
	514	tcp	open	shell
	554	tcp	open	rtsp
	873	tcp	open	rsync
	26	tcp	open	rsftp
	111	tcp	open	rpcbind
	7070	tcp	open	realserver
	4899	tcp	open	radmin

2.2 Pengujian INTEGRASI



2.2.1 Pengujian DUPL-01 Login User

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

PROSES(DFD-1)/ USE CASE	Kasus dan Hasil Uji (Data normal)			
	Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
	Username: user123 Password: Password123	Pengguna berhasil login ke dashboard user	Pengguna berhasil masuk ke dashboard tanpa ada pesan kesalahan.	Sesuai
	Username: user123 Password: wrongpass	Login gagal dengan pesan kesalahan	Sistem menampilkan pesan kesalahan sesuai, dan	Sesuai

PROSES(DFD-1)/ USE CASE	Kasus dan Hasil Uji (Data normal)			
	Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
		<i>"Username atau password salah."</i>	login tidak berhasil.	
	Username: newuser Email: newuser@mail.com Password: NewPass123	Akun berhasil dibuat dan data disimpan di database	Akun baru dibuat, pengguna diarahkan ke halaman login atau dashboard.	Sesuai

2.2.1.1 Pengujian DUPL-01_01 Pendaftaran *User* Baru

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

Tabel 2 Pengujian Pendaftaran *User* Baru

USE CASE	Kasus dan Hasil Uji (Data normal)			
	Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
	Username: testuser Password: ValidPass123	Pengguna berhasil login ke dashboard user	Pengguna berhasil masuk ke dashboard tanpa pesan kesalahan.	Sesuai
	Username: testuser Password: wrongpassword	Login gagal dengan pesan kesalahan <i>"Username atau password salah."</i>	Sistem menampilkan pesan kesalahan sesuai, dan login tidak berhasil.	Sesuai
	Username: newuser	Akun berhasil dibuat dan data	Akun berhasil	Sesuai

USE CASE	Kasus dan Hasil Uji (Data normal)			
	Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
	Email: newuser@mail.com Password: NewPass123	disimpan di database	dibuat, pengguna diarahkan ke halaman login atau dashboard.	

Use CASE	Kasus dan Hasil Uji (Data salah:)			
	Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
	Username: invalidUser Password: wrongPass	Login gagal dengan pesan kesalahan <i>"Username atau password salah."</i>	Sistem menampilkan pesan kesalahan sesuai, dan login tidak berhasil.	Sesuai
	Klik tombol bayar dengan keranjang belanja kosong	Sistem menolak dengan pesan <i>"Keranjang belanja kosong, tidak bisa melanjutkan pembayaran."</i>	Sistem menampilkan pesan kesalahan sesuai, dan pembayaran tidak berhasil.	Sesuai

2.3 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

2.4 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
Membuka halaman utama (homepage)	Mengakses homepage untuk memeriksa tampilan dan fitur yang tersedia	Homepage terbuka dengan semua elemen terlihat jelas dan dapat diakses	Homepage terbuka dengan beberapa elemen memerlukan waktu lebih lama untuk dimuat	Perlu optimasi performa untuk memuat elemen lebih cepat
Melakukan registrasi pengguna	Memasukkan data valid ke form registrasi dan klik tombol "Daftar"	Akun berhasil dibuat dan pengguna diarahkan ke halaman login	Akun berhasil dibuat tanpa kendala	Sesuai dengan ekspektasi
Login menggunakan akun terdaftar	Memasukkan username dan password yang valid	Pengguna berhasil login ke dashboard	Pengguna berhasil login tanpa kendala	Sesuai dengan ekspektasi
Memesan paket wisata	Memilih paket wisata, mengisi informasi pemesanan, dan klik tombol "Pesan"	Pemesanan berhasil dan pengguna mendapatkan konfirmasi	Pemesanan berhasil, tetapi beberapa detail konfirmasi kurang jelas	Perlu perbaikan pada detail informasi konfirmasi

Logout dari dashboard	Klik tombol "Logout" setelah login	Sesi pengguna diakhiri, dan pengguna diarahkan ke halaman login	Logout berhasil tanpa kendala	Sesuai dengan ekspektasi
-----------------------	------------------------------------	---	-------------------------------	--------------------------

2.5 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

3 Lampiran

Share result via url: <https://clickjacker.io/test?url=https://tekkomgym.pythonanywhere.com/>
COPY

Out

Off

0

0

4

4

2

+

TEKKOM-GYM

Home

Anggota

Tentang Kami

Contact

Kelas

Tekkom Gym

Bakar kalori dengan senang dan ringan

Dengan latihan yang dilakukan oleh instruktur yang kompeten, kami menjamin hasil yang anda inginkan akan tercapai

Test Results:

Site: <https://tekkomgym.pythonanywhere.com/>

IP Address: 35.173.69.207

Time: Tue Nov 19 2024 23:48:01 GMT+0000 (Coordinated Universal Time)

X-Frame-Options: ✖ Missing header

CSP Header (Frame-Ancestors): ✖ Missing anti-framing policy

Total scans so far: 2,730,647

History 42 WebSockets

It is vulnerable to clickjacking attack

Sites

Start

Start

Start

Off

0

3

2

4

+

Berdasarkan hasil pengujian (*Penetration Test*), berikut adalah temuan kerentanannya:

Kerentanan yang Ditemukan

1. X-Frame-Options Header Missing

- **Deskripsi:**
Header *X-Frame-Options* tidak ada, sehingga situs ini rentan terhadap serangan **Clickjacking**.
- **Dampak:**
Penyerang dapat membuat situs palsu untuk menipu pengguna agar berinteraksi dengan elemen yang tersembunyi dalam iframe, misalnya, klik tombol tanpa sepengetahuan pengguna.

2. CSP Header (Frame-Ancestors) Missing

- **Deskripsi:**
Kebijakan CSP (Content Security Policy) yang mengatur *frame-ancestors* tidak diterapkan. Tanpa kebijakan ini, situs dapat di-embed oleh domain tidak sah, meningkatkan risiko serangan Clickjacking.

Rekomendasi Perbaikan

1. Tambahkan Header X-Frame-Options:

Situs tidak memiliki header *X-Frame-Options*, sehingga rentan terhadap serangan clickjacking.

- **Rekomendasi Perbaikan:**

Tambahkan header *X-Frame-Options* untuk mencegah situs ditampilkan dalam iframe domain lain, misalnya:

http

X-Frame-Options: SAMEORIGIN

2. Implementasikan CSP untuk Frame-Ancestors:

Kebijakan Content Security Policy tidak diterapkan untuk membatasi penggunaan *frame-ancestors*.

- **Rekomendasi Perbaikan:**

Tambahkan kebijakan CSP untuk mengontrol domain yang dapat menampilkan situs, contohnya:

http

Content-Security-Policy: frame-ancestors 'self';

2. Vulnerability Test

Berdasarkan hasil pengujian menggunakan *clickjacker.io* ([t-kom -gym](https://t-kom-gym.clickjacker.io)), ditemukan kerentanan berikut:

Kerentanan yang Ditemukan

Kerentanan	Tingkat Risiko
Cloud Metadata Potentially Exposed	Tinggi
Content Security Policy (CSP) Header Not Set	Sedang
Missing Anti-clickjacking Header	Sedang
Hidden File Found	Sedang
Cookie No HttpOnly Flag	Rendah
Cookie without SameSite Attribute	Rendah

Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Rendah
X-Content-Type-Options Header Missing	Rendah

A. Situs Uji: <https://tekkomgym.pythonanywhere.com>