

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Aplikasi informasi peminjaman ruangan dan inventaris FTTI UNJAYA dikembangkan menggunakan metode Waterfall dengan *backend* berbasis Python (Flask), frontend HTML, CSS, dan JavaScript, serta *database* MySQL. Aplikasi ini merupakan digitalisasi dari proses peminjaman yang sebelumnya dilakukan secara manual melalui bagian operasional.

Aplikasi ini memungkinkan pengguna untuk mengajukan peminjaman ruangan atau barang sesuai kebutuhan serta melihat ketersediaan melalui fitur kalender. Admin sebagai aktor utama dapat mengelola data pengguna, ruangan, inventaris, serta menyetujui atau menolak permohonan peminjaman yang masuk.

4.2 IMPLEMENTASI DESAIN ANTARMUKA

Implementasi antarmuka pengguna merupakan bagian penting dalam pengembangan sistem karena menjadi titik interaksi langsung antara pengguna dan sistem. Antarmuka pada aplikasi ini dirancang dengan tampilan yang sederhana namun fungsional agar mudah digunakan oleh berbagai jenis pengguna, baik mahasiswa maupun admin.

4.2.1 Halaman Login

Sebelum dapat mengakses sistem, pengguna akan diarahkan ke halaman login sebagai tahap awal otentikasi. Baik pengguna maupun admin diwajibkan untuk memasukkan username dan kata sandi yang telah terdaftar di dalam basis data. Sistem akan melakukan validasi terhadap data tersebut untuk menentukan akses pengguna. Apabila informasi yang dimasukkan benar, maka pengguna akan diarahkan menuju halaman sesuai dengan perannya. Pengguna biasa akan masuk ke tampilan layanan utama seperti formulir peminjaman, sedangkan admin

diarahkan menuju halaman dashboard. Sebaliknya, apabila terjadi kesalahan input, sistem akan menampilkan pesan peringatan bahwa data login tidak valid.

Halaman login dirancang dengan dua navigasi utama: Login & Panduan serta Kalender Ketersediaan. Navigasi ini memungkinkan pengguna untuk tidak hanya masuk ke sistem, tetapi juga dapat melihat kalender ketersediaan ruangan sebelum mengajukan permohonan peminjaman. Tampilan halaman ini dibuat sederhana, responsif, dan mendukung perangkat seluler agar mudah diakses oleh semua jenis pengguna. Gambar 4.1 memperlihatkan tampilan halaman login dan Gambar 4.2 menampilkan kalender.



Gambar 4. 1 Tampilan Halaman Login



Gambar 4. 2 Tampilan Halaman Kalender

Berikut Merupakan kode untuk menampilkan halaman login.

```
@app.route('/login', methods=['POST'])
def login():
    username = request.form['username']
    password = request.form['password']

    user = User.query.filter_by(username=username).first()
    print(user.password, password, type(user.password),
          type(password), check_password_hash(user.password, password))

    if user and check_password_hash(user.password, password):
        session['username'] = user.username
        session['role'] = user.role
        session['user_id'] = user.id
        print("Login Berhasil")

    if user.role == 'admin':
        return redirect('/admin')
    else:
        return redirect('/user/peminjaman')

    return "Login gagal: Username atau password salah", 401
```

Dan kode kalender dibawah ini.

```
<script
src="https://cdn.jsdelivr.net/npm/fullcalendar@6.1.17/index.glob
al.min.js"></script>
<script>
document.addEventListener('DOMContentLoaded', function () {
    var calendarEl = document.getElementById('calendar');
    var calendar = new FullCalendar.Calendar(calendarEl, {
        initialView: 'dayGridMonth',
        locale: 'id',
        headerToolbar: {
            left: 'prev,next today',
            center: 'title',
            right: 'dayGridMonth,timeGridWeek,listWeek',
        },
        events: '/api/peminjaman',
    });
    calendar.render();
```

4.2.2 Halaman Form Pengajuan

Halaman pengajuan peminjaman terdiri dari beberapa fitur utama yang mendukung proses input data oleh pengguna. Fitur pertama adalah formulir peminjaman, yang memuat input tanggal peminjaman, pilihan ruangan dan barang, unggahan surat izin, serta kolom deskripsi acara. Form ini juga dilengkapi dengan

checkbox pernyataan tanggung jawab yang harus disetujui sebelum pengajuan dapat dikirim. Dan ada kalender ketersediaan, yang ditampilkan berdampingan dengan form. Kalender ini menampilkan jadwal penggunaan ruangan secara real-time, sehingga pengguna dapat menyesuaikan jadwal peminjaman dengan waktu yang tersedia. Kedua fitur ini dirancang untuk saling melengkapi guna mendukung proses peminjaman yang efisien dan informatif. Berikut Gambar 4.3 yang menampilkan tampilan pengajuan

Gambar 4. 3 Tampilan Halaman Form Pengajuan

```

if tanggal <= datetime.now().date():
    flash('Tanggal harus minimal besok.', 'Error')
    print("ini flash")
    return redirect('/user/peminjaman')

if tanggal < datetime.now().date() + timedelta(days=3):
    flash('Peminjaman hanya bisa dilakukan minimal 3
hari ke depan.', 'Error')
    return redirect('/user/peminjaman')

ada_peminjam = Peminjaman.query.filter_by(
    tanggal_pinjam=tanggal,
    ruangan_id=ruangan_id
).first()

if ada_peminjam:
    tanggal_alternatif = cari_tanggal_kosong(ruangan_id, tanggal + timedelta(days=3))
    if tanggal_alternatif:
        flash(f"Ruangan sudah terpakai. Coba tanggal
alternatif: {tanggal_alternatif.strftime('%Y-%m-%d')}", 'Error')
    else:

```

```

flash("Ruangan tidak tersedia selama 30 hari
ke depan.", 'Error')
return redirect('/user/peminjaman')

```

Kode tersebut melakukan validasi tanggal peminjaman ruangan. Pertama, sistem memastikan tanggal yang diajukan minimal besok dan setidaknya 3 hari dari hari ini. Jika tidak sesuai, pengguna diberi peringatan dan diarahkan kembali. Selanjutnya, sistem memeriksa apakah ruangan sudah dipinjam pada tanggal tersebut. Jika sudah, sistem mencari tanggal alternatif. Jika ditemukan, tanggal alternatif ditampilkan; jika tidak, pengguna diberi tahu bahwa ruangan tidak tersedia dalam 30 hari ke depan. Semua notifikasi ditampilkan melalui flash message.

4.2.3 Halaman Cek Status

Halaman ini menyediakan fitur tabel status peminjaman yang menampilkan daftar seluruh permohonan yang telah diajukan oleh pengguna. Informasi yang ditampilkan meliputi tanggal peminjaman, ruangan, barang yang dipinjam, status pengajuan (*pending*, *disetujui*, atau *ditolak*), serta tautan untuk melihat dokumen surat disposisi apabila pengajuan disetujui. Fitur ini dirancang untuk membantu pengguna dalam memantau perkembangan setiap permohonan tanpa perlu menghubungi admin secara langsung. Sistem juga menampilkan notifikasi atau pesan umpan balik apabila terjadi perubahan status, sehingga pengguna tetap mendapatkan informasi secara real-time.



Tanggal	Ruangan	Barang	Status	Surat Disposisi
08-07-2025	Laboratorium 3A	Sofa	Disetujui	Lihat
28-06-2025	Ruang Kelen 2B	Sofa	Pending	-

Gambar 4. 4 Tampilan Halaman Cek Status

```

<tbody>
{% for p in peminjaman %}
<tr>
<td>{{ p.tanggal_pinjam.strftime('%d-%m-%Y') }}</td>
<td>{{ p.ruangan.nama }}</td>
<td>{{ p.barang.nama }}</td>

```

```
  | | | | |
```

Kode tersebut menampilkan data peminjaman dalam bentuk tabel yang mencakup tanggal, nama ruangan, nama barang, status, dan surat disposisi jika tersedia. Jika status peminjaman disetujui dan surat telah diunggah, pengguna dapat melihat dokumen tersebut melalui tautan. Jika tidak, akan ditampilkan tanda strip. Bila tidak ada data peminjaman, sistem akan menampilkan pesan “Belum ada peminjaman” dalam satu baris tabel.

4.2.4 Halaman Dashboard Admin

Halaman dashboard admin menyediakan tampilan terpusat bagi administrator untuk memantau dan mengelola seluruh aktivitas peminjaman dalam sistem. Halaman ini adalah tabel data peminjaman, yang menampilkan informasi lengkap seperti nama pengguna, ruangan, barang yang dipinjam, tanggal peminjaman, dan status pengajuan, Seperti yang ditunjukkan pada Gambar 4.5.

The screenshot shows the Admin Dashboard with a sidebar menu on the left containing 'ADMINISTRATOR', 'Dashboard', 'Kelola Pengguna', 'Kelola Ruangan', and 'Log out'. The main content area is titled 'Halaman Admin' and 'Data Peminjaman'. It features a table with the following data:

User	Ruangan	Barang	Tanggal	Status
hmatelkourjaya	Ruang Kelas 2A	Proyektor	27-06-2025	disetujui
hmbunjaya	Ruang Kelas 2B	Proyektor	27-06-2025	disetujui
hmbunjaya	Ruang Kelas 2E	Proyektor	26-06-2025	pending
bemfili	Ruang Kelas 2B	Sofa	28-06-2025	pending
hmbunjaya	Ruang Riset Terpadu	Proyektor	30-06-2025	pending
bemfili	Laboratorium 3A	Sofa	08-07-2025	disetujui

Below the table is a 'Download Laporan Peminjaman' button. Underneath is a 'Cek Status Peminjaman (Kalender)' section showing a calendar for 'Juni 2025' with days of the week (Min, Sen, Sel, Rab, Kam, Jum, Sab) and dates 1 through 6.

Gambar 4. 5 Tampilan Halaman Dashboard Admin

```

if session.get('role') != 'admin':
    return redirect('/')
if request.method == 'POST':
    if 'nama_ruangan' in request.form:
        r = Ruangan(nama=request.form['nama_ruangan'],
        kapasitas=request.form['kapasitas_ruangan'])
        db.session.add(r)
    elif 'nama_barang' in request.form:
        b = Barang(nama=request.form['nama_barang'],
        deskripsi=request.form['deskripsi_barang'])
        db.session.add(b)
    db.session.commit()
    return redirect('/admin')

data = Peminjaman.query.filter_by(user_id=session['user_id']).order_by(
Peminjaman.tanggal_pinjam.desc()).all()
peminjaman_data = Peminjaman.query.all()

events = []
for p in peminjaman_data:
    print(p.tanggal_pinjam, p.ruangan, p.barang)
    events.append({
        "title": f"{p.ruangan.nama} - {p.barang.nama}",
        "start": p.tanggal_pinjam.strftime('%Y-%m-%d'),
        "color": "#3788d8" if p.status == "disetujui" else
"#ff9f89"
    })

```

Kode ini berfungsi untuk mengatur akses dan pengelolaan data oleh admin. Pertama, sistem memverifikasi bahwa hanya pengguna dengan peran admin yang dapat mengakses halaman ini. Jika menerima permintaan POST, sistem memeriksa apakah data yang dikirim berupa ruangan atau barang, lalu menyimpannya ke

database. Setelah itu, sistem mengambil data peminjaman dari *database* dan membentuk daftar event untuk ditampilkan di kalender, dengan warna penanda status peminjaman (biru untuk disetujui, oranye untuk lainnya).

4.2.5 Halaman Kelola Pengguna

Halaman kelola pengguna dirancang untuk memberikan akses kepada admin dalam mengelola seluruh aktivitas peminjaman yang dilakukan oleh pengguna. Fitur utama pada halaman ini adalah tabel peminjaman lengkap, yang menampilkan nama peminjam, peran pengguna, ruangan dan barang yang diajukan, tanggal peminjaman, status permohonan, serta tautan surat disposisi seperti pada Gambar 4.6.

Name Peminjam	Role	Barang	Ruangan	Tanggal	Status	Surat Disposisi	Edit
benfiti	mahasiswa	Sofa	Laboratorium 3A	88-07-2025	Disetujui	Lihat Card	Hapus Edit
hmilanjaya	mahasiswa	Proyektor	Ruang Riset	30-06-2025	Pending	Pilih File Upload	Hapus Edit
benfiti	mahasiswa	Sofa	Ruang Kelas 2B	28-06-2025	Pending	Pilih File Upload	Hapus Edit
hmilanjaya	mahasiswa	Proyektor	Ruang Kelas 2A	27-06-2025	Disetujui	Lihat Card	Hapus Edit
hmilanjaya	mahasiswa	Proyektor	Ruang Kelas 2B	27-06-2025	Disetujui	Lihat Card	Hapus Edit

Gambar 4. 6 Tampilan Halaman Kelola Pengguna

```

if session.get('role') != 'admin':
    return redirect('/')
if request.method == 'POST':
    if 'nama_ruangan' in request.form:
        r = Ruangan(nama=request.form['nama_ruangan'],
                    kapasitas=request.form['kapasitas_ruangan'])
        db.session.add(r)
    ini juga buatn penjelasan tentang kode yang akan dijadikan
    kalimat dan buatn dalam bahasa akademis dan plagiarism dibawah
    25%
    elif 'nama_barang' in request.form:
        b = Barang(nama=request.form['nama_barang'],
                    deskripsi=request.form['deskripsi_barang'])
        db.session.add(b)
        db.session.commit()
        return redirect('/admin')

```

```

data = Peminjaman.query.filter_by(user_id=session['user_id']).order_by(
Peminjaman.tanggal_pinjam.desc()).all()
peminjaman_data = Peminjaman.query.all()

events = []
for p in peminjaman_data:
    print(p.tanggal_pinjam, p.ruangan, p.barang)
    events.append({
        "title": f"{p.ruangan.nama} - {p.barang.nama}",
        "start": p.tanggal_pinjam.strftime('%Y-%m-%d'),
        "color": "#3788d8" if p.status == "disetujui" else
"#ff9f89"
    })

```

Kode ini menampilkan tabel data peminjaman untuk admin, mencakup informasi pengguna, barang, ruangan, dan tanggal peminjaman. Admin dapat langsung mengubah status peminjaman melalui dropdown, mengunggah atau mengganti surat disposisi, serta melakukan aksi hapus dan edit data. Semua fitur dirancang untuk memudahkan pengelolaan data secara interaktif dan efisien.

4.2.6 Halaman Kelola Ruangan

Halaman ini terdiri atas dua fitur utama, yaitu form tambah ruangan dan form tambah barang, yang masing-masing digunakan untuk menambahkan data baru ke dalam sistem. Selain itu, terdapat dua daftar data yang menampilkan daftar ruangan dan daftar barang beserta informasi detail seperti nama, kapasitas, dan deskripsi barang. Seluruh fitur ini ditujukan pada Gambar 4.7, yang bertujuan memudahkan admin dalam mengelola aset yang dapat dipinjam pengguna.

The screenshot shows a web application interface for an administrator. On the left is a teal sidebar with the title 'ADMINISTRATOR' and menu items: 'Dashboard', 'Kelola Pengguna', 'Kelola Ruangan', and 'Log out'. The main content area has a white background. It contains two form sections: 'Tambah Ruangan' and 'Tambah Barang'. Each form has input fields for 'Nama' and 'Kapasitas' (or 'Deskripsi' for barang), followed by a blue 'Tambah' button. Below these forms is a table titled 'Daftar Ruangan' with five rows, each showing a room name like 'Ruang Kelas 2A'.

Gambar 4. 7 Tampilan Halaman Kelola Ruangan

```

if request.method == 'POST':
    if 'nama_ruangan' in request.form and 'kapasitas_ruangan'
in request.form:
        ruangan_baru = Ruangan(
            nama=request.form['nama_ruangan'],
            kapasitas=request.form['kapasitas_ruangan']
        )
        db.session.add(ruangan_baru)
        db.session.commit()
        flash('Ruangan berhasil ditambahkan!', 'success')

    elif 'nama_barang' in request.form and 'deskripsi_barang'
in request.form:
        barang_baru = Barang(
            nama=request.form['nama_barang'],
            deskripsi=request.form['deskripsi_barang']
        )
        db.session.add(barang_baru)
        db.session.commit()
        flash('Barang berhasil ditambahkan!', 'success')

    return redirect('/admin_barang')

ruangan_list = Ruangan.query.all()
barang_list = Barang.query.all()
return render_template('admin_barang.html',
ruangan=ruangan_list, barang=barang_list)

```

Kode ini digunakan untuk mengelola proses penambahan data ruangan dan barang melalui metode POST. Jika form yang dikirim berisi data nama dan kapasitas ruangan, maka sistem akan membuat entitas Ruangan baru dan menyimpannya ke dalam basis data. Begitu pula, jika data yang dikirim berupa

nama dan deskripsi barang, maka objek Barang akan dibuat dan disimpan. Setelah proses penyimpanan berhasil, sistem memberikan notifikasi sukses kepada pengguna dan mengarahkan kembali ke halaman pengelolaan.

Selain itu, saat halaman diakses dengan metode GET, sistem akan mengambil seluruh data ruangan dan barang dari *database*, lalu menampilkannya melalui template `admin_barang.html`. Hal ini memungkinkan admin untuk melihat dan mengelola daftar ruangan serta barang yang tersedia.

4.3 IMPLEMENTASI BASIS DATA

Dalam pengembangan sistem informasi peminjaman ruangan dan inventaris FTTI UNJAYA, digunakan MySQL sebagai sistem manajemen basis data yang berperan sebagai penghubung antara kode program *backend* (Flask) dan antarmuka pengguna. MySQL dipilih karena mendukung berbagai jenis tipe data seperti VARCHAR, TEXT, INT, dan DATE yang dibutuhkan dalam penyimpanan data dinamis.

Basis data sistem ini terdiri dari empat tabel utama, yaitu: user, ruangan, barang, dan peminjaman. Masing-masing tabel berfungsi menyimpan informasi penting seperti data pengguna, daftar fasilitas yang dapat dipinjam, serta riwayat peminjaman. Setiap tabel dihubungkan melalui relasi foreign key untuk menjaga integritas dan keterkaitan data antar entitas.

Struktur lengkap tabel beserta relasinya telah dijelaskan pada Bab 3. Gambar 4.8 berikut menampilkan daftar tabel utama yang digunakan dalam sistem.

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
☐ barang	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	7	InnoDB	utf8mb4_0900_ai_ci	16.0 KB	-
☐ peminjaman	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	3	InnoDB	utf8mb4_0900_ai_ci	64.0 KB	-
☐ ruangan	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	18	InnoDB	utf8mb4_0900_ai_ci	16.0 KB	-
☐ user	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	10	InnoDB	utf8mb4_0900_ai_ci	32.0 KB	-
4 tabel	Jumlah	38	InnoDB	utf8mb4_0900_ai_ci	128.0 KB	0 B

Gambar 4. 8 Struktur Tabel Basis Data

Berdasarkan daftar tabel pada Gambar 4.8 tersebut, berikut dilampirkan salah satu struktur tabel peminjaman. Struktur tabel peminjaman ini berfungsi untuk

menampung data transaksi peminjaman ruangan dan inventaris oleh pengguna, yang ditunjukkan pada Gambar 4.9.

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	user_id	int		Ya		NULL			Ubah Hapus Lainnya
3	ruangan_id	int		Ya		NULL			Ubah Hapus Lainnya
4	barang_id	int		Ya		NULL			Ubah Hapus Lainnya
5	tanggal_pinjam	date		Ya		NULL			Ubah Hapus Lainnya
6	status	varchar(50)	utf8mb4_0900_ai_ci	Ya		NULL			Ubah Hapus Lainnya
7	surat_disposisi	varchar(255)	utf8mb4_0900_ai_ci	Ya		NULL			Ubah Hapus Lainnya
8	tanggal_kembali	date		Ya		NULL			Ubah Hapus Lainnya

Gambar 4. 9 Struktur Tabel Peminjaman

4.4 PENGUJIAN APLIKASI

Pengujian sistem dilakukan untuk memastikan bahwa seluruh fitur dalam aplikasi berjalan sesuai dengan kebutuhan dan dapat digunakan secara fungsional oleh pengguna. Dalam tahap ini, metode Blackbox Testing digunakan untuk menguji apakah fitur memberikan keluaran yang sesuai dengan input yang diberikan, tanpa memperhatikan proses internal atau logika program. Selain itu, dilakukan juga User Acceptance Test (UAT) untuk menilai kepuasan dan kemudahan penggunaan dari sudut pandang pengguna akhir.

4.4.1 Blackbox Testing

Blackbox Testing merupakan metode pengujian untuk mengevaluasi fungsionalitas sistem dengan cara memeriksa kesesuaian keluaran berdasarkan masukan yang diberikan, tanpa memperhatikan struktur internal atau kode program aplikasi. Pada pengujian ini, pengguna hanya memfokuskan pada fungsi-fungsi yang tersedia pada sistem, memastikan setiap fitur dapat berjalan sesuai dengan kebutuhan dan spesifikasi yang telah dirancang.

Pengujian dilakukan oleh satu orang penguji teknis (*developer*) dan satu orang admin, dengan tujuan untuk memeriksa apakah setiap alur kerja, seperti login, pengajuan peminjaman, pengecekan status, serta pengelolaan data ruangan dan inventaris, dapat berjalan dengan baik pada kondisi input yang berbeda. Setiap

masukan diuji untuk melihat bagaimana sistem memberikan respon yang sesuai, baik pada masukan valid maupun masukan yang tidak valid.

Hasil detail pengujian dapat dilihat pada tabel *Blackbox Testing* pada bagian di bawah. Secara keseluruhan, dari hasil pengujian yang telah dilakukan, sistem dinyatakan berfungsi dengan baik dan siap digunakan dalam lingkungan operasional Fakultas Teknik dan Teknologi Informasi Universitas Jenderal Achmad Yani Yogyakarta. Hal ini menunjukkan bahwa sistem telah memenuhi kebutuhan fungsional yang dirancang, serta dapat membantu meningkatkan efisiensi dan akurasi dalam proses peminjaman inventaris dan ruangan.

Tabel 4. 1 Tabel Pengujian *Blackbox*

Fitur yang Diuji	Developer	Admin	Hasil Uji
Login User	✓	✓	Berhasil
Form Pengajuan Peminjaman	✓	✓	Berhasil
Validasi Rules Based	✓	✓	Berhasil
Fullcalender intraktif	✓	✓	Berhasil
Cek Status	✓	✓	Berhasil
Surat disposisi	✓	✓	Berhasil
Login Admin	✓	✓	Berhasil
Kelola pengguna	✓	✓	Berhasil
Tambah Ruangan dan barang	✓	✓	Berhasil
Download data peminjam	✓	✓	Berhasil

4.4.2 User Acceptance Test (UAT)

User acceptance testing pada aplikasi akan dilakukan dengan mengajukan beberapa pernyataan terkait kelayakan fungsionalitas aplikasi menggunakan metode *USE Questionnaire : Usefulness, Satisfaction, Ease of Use, Ease of Learn* kepada 5 responden, yaitu 5 mahasiswa pengurus Ormawa. Rumus dalam menghitung persentase jawaban responden sesuai dengan skor yang akan dijabarkan seperti dibawah ini:

$$Presentase = \frac{Skor\ Total}{Skor\ Ideal} \times 100\%$$

Tabel 4. 2 Tabel Usefulness

Responden	P1	P2	P3	P4	Total
Naufal El Qolbie	5	5	3	3	16
Syahrul Himawan	5	5	5	5	20
Fadhil Sani	5	5	5	5	20
Ilham Nur	5	5	5	5	20
Dafa firzatullah	4	5	3	4	16
Total					92
Skor Ideal					100
Presentase					92%

Keterangan dari Tabel 4.2 tersebut menjelaskannya bahwa hasil dari perhitungan Usefulness ialah 92% dengan perhitungan sebagai berikut.

$$P = \frac{92}{100} \times 100\% = 92\%$$

Tabel 4. 3 Tabel Satisfaction

Responden	P1	P2	P3	P4	Total
Naufal El Qolbie	3	3	3	4	13
Syahrul Himawan	5	5	5	5	20
Fadhil Sani	5	5	5	5	20
Ilham Nur	5	5	5	5	20
Dafa firzatullah	3	4	5	4	16
Total					89

Skor Ideal	100
Presentase	89%

Keterangan dari Tabel 4.3 tersebut menjelasnya bahwa hasil dari perhitungan Satication ialah 89% dengan perhitungan sebagai berikut.

$$P = \frac{89}{100} \times 100\% = 89\%$$

Tabel 4. 4 Tabel *Ease of Use*

Responden	P1	P2	P3	P4	Total
Naufal El Qolbie	3	4	3	5	15
Syahrul Himawan	5	5	5	5	20
Fadhil Sani	5	5	5	5	20
Ilham Nur	5	5	5	5	20
Dafa firzatullah	3	3	4	4	14
Total					89
Skor Ideal					100
Presentase					89%

Keterangan dari Tabel 4.4 tersebut menjelasnya bahwa hasil dari perhitungan *Ease of Use* ialah 89% dengan perhitungan sebagai berikut.

$$P = \frac{89}{100} \times 100\% = 89\%$$

Tabel 4. 5 Tabel *Ease of Learn*

Responden	P1	P2	P3	P4	Total
Naufal El Qolbie	4	4	5	3	16
Syahrul Himawan	5	5	5	5	20
Fadhil Sani	5	5	5	5	20
Ilham Nur	5	5	5	5	20
Dafa firzatullah	5	5	5	4	19
Total					95
Skor Ideal					100
Presentase					95%

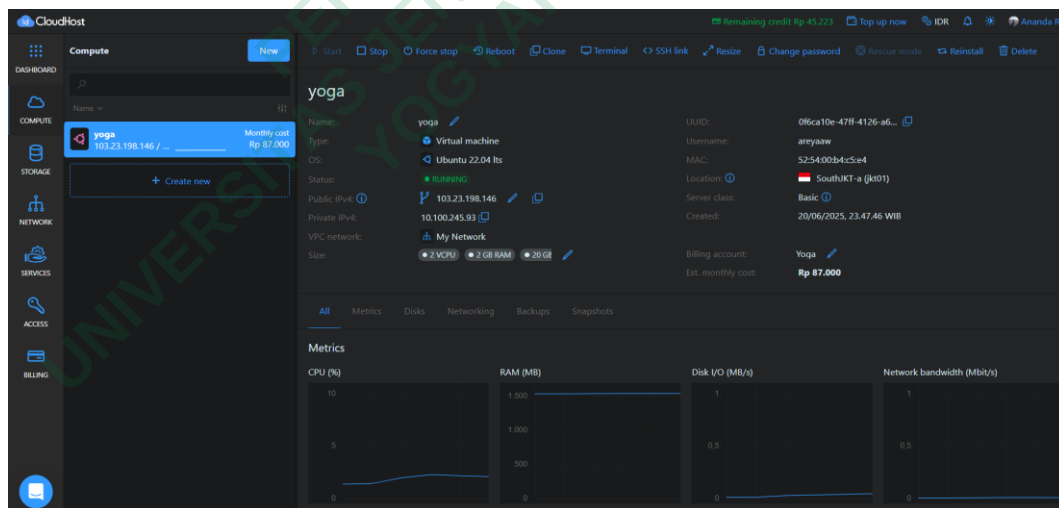
Keterangan dari Tabel 4.5 tersebut menjelasnya bahwa hasil dari perhitungan *Ease of Learn* ialah 95% dengan perhitungan sebagai berikut.

$$P = \frac{95}{100} \times 100\% = 95\%$$

4.5 DEPLOYMENT

Tahap *deployment* merupakan proses penting dalam pengembangan perangkat lunak, di mana aplikasi yang telah selesai dibangun dipindahkan dari lingkungan pengembangan lokal ke server agar dapat diakses secara daring. Dalam penelitian ini, sistem informasi peminjaman ruangan dan inventaris FTTI UNJAYA dikembangkan menggunakan bahasa pemrograman Python dengan *framework* Flask, serta menggunakan MySQL sebagai sistem manajemen basis data.

Proses implementasi sistem ke server dilakukan pada mesin berbasis Ubuntu Server yang diakses melalui protokol SSH. Untuk menjalankan aplikasi secara optimal dan dapat diakses melalui jaringan publik, digunakan kombinasi Gunicorn sebagai WSGI server dan Nginx sebagai reverse proxy.



Gambar 4. 10 Tampilan VPS Menggunakan IDCloudHost.

Setelah proses konfigurasi selesai, aplikasi berhasil dijalankan dan dapat diakses menggunakan alamat domain siperu.my.id. Seperti pada Gambar 4.11



Gambar 4. 11 Hasil Dari *Deployment*

4.6 PEMBAHASAN

Sistem informasi peminjaman ruangan dan inventaris FTI UNJAYA dikembangkan sebagai solusi terhadap permasalahan yang muncul akibat proses peminjaman yang sebelumnya dilakukan secara manual. Ketidakteraturan dalam pencatatan, keterlambatan konfirmasi, dan kurangnya dokumentasi menjadi hambatan utama yang dihadapi pengguna, baik mahasiswa, dosen, maupun pihak pengelola.

Dengan diimplementasikannya sistem ini, seluruh alur mulai dari pengajuan peminjaman, pengecekan ketersediaan melalui kalender, hingga manajemen data ruangan dan barang dapat dilakukan secara digital dan terpusat. Sistem ini mendukung kemudahan akses, keteraturan proses, serta transparansi data.

Hasil pengujian menggunakan metode *Blackbox* menunjukkan bahwa seluruh fitur inti sistem, seperti autentikasi pengguna, form pengajuan, cek status, kelola pengguna, kelola ruangan, dan kelola disposisi berjalan sesuai dengan rancangan dan tidak ditemukan kesalahan fungsional yang berarti. Seluruh skenario pengujian menghasilkan output yang sesuai dengan input yang diberikan. Pengujian lebih lanjut melalui *User Acceptance Test* (UAT) juga menghasilkan respons positif

dari pengguna. Dengan nilai Usefulness mencapai 92%, serta aspek Satisfaction, Ease of Use, dan Ease of Learn masing-masing di atas 90%, dapat disimpulkan bahwa sistem diterima dengan baik oleh pengguna dan mudah digunakan tanpa memerlukan pelatihan khusus.

Selain dari sisi pengembangan dan pengujian, sistem ini juga telah berhasil dideploy ke server VPS berbasis Linux dan diakses melalui domain publik (<http://siperu.my.id>). Proses *deployment* dilakukan menggunakan Gunicorn sebagai application server dan Nginx sebagai reverse proxy, serta telah diuji berjalan stabil dalam lingkungan produksi. Hal ini memperkuat bahwa sistem tidak hanya layak secara teknis dan fungsional, tetapi juga siap diimplementasikan secara operasional.

Dengan demikian, sistem ini telah berhasil menjawab permasalahan awal yang melatarbelakangi penelitian dan memberikan solusi digital yang efektif, efisien, serta dapat terus dikembangkan untuk mendukung manajemen fasilitas kampus secara berkelanjutan.