

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Bagian ini menjelaskan hasil dari pengembangan sistem informasi penyewaan alat musik yang telah dirancang dan diimplementasikan. Sistem dibangun berdasarkan kebutuhan pengguna untuk mempermudah proses penyewaan alat musik secara online, khususnya dalam hal efisiensi waktu, kemudahan akses, serta penyampaian informasi yang lebih terstruktur.

Hasil penelitian ini mencakup implementasi dari antarmuka pengguna, alur sistem, serta fitur utama yang telah dirancang pada bab sebelumnya. Selain itu, bagian ini juga menguraikan struktur sistem, arsitektur database, serta alur kerja sistem yang telah diuji.

Implementasi dilakukan menggunakan bahasa pemrograman berbasis web, yaitu HTML, CSS, dan JavaScript untuk tampilan antarmuka, serta Django untuk sisi backend. Sistem ini juga terintegrasi dengan WhatsApp API untuk proses konfirmasi penyewaan secara langsung kepada penyedia jasa.

4.2 IMPLEMENTASI DESAIN ANTARMUKA

Implementasi sistem merupakan tahap realisasi dari perancangan yang telah dilakukan pada bab sebelumnya. Pada tahap ini, seluruh komponen sistem mulai dari antarmuka pengguna, alur proses, hingga integrasi fitur telah diubah menjadi kode program yang dapat dijalankan secara nyata dalam bentuk aplikasi berbasis web.

Sistem penyewaan alat musik ini dibangun menggunakan framework Django sebagai backend, serta HTML, CSS, dan JavaScript untuk bagian frontend. Selain itu, sistem juga mengintegrasikan fitur WhatsApp untuk mempermudah proses komunikasi antara pengguna dan penyedia jasa sewa.

4.2.1 Halaman Beranda Pengguna

Halaman beranda merupakan tampilan pertama yang dilihat oleh pengguna saat mengakses aplikasi Paranada Musik. Halaman ini dirancang untuk memberikan kesan awal yang informatif dan menarik, dengan menampilkan berbagai elemen penting seperti menu navigasi, ajakan untuk menyewa, keunggulan layanan, hingga testimoni pelanggan secara langsung.



Gambar 4.1 Halaman Beranda

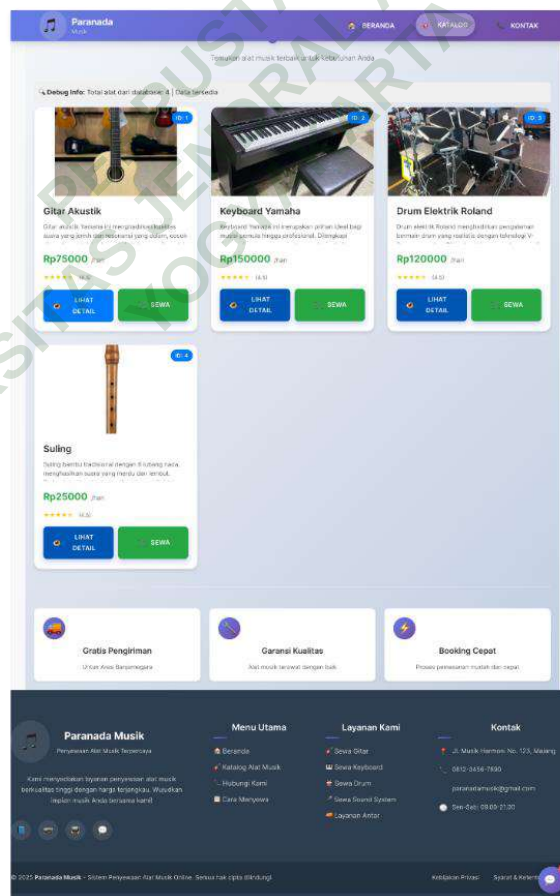
Berikut merupakan cuplikan kode yang digunakan untuk membangun halaman beranda pelanggan:

```
# views.py
def beranda(request):
    return render(request, 'penyewaan/beranda.html')
```

Kode ini adalah bagian dari *views.py* di *framework Django*, dan fungsinya untuk menangani permintaan (request) yang masuk ke halaman beranda dari aplikasi web.

4.2.2 Halaman Katalog

Halaman katalog menampilkan daftar alat musik yang tersedia untuk disewa oleh pengguna. Tampilan halaman ini disusun secara rapi dan interaktif agar memudahkan pengguna dalam menelusuri berbagai jenis alat musik yang ditawarkan. Setiap item dalam katalog dilengkapi dengan gambar, nama alat, dan tombol untuk melihat detail, sehingga pengguna dapat memperoleh informasi awal sebelum melanjutkan ke proses penyewaan.



Gambar 4.2 Halaman Katalog

Cuplikan kode yang menampilkan halaman ini sebagai berikut:

```
# views.py
def katalog(request, alat_id=None):
    if alat_id:
        alat = get_object_or_404(AlatMusik, id=alat_id)
        return render(request, 'penyewaan/katalog.html',
            {'alat': alat})
    alat_list = AlatMusik.objects.all()
    return render(request, 'penyewaan/katalog.html',
        {'alat_list': alat_list})
```

Fungsi katalog digunakan untuk menampilkan data alat musik. Jika parameter `alat_id` diberikan, maka akan ditampilkan detail alat musik berdasarkan ID tersebut. Jika tidak, maka seluruh daftar alat musik akan ditampilkan.

4.2.3 Halaman Detail Alat

Halaman detail alat digunakan untuk menampilkan informasi lengkap mengenai alat musik yang dipilih oleh pengguna dari katalog. Pada halaman ini, pengguna dapat melihat nama alat, gambar, deskripsi, serta harga sewa yang ditawarkan.



Gambar 4.3 Halaman Detail Alat

Cuplikan kode yang menampilkan halaman ini sebagai berikut:

```
# views.py
def detail_alat(request, alat_id):
    alat = get_object_or_404(AlatMusik, id=alat_id)
    return render(request, 'penyewaan/katalog.html',
        {'alat': alat})
```

Fungsi *detail_alat* digunakan untuk menampilkan halaman detail dari satu alat musik berdasarkan ID yang dipilih. Data alat musik diambil dari model *AlatMusik* menggunakan fungsi *get_object_or_404*, dan kemudian ditampilkan melalui template *catalog.html* dengan konteks berupa objek alat.

4.2.4 Halaman Form Sewa

Dalam halaman ini, pengguna diminta untuk mengisi informasi seperti nama, tanggal awal dan akhir sewa.

Gambar 4.4 Halaman Form Sewa

Cuplikan kode pada *views.py* yang digunakan untuk menampilkan halaman tersebut ditunjukkan sebagai berikut:

```
#views.py
def sewa_alat(request, alat_id):
    alat = get_object_or_404(AlatMusik, id=alat_id)
    if request.method == 'POST':
        form = SewaForm(request.POST)
        if form.is_valid():
            sewa = form.save(commit=False)
            sewa.alat = alat

            if sewa.tanggal_awal < (timezone.now().date() +
timedelta(days=2)):
                messages.error(request, 'Tanggal pemesanan
terlalu
mepet. Minimal H-2 dari hari ini.')
```

```

        return render(request,
            'penyewaan/sewa_alat.html',
            {'form': form, 'alat': alat})

        if sewa.tanggal_awal and sewa.tanggal_akhir:
            durasi_hari = (sewa.tanggal_akhir
                sewa.tanggal_awal).days
            if durasi_hari >= 0:
                sewa.tanggal_kembali = sewa.tanggal_akhir
                sewa.total_harga = durasi_hari * alat.harga

            sewa.save()
            return redirect('pembayaran', sewa_id=sewa.id)
        else:
            initial_data = {
                'tanggal_awal': timezone.now().date(),
                'tanggal_akhir': timezone.now().date() +
                    timedelta(days=1),
            }
            form = SewaForm(initial=initial_data)

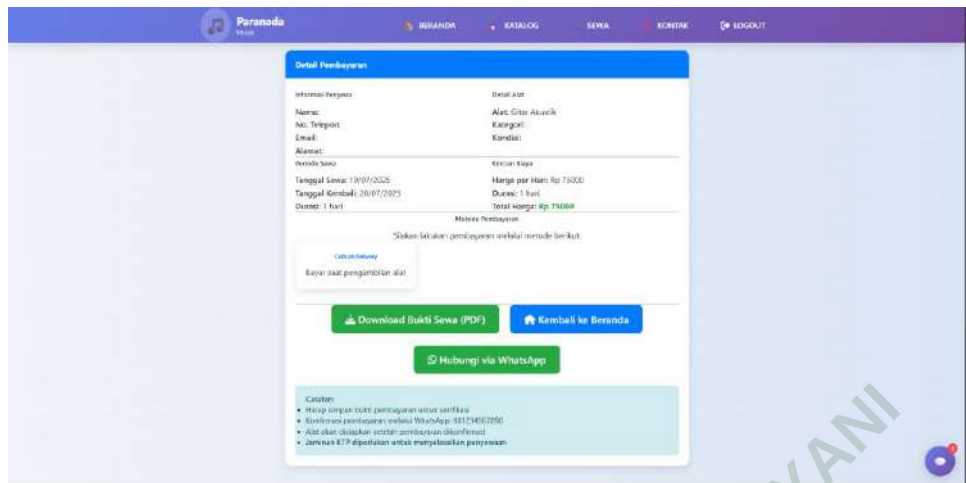
        return render(request, 'penyewaan/sewa_alat.html', {'form':
            form, 'alat': alat})

```

Fungsi `sewa_alat` digunakan untuk menangani proses penyewaan alat musik berdasarkan ID alat yang dipilih. Jika metode permintaan adalah *POST*, maka data yang dikirimkan melalui form akan divalidasi. Jika valid, sistem akan memeriksa apakah tanggal penyewaan minimal H-2 dari tanggal hari ini. Jika tidak memenuhi, akan muncul pesan peringatan. Selanjutnya, sistem menghitung durasi penyewaan dan total harga berdasarkan harga alat per hari. Setelah data disimpan, pengguna akan diarahkan ke halaman pembayaran. Jika permintaan bukan *POST*, maka form penyewaan akan ditampilkan dengan tanggal awal dan akhir yang sudah diisi otomatis.

4.2.5 Detail Pembayaran

Tampilan halaman pembayaran dirancang untuk menyajikan informasi penyewaan secara jelas dan memudahkan pengguna dalam menyelesaikan proses transaksi. Di dalamnya ditampilkan data penyewa, detail alat musik yang disewa, perhitungan biaya, serta beberapa opsi metode pembayaran. Selain itu, tersedia tombol untuk mengunduh bukti sewa dalam format PDF, kembali ke halaman utama, dan menghubungi admin melalui WhatsApp guna melakukan konfirmasi pembayaran.



Gambar 4.5 Halaman Detail Pembayaran

Cuplikan kode pada *views.py* yang digunakan untuk menampilkan halaman tersebut ditunjukkan sebagai berikut.

```
def pembayaran(request, sewa_id):
    from .models import Sewa

    sewa = get_object_or_404(Sewa, id=sewa_id)

    # Hitung jumlah hari sewa
    jumlah_hari = (sewa.tanggal_akhir - sewa.tanggal_awal).days or 1 # Minimum 1 hari

    # Hitung total harga berdasarkan harga alat
    harga_per_hari = int(sewa.alat.harga)
    total_harga = jumlah_hari * harga_per_hari

    return render(request, 'penyewaan/pembayaran.html', {
        'sewa': sewa,
        'jumlah_hari': jumlah_hari,
        'total_harga': total_harga,
    })
```

Fungsi pembayaran bertugas untuk menampilkan detail pembayaran dari suatu transaksi penyewaan. Fungsi ini mengambil data sewa berdasarkan ID, menghitung durasi hari dari tanggal sewa hingga tanggal pengembalian, lalu menghitung total biaya sewa dengan mengalikan durasi hari dengan harga per hari dari alat musik yang disewa. Selanjutnya, data tersebut dikirimkan ke template *pembayaran.html* untuk ditampilkan kepada pengguna.

4.2.6 Bukti Sewa Pengguna

Dokumen bukti penyewaan alat musik adalah hasil cetak otomatis dari sistem setelah proses pemesanan berhasil dilakukan. Di dalamnya tercantum detail lengkap transaksi, seperti nomor ID sewa, identitas penyewa, nama alat musik yang disewa, tanggal sewa dan pengembalian, durasi penggunaan, tarif sewa per hari, total biaya, serta status pembayaran. Selain itu, dokumen ini juga menyertakan catatan penting mengenai ketentuan pengembalian dan tanggung jawab penyewa, serta informasi waktu pembuatan dokumen sebagai bukti valid atas transaksi yang dilakukan.

BUKTI PENYEWAAN ALAT MUSIK

DETAIL PENYEWAAN

ID Sewa	#139
Nama Penyewa	karisma
Alat yang Disewa	Gitar Akustik
Tanggal Sewa	30/07/2025
Tanggal Kembali	31/07/25
Durasi	1 hari
Harga per Hari	Rp 75.000
Total Harga	Rp 75.000
No. KTP Jaminan	1234567890123456
Nama KTP	Karisma Indah Rumanti
Status Pembayaran	Sudah Dibayar

CATATAN PENTING:

- Harap membawa alat musik dalam kondisi baik saat pengembalian
- Keterlambatan pengembalian akan dikenakan denda sesuai ketentuan
- Kerusakan alat menjadi tanggung jawab penyewa
- KTP akan dikembalikan setelah alat dikembalikan
- Simpan bukti ini sebagai tanda bukti penyewaan

Gambar 4.6 Bukti Sewa Pengguna

Cuplikan kode *views.py* yang menampilkan halaman tersebut disajikan sebagai berikut:

```
def pembayaran(request, sewa_id):
    from .models import Sewa

    sewa = get_object_or_404(Sewa, id=sewa_id)

    # Hitung jumlah hari sewa
    jumlah_hari = (sewa.tanggal_akhir - sewa.tanggal_awal).days
    or 1 # Minimum 1 hari

    # Hitung total harga berdasarkan harga alat
    harga_per_hari = int(sewa.alat.harga)
    total_harga = jumlah_hari * harga_per_hari

    return render(request, 'penyewaan/pembayaran.html', {
        'sewa': sewa,
        'jumlah_hari': jumlah_hari,
        'total_harga': total_harga,
    })
```

Fungsi `pembayaran()` bertugas untuk menampilkan halaman pembayaran penyewaan alat musik berdasarkan data sewa yang dipilih. Fungsi ini menerima parameter `request` dan `sewa_id`, di mana `sewa_id` digunakan untuk mengambil data penyewaan tertentu dari database melalui model `Sewa`.

Langkah pertama dalam fungsi ini adalah mengambil objek `Sewa` menggunakan `get_object_or_404()`. Jika data sewa dengan ID yang dimaksud tidak ditemukan, maka halaman error 404 akan ditampilkan.

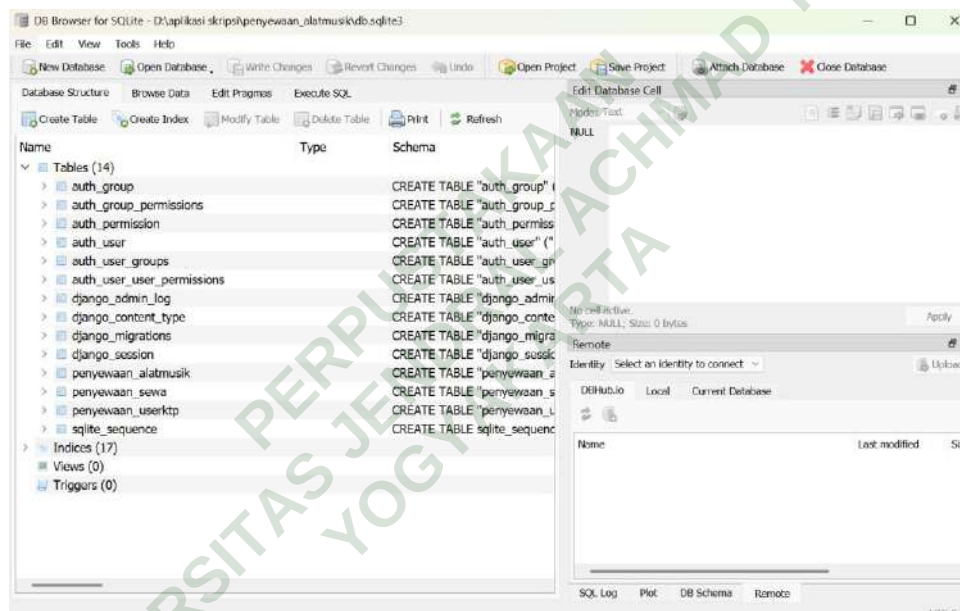
Setelah data sewa ditemukan, sistem menghitung jumlah hari penyewaan dengan cara mengurangkan `tanggal_akhir` dengan `tanggal_awal`. Jika hasilnya 0 atau negatif, maka jumlah hari dianggap **minimal satu hari** sebagai kebijakan default.

Kemudian, harga sewa per hari diambil dari harga alat musik yang disewa (`sewa.alat.harga`), dan dikonversi ke bentuk integer untuk memastikan tidak terjadi kesalahan saat perhitungan. Total harga sewa dihitung dengan mengalikan jumlah hari sewa dengan harga per hari.

Akhirnya, fungsi ini merender halaman `pembayaran.html` dan mengirimkan tiga data ke template tersebut:

4.3 BASIS DATA

Bab IV membahas tahap implementasi basis data untuk sistem informasi penyewaan alat musik yang telah dirancang sebelumnya. Pada tahap ini, *SQLite* digunakan sebagai sistem manajemen basis data yang ringan dan terintegrasi dengan *framework Django*. Proses pembuatan tabel baik tabel bawaan *Django* seperti *auth_user* dan *django_migrations*, maupun tabel khusus seperti *penyewaan_sewa* dan *penyewaan_alatmusik* akan dijelaskan secara detail. Selain itu, uraian juga mencakup bagaimana struktur basis data tersebut mendukung kebutuhan penyimpanan dan pengelolaan data dalam aplikasi.

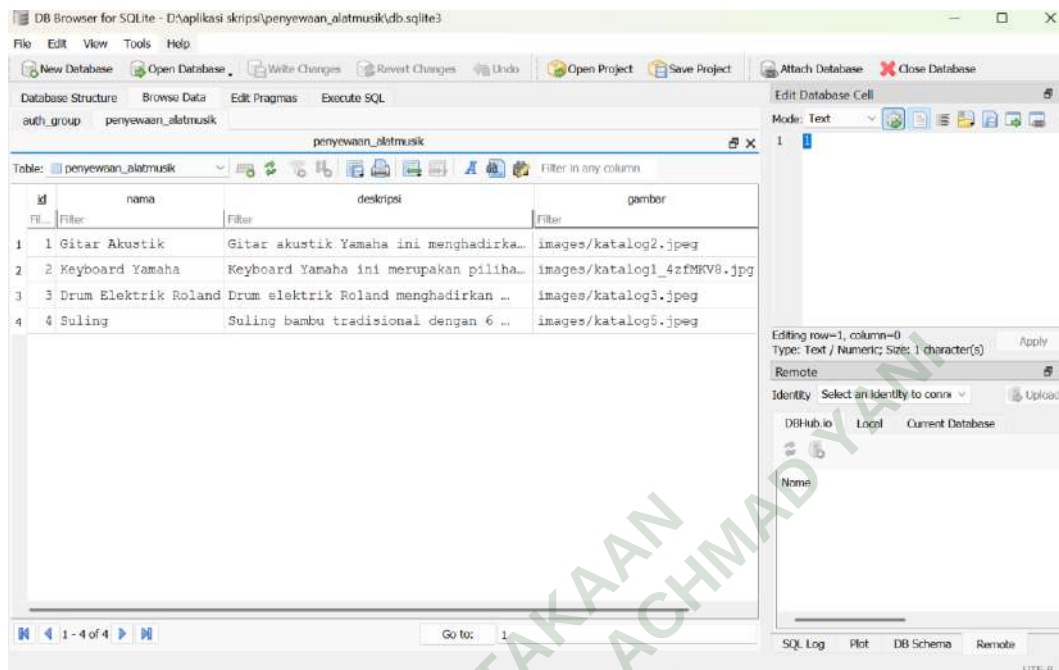


Gambar 4.7 Implementasi Struktur Basis Data

4.3.1 Tabel *penyewaan_alatmusik* (Admin)

Tabel ini menyimpan data mengenai alat musik yang tersedia untuk disewa. Setiap baris dalam tabel mewakili satu alat musik. Struktur kolom utamanya meliputi:

- ID unik dari alat musik (Primary Key)
- Nama Alat Musik
- Harga sewa per hari
- Deskripsi singkat mengenai alat musik
- Gambar alat musik



Gambar 4.8 Tabel penyewaan_alatmusik (admin)

4.3.2 Tabel penyewaan_sewa (pengguna)

Tabel *penyewaan_sewa* berfungsi untuk mencatat seluruh aktivitas penyewaan musik oleh alat pengguna. Data yang tersimpan meliputi informasi penyewa, jenis alat musik yang disewa, durasi penyewaan, metode pembayaran, serta waktu pencatatan transaksi oleh sistem.

Rincian masing-masing atribut pada tabel ini adalah sebagai berikut:

- ID: Merupakan kunci utama yang mengidentifikasi setiap entri penyewaan secara unik (primary key).
- Nama: Menyimpan nama individu yang melakukan penyewaan.
- dibuat_pada: Merekam waktu saat entri penyewaan dibuat, secara otomatis oleh sistem.
- Alat_id: Merupakan relasi ke tabel penyewaan_alatmusik yang menunjukkan alat musik yang disewa.
- Tanggal_awal: Tanggal dimulainya penyewaan.

- f. Metode_pembayaran: Menunjukkan jenis pembayaran yang dipilih oleh penyewa, seperti transfer bank, e-wallet, QRIS, atau pembayaran langsung saat pengambilan alat (COD).

Tabel ini memainkan peran penting dalam pengelolaan data penyewaan, memudahkan sistem dalam memverifikasi ketersediaan alat, menghitung total biaya sewa berdasarkan durasi, serta mencetak dokumen bukti penyewaan secara otomatis dalam format PDF.

	id	nama	dibuat_pada	alat_id	tanggal_akhir	tanggal_awal	metode_pembayaran
1	1	khabil	2025-06-18 16:15:01.315392	1	2025-06-11	2025-06-20	MOTOL
2	2	putra	2025-06-20 14:51:30.757226	2	2025-06-21	2025-06-20	MOTOL
3	3	pratama	2025-06-20 15:09:00.666669	2	2025-06-21	2025-06-20	MOTOL
4	4	khabil	2025-06-20 15:09:15.996760	1	2025-06-21	2025-06-20	MOTOL
5	5	khabil	2025-06-20 15:17:40.090557	2	2025-06-21	2025-06-20	MOTOL
6	6	khabil	2025-06-20 15:18:42.573206	2	2025-06-21	2025-06-20	MOTOL
7	7	khabil	2025-06-20 15:20:40.545448	2	2025-06-21	2025-06-20	MOTOL
8	8	khabil	2025-06-20 15:22:39.586311	2	2025-06-21	2025-06-20	MOTOL
9	9	putra	2025-06-20 16:30:13.244010	2	2025-06-21	2025-06-20	E-Wallet (Dans/OVO/GC
10	10	khabil	2025-06-20 16:36:39.345813	2	2025-06-21	2025-06-20	MOTOL
11	11	khabil	2025-06-20 16:44:10.115037	2	2025-06-21	2025-06-20	MOTOL
12	12	khabil	2025-06-21 18:28:19.994900	2	2025-06-22	2025-06-21	MOTOL
13	13	khabil	2025-06-22 14:04:45.988883	1	2025-06-23	2025-06-22	E-Wallet
14	14	khabil	2025-06-22 14:05:38.598530	2	2025-06-23	2025-06-22	MOTOL

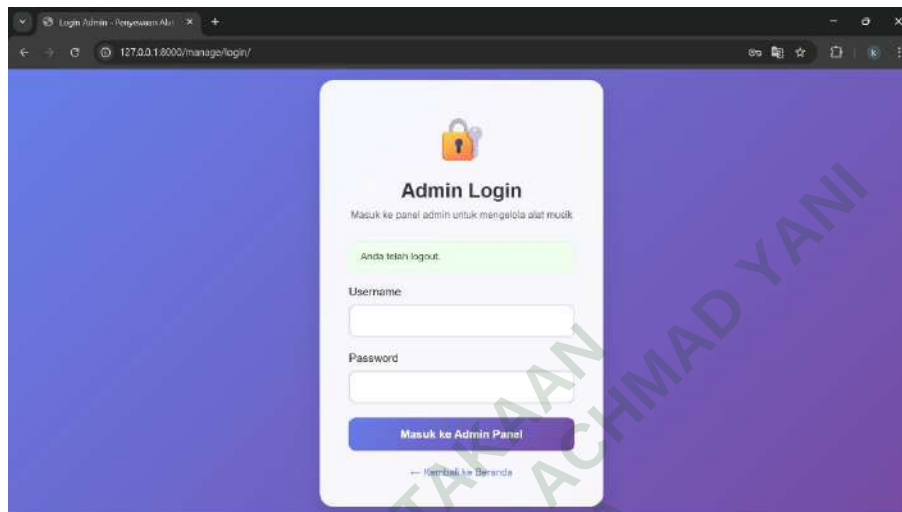
Gambar 4.9 Tabel penyewaan_sewa (pengguna)

4.4 FITUR-FITUR SISTEM

Sistem informasi penyewaan alat musik ini dikembangkan dengan menyertakan sejumlah fitur utama yang saling terintegrasi, guna mempermudah pengguna dalam melakukan penyewaan. Fitur-fitur tersebut mencakup tampilan katalog, formulir penyewaan, pilihan metode pembayaran, serta kemampuan mencetak bukti transaksi secara otomatis. Setiap fitur dirancang agar proses penyewaan menjadi lebih efisien, praktis, dan informatif.

4.4.1 Login Admin

Fitur login admin berfungsi sebagai akses utama bagi administrator untuk masuk ke dalam sistem. Melalui halaman ini, admin harus memasukkan username dan password yang valid agar dapat mengelola data alat musik yang tersedia.



Gambar 4.10 Login Admin

Cuplikan kode *views.py* untuk halaman Login Admin ditampilkan sebagai berikut.

```
def admin_login(request):
    if request.user.is_authenticated and request.user.is_superuser:
        return redirect('admin_alat')

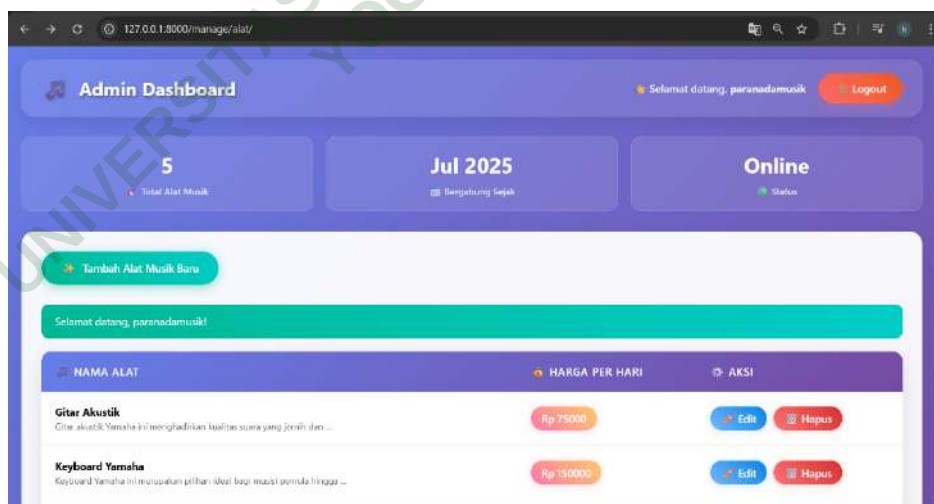
    if request.method == 'POST':
        form = AuthenticationForm(request, data=request.POST)
        if form.is_valid():
            username = form.cleaned_data.get('username')
            password = form.cleaned_data.get('password')
            user = authenticate(username=username, password=password)
            if user is not None and user.is_superuser:
                login(request, user)
                messages.success(request, f'Selamat datang, {username}!')
                return redirect('admin_alat')
            else:
                messages.error(request, 'Akses ditolak. Hanya admin yang dapat login.')
        else:
            messages.error(request, 'Username atau password tidak valid.')
        else:
            form = AuthenticationForm()
```

```
return render(request, 'penyewaan/admin_login.html',
{'form': form})
```

Fitur login admin berfungsi sebagai akses awal yang hanya bisa digunakan oleh pengguna dengan hak sebagai administrator. Pada halaman ini, admin perlu memasukkan *username* dan kata sandi yang valid. Setelah berhasil diverifikasi dan dikenali sebagai *superuser*, sistem akan secara otomatis mengarahkan admin ke panel pengelolaan. Mekanisme ini dirancang untuk menjaga keamanan data dan membatasi akses hanya kepada pihak yang berwenang.

4.4.2 Dashboard Admin

Dashboard Admin merupakan halaman pusat kendali bagi administrator untuk mengelola data alat musik yang tersedia dalam sistem penyewaan. Melalui antarmuka ini, admin dapat melihat jumlah total alat musik, waktu bergabungnya akun admin, serta status koneksi secara real-time. Fitur tambahan seperti tombol tambah alat musik baru, serta opsi edit dan hapus pada setiap entri alat juga disediakan guna mempermudah pengelolaan data secara langsung. Informasi disusun secara ringkas dan interaktif, sehingga memudahkan admin dalam melakukan kontrol sistem secara efisien.



Gambar 4.11 Dashboard Admin

Berikut adalah penjelasan dalam bentuk kalimat mengenai kode views untuk halaman Dashboard Admin:

```
@user_passes_test(is_admin, login_url='admin_login')
def admin_alat(request):
    alat_list = AlatMusik.objects.all()
    return render(request, 'penyewaan/admin_alat.html',
{'alat_list': alat_list})
```

Tampilan Dashboard Admin dalam aplikasi ini dikendalikan oleh fungsi `admin_alat` yang berada pada file `views.py`. Fungsi tersebut bertugas untuk mengambil seluruh data alat musik dari basis data dengan perintah `AlatMusik.objects.all()` dan meneruskannya ke halaman `admin_alat.html` agar dapat ditampilkan kepada admin dalam bentuk visual yang informatif.

Sebelum fungsi ini dijalankan, sistem akan memverifikasi bahwa pengguna adalah seorang admin menggunakan dekorator `@user_passes_test(is_admin, login_url='admin_login')`. Jika bukan admin, pengguna akan diarahkan ke halaman login. Dengan adanya verifikasi ini, halaman dashboard hanya dapat diakses oleh admin yang telah berhasil masuk ke sistem.

4.4.3 Halaman Edit Alat Musik

Fitur edit alat musik memberikan kemudahan bagi admin untuk memperbaharui detail alat musik yang telah terdaftar di sistem, mulai dari nama, keterangan, hingga harga dan gambar. Proses ini dijalankan melalui fungsi `edit_alat` di `views.py`, yang akan mengambil data alat berdasarkan ID-nya, lalu menampilkan *form* dengan data yang sudah ada sebelumnya. Jika admin melakukan perubahan dan menyimpan *form*, maka sistem akan memperbarui informasi tersebut ke dalam basis data.

Gambar 4.12 Halaman Edit Alat Musik

Berikut adalah kode views.py untuk Halaman Edit Alat Musik beserta penjelasannya:

```
@user_passes_test(is_admin, login_url='admin_login')
def edit_alat(request, alat_id):
    alat = get_object_or_404(AlatMusik, id=alat_id)
    if request.method == 'POST':
        form = AlatMusikForm(request.POST, request.FILES,
instance=alat)
        if form.is_valid():
            form.save()
            messages.success(request, 'Data alat musik
berhasil diperbarui.')
            return redirect('admin_alat')
        else:
            form = AlatMusikForm(instance=alat)
            return render(request, 'penyewaan/form_alat.html',
{'form': form, 'judul': 'Edit Alat Musik'})
```

Fungsi `edit_alat` pada sisi server bertugas menangani proses perubahan data alat musik yang dilakukan oleh admin. Sistem akan mencari data berdasarkan ID alat yang dipilih, kemudian menampilkan formulir yang telah diisi dengan data sebelumnya. Jika admin mengubah data dan menekan tombol simpan, maka data baru akan diperiksa dan disimpan ke dalam basis data apabila valid. Setelahnya, admin akan dibawa kembali ke halaman dashboard dengan notifikasi bahwa data telah diperbarui.

4.5 PENGUJIAN SISTEM

Dalam setiap melakukan perancangan system aplikasi, maka tahap terakhir sebelum dipublikasikan ke masyarakat pengguna, tentunya harus dilakukan pengecekan/testing pada software yang dibuat. Software Testing merupakan sebuah metode untuk mengetahui apakah software yang dibuat, berfungsi dengan baik dan benar. Tanpa adanya software testing kita tidak dapat mengetahui apakah sebuah software sudah memenuhi kriteria yang dibutuhkan untuk user.

Ada beberapa metode pengujian software testing yaitu *white Box Testing* dan *Black Box Testing* (Bois Bezier). Untuk *white Box Testing* merupakan salah satu cara untuk menguji suatu aplikasi atau software dengan melihat modul untuk memeriksa dan menganalisa kode program apakah ada yang salah atau tidak. Sedangkan untuk *Black Box Testing*, melakukan pengujian berdasarkan pada detail aplikasi seperti tampilan aplikasi, fungsifungsi yang ada pada aplikasi dan kesesuaian alur fungsi dengan sistem kerja yang diinginkan perancangannya (Uminingsih *et al.*, 2022).

4.5.1 Black Box Testing

Dalam penelitian ini, pengujian dilakukan dengan metode black box testing sebanyak tiga kali percobaan. Metode ini bertujuan untuk memastikan bahwa fungsi-fungsi perangkat lunak berjalan sesuai dengan rencana tanpa melihat struktur internal program. Black box testing fokus pada pengujian fungsionalitas dari sisi output untuk menemukan kemungkinan masalah (Purnamasari, D., Damayanti, S. U., Jumrianto, & Aini, N., 2024).

Salah satu fungsi yang diuji adalah halaman login admin, dimana pengujian dilakukan untuk memastikan proses autentikasi berjalan dengan benar. Pada pengujian ini, dilakukan input data username dan password valid serta tidak valid untuk mengetahui apakah sistem dapat mengizinkan akses yang sah dan menolak akses yang tidak sah sesuai dengan yang diharapkan.

Tabel 4.1 Login Admin

No	Langkah Pengujian	Input	Ekspetasi Hasil	Status
1	Masukkan username dan password valid	username: paranadamusik password: paranada123	Sistem menerima input dan berhasil masuk ke halaman dashboard admin	Berhasil
2	Masukkan username valid dan password salah	username: paranadamusik password: paranda111	Sistem menampilkan pesan error "Username atau password salah" dan tidak mengizinkan akses	Berhasil
3	Masukkan username kosong dan password valid	username: (kosong) password: paranada123	Sistem menampilkan pesan error "Username wajib diisi" dan tidak mengizinkan akses	Berhasil
4	Masukkan username dan password kosong	username: (kosong) password: (kosong)	Sistem menampilkan pesan error "Username dan password wajib diisi" dan tidak mengizinkan akses	Berhasil

Setelah proses login berhasil dilakukan, admin dapat mengakses fitur untuk mengelola data alat musik dalam sistem. Salah satu fitur yang diuji adalah penambahan data alat musik baru. Pengujian ini bertujuan untuk memastikan bahwa sistem dapat menerima input data dengan benar, serta mampu melakukan validasi terhadap kolom nama, harga, deskripsi, dan unggahan gambar. Berbagai skenario diuji menggunakan input yang valid maupun tidak valid untuk melihat apakah sistem merespons sesuai dengan yang diharapkan.

Tabel 4.2 Tambah Alat Musik (Admin)

No	Langkah Pengujian	Input	Ekspetasi Hasil	Status
1	Mengisi semua field dengan data valid	Nama: Gitar Akustik Harga: 500000 Deskripsi: Alat musik senar Gambar: katalog1.jpg	Sistem menyimpan data alat musik dan menampilkan notifikasi berhasil	Berhasil
2	Kolom nama dikosongkan	Nama: (kosong) Harga: 500000 Deskripsi: Alat musik senar Gambar: katalog1.jpg	Sistem menampilkan pesan error bahwa nama wajib diisi	Berhasil
4	Kolom harga dikosongkan	Nama: Gitar Akustik Harga: (kosong) Deskripsi: Alat musik senar Gambar: katalog1.jpg	Sistem menampilkan pesan error bahwa harga wajib diisi	Berhasil

5	Deskripsi dikosongkan	Nama: Gitar Akustik Harga: 500000 Deskripsi: (kosong) Gambar: katalog1.jpg	Sistem menampilkan pesan error karena deskripsi wajib diisi dan form tidak dapat disubmit jika kosong.	Berhasil
6	Tidak mengupload gambar	Nama: Gitar Akustik Harga: 500000 Deskripsi: Alat musik senar Gambar: (kosong)	Sistem tetap menyimpan data tanpa gambar karena gambar bersifat opsional	Berhasil

Selain menambahkan data, admin juga memiliki hak akses untuk menghapus data alat musik. Fitur ini berfungsi untuk mengelola data yang tidak lagi digunakan atau sudah tidak tersedia. Pengujian dilakukan untuk memastikan bahwa sistem dapat menghapus data dengan benar dan menampilkan konfirmasi sebelum penghapusan dilakukan.

Tabel 4.3 Hapus Alat Musik (Admin)

No	Langkah Pengujian	Input	Ekspetasi Hasil	Status
1	Klik tombol hapus pada data alat musik tertentu	Admin mengklik tombol "Hapus" pada baris alat musik di daftar data	Sistem menampilkan dialog konfirmasi sebelum menghapus	Berhasil
2	Konfirmasi penghapusan	Admin memilih "Ya" pada dialog konfirmasi	Data alat musik dihapus dari database dan tidak muncul lagi di daftar alat	Berhasil
3	Membatalkan penghapusan	Admin memilih "Batal" atau "Tidak" saat konfirmasi	Sistem membatalkan proses hapus dan data tetap ada di daftar	Berhasil

Fitur login dan registrasi merupakan bagian dari proses autentikasi pengguna. Registrasi digunakan oleh pengguna baru untuk membuat akun, sedangkan login digunakan oleh pengguna yang sudah terdaftar untuk mengakses sistem. Pengujian dilakukan untuk memastikan bahwa sistem dapat memvalidasi data input dengan benar, memberikan respon yang sesuai, dan mengelola hak akses pengguna secara aman.

Tabel 4.4 Registrasi/Login User

No	Langkah Pengujian	Input	Ekspetasi Hasil	Status
1	Registrasi dengan data lengkap dan valid	Username: user123 Password: user@2024 Konfirmasi Password: user@2024	Sistem menyimpan data dan menampilkan notifikasi registrasi berhasil	Berhasil
2	Username dikosongkan saat registrasi	Username: (kosong) Password: user@2024 Konfirmasi: user@2024	Sistem menampilkan pesan error bahwa username wajib diisi	Berhasil
3	Password dikosongkan saat registrasi	Username: user123 Password: (kosong) Konfirmasi: user@2024	Sistem menampilkan pesan error bahwa password wajib diisi	Berhasil
4	Konfirmasi password dikosongkan	Username: user123 Password: user@2024 Konfirmasi: (kosong)	Sistem menampilkan pesan error bahwa konfirmasi password wajib diisi	Berhasil
5	Password dan konfirmasi password tidak sama	Username: user123 Password: user@2024 Konfirmasi: beda123	Sistem menampilkan pesan error bahwa password tidak cocok	Berhasil
6	Username sudah digunakan saat registrasi	Username: user123 Password: user@baru Konfirmasi: user@baru	Sistem menolak dan menampilkan pesan bahwa username telah terdaftar	Berhasil
7	Login dengan data valid	Username: user123 Password: user@2024	Sistem berhasil login dan diarahkan ke halaman user	Berhasil
8	Login dengan password salah	Username: user123 Password: salah123	Sistem menampilkan pesan error bahwa login gagal	Berhasil
9	Login dengan username kosong	Username: (kosong) Password: user@2024	Sistem menampilkan pesan error bahwa username wajib diisi	Berhasil
10	Login dengan password kosong	Username: user123 Password: (kosong)	Sistem menampilkan pesan error bahwa password wajib diisi	Berhasil
11	Logout User	Klik tombol logout	Sistem menghapus sesi login dan kembali ke halaman login	Berhasil

Setelah berhasil melakukan registrasi akun dan login, pengguna diwajibkan untuk mengisi data KTP sebagai syarat sebelum dapat melakukan penyewaan alat musik. Fitur ini dirancang agar pengguna memasukkan informasi identitas berupa NIK, nama sesuai KTP, dan alamat tempat tinggal. Pengujian dilakukan untuk

memastikan bahwa sistem dapat memvalidasi input dengan benar, seperti memastikan panjang NIK sesuai ketentuan, serta semua field wajib diisi sebelum data dapat disimpan ke dalam sistem.

Tabel 4.5 Registrasi KTP

No	Langkah Pengujian	Input	Ekspetasi Hasil	Status
1	Input data lengkap dan valid	NIK: 1234567890123456 Nama: Budi Santoso Alamat: Jl. Merdeka No. 10	Sistem menyimpan data dan menampilkan notifikasi berhasil	Berhasil
2	NIK dikosongkan	NIK: (kosong) Nama: Budi Santoso Alamat: Jl. Merdeka No. 10	Sistem menampilkan pesan error bahwa NIK wajib diisi	Berhasil
3	NIK kurang dari 16 digit	NIK: 12345678901234 Nama: Budi Santoso Alamat: Jl. Merdeka No. 10	Sistem menolak input dan menampilkan pesan error bahwa NIK harus 16 digit	Berhasil
4	Nama dikosongkan	NIK: 1234567890123456 Nama: (kosong) Alamat: Jl. Merdeka No. 10	Sistem menampilkan pesan error bahwa nama wajib diisi	Berhasil
5	Alamat dikosongkan	NIK: 1234567890123456 Nama: Budi Santoso Alamat: (kosong)	Sistem menampilkan pesan error bahwa alamat wajib diisi	Berhasil
6	NIK berisi huruf	NIK: ABCD567890123456 Nama: Budi Santoso Alamat: Jl. Merdeka No. 10	Sistem menolak input dan menampilkan pesan error bahwa NIK harus angka	Berhasil

Setelah mengisi data KTP, pengguna akan diarahkan ke form sewa untuk menentukan periode peminjaman pada alat musik yang telah dipilih sebelumnya. Form ini terdiri atas pilihan tanggal mulai dan tanggal akhir sewa. Pengujian dilakukan untuk memastikan bahwa sistem memvalidasi kelengkapan data, mengecek logika tanggal (misalnya tanggal akhir tidak boleh sebelum tanggal mulai), dan menyimpan data sewa apabila semua input valid.

Tabel 4.6 Form Sewa Alat Musik.

No	Langkah Pengujian	Input	Ekspetasi Hasil	Status
1	Input tanggal awal dan akhir sewa valid	Tanggal awal: 2025-07-15 Tanggal akhir: 2025-07-20	Sistem menyimpan data sewa dan menampilkan konfirmasi berhasil	Berhasil

2	Tanggal akhir dikosongkan	Tanggal awal: 2025-07-15 Tanggal akhir: (kosong)	Sistem menampilkan pesan error bahwa tanggal akhir wajib diisi	Berhasil
3	Tanggal awal dikosongkan	Tanggal awal: (kosong) Tanggal akhir: 2025-07-20	Sistem menampilkan pesan error bahwa tanggal mulai wajib diisi	Berhasil
4	Tanggal akhir lebih awal dari tanggal mula	Tanggal awal: 2025-07-20 Tanggal akhir: 2025-07-15	Sistem menolak input dan menampilkan pesan error logika tanggal tidak valid	Berhasil
5	Tanggal mulai dan akhir sama	Tanggal awal: 2025-07-15 Tanggal akhir: 2025-07-15	Sistem menerima input jika diperbolehkan sewa 1 hari	Berhasil
6	Menyewa alat tanpa mengisi form sewa	Langsung klik “Sewa” tanpa input tanggal	Sistem menolak proses dan menampilkan pesan bahwa form harus diisi	Berhasil

Dari hasil pengujian black box yang dilakukan pada seluruh fitur dalam sistem, diketahui bahwa setiap fungsi berjalan sesuai dengan perencanaan tanpa ditemukan kesalahan. Oleh karena itu, secara fungsional, aplikasi ini telah memenuhi standar pengujian pada tahap pengembangan.

4.5.2 User Acceptance Testing (UAT)

User Acceptance Testing (UAT) merupakan tahapan verifikasi untuk memastikan bahwa solusi yang dikembangkan dalam sistem telah sesuai dengan kebutuhan dan harapan pengguna akhir (Suprpto, 2021). Pengujian UAT pada umumnya dilakukan sebelum peluncuran sebuah fitur baru di dalam aplikasi. Dengan melakukan ini pengembang dapat memahami apakah rancangan yang dibuat sudah memenuhi harapan pengguna (Gayatri, *et.,al*, 2022).

Pengujian dilakukan secara internal oleh pengembang dengan mensimulasikan skenario penggunaan sistem seperti registrasi, login, pencarian alat musik, proses penyewaan, hingga interaksi dengan chatbot. Penilaian dilakukan menggunakan skala Likert 1–5 terhadap lima aspek utama dari sisi pengguna, yaitu: kemudahan dalam menggunakan aplikasi, kejelasan informasi pada antarmuka, kelengkapan fitur penyewaan, responsivitas sistem dan kecepatan akses, serta penggunaan fitur chatbot dalam memberikan informasi.

Berikut adalah daftar pertanyaan yang digunakan dalam pengujian User Acceptance Testing (UAT) untuk mengevaluasi sistem penyewaan alat musik berbasis web. Pengguna diminta memberikan nilai pada setiap pertanyaan menggunakan skala Likert 1–5, dengan keterangan sebagai berikut:

1 = Sangat Tidak Setuju

2 = Tidak Setuju

3 = Netral

4 = Setuju

5 = Sangat Setuju

Tabel 4.7 Daftar Pertanyaan.

No	Pertanyaan
1	Aplikasi ini mudah digunakan
2	Informasi pada tampilan aplikasi mudah dipahami
3	Fitur penyewaan pada aplikasi ini sudah lengkap.
4	Aplikasi ini berjalan dengan cepat dan responsif.
5	Fitur chatbot dalam aplikasi ini membantu saya memahami informasi sewa.

Nilai yang ditampilkan pada tabel berikut merupakan rata-rata dari lima responden terhadap masing-masing pertanyaan yang diajukan.

Tabel 4.8 Hasil Penilaian User Acceptance Testing (UAT)

No	Aspek Pengujian	Rata rata skor (1-5)
1	Kemudahan dalam menggunakan aplikasi	4.4
2	Kejelasan informasi pada antarmuka	4.6
3	Kelengkapan fitur penyewaan	4.2
4	Responsivitas sistem dan kecepatan akses	4.4
5	Kegunaan fitur chatbot	4.3

Tabel 4.9 Penilaian User per Aspek

Responden	Kemudahan	Kejelasan	Kelengkapan	Kecepatan	Chatbot
R1	5	4	4	5	5
R2	4	4	4	4	4
R3	5	5	4	5	5
R4	4	4	5	4	5
R5	5	5	4	5	3

Langkah perhitungan:

- Kemudahan
 $(5 + 4 + 5 + 4 + 5) \div 5 = 23 \div 5 = 4.6$
- Kejelasan
 $(4 + 4 + 5 + 4 + 5) \div 5 = 22 \div 5 = 4.4$
- Kelengkapan
 $(4 + 4 + 4 + 5 + 4) \div 5 = 21 \div 5 = 4.2$
- Kecepatan
 $(5 + 4 + 5 + 4 + 5) \div 5 = 23 \div 5 = 4.6$
- Chatbot
 $(5 + 4 + 5 + 5 + 3) \div 5 = 22 \div 5 = 4.4$

Berdasarkan hasil simulasi pengujian, seluruh aspek memperoleh skor rata-rata di atas 4, yang mengindikasikan bahwa sistem telah memenuhi standar kelayakan dari segi fungsionalitas dan pengalaman pengguna. Oleh karena itu, aplikasi ini dinilai cukup layak untuk dikembangkan lebih lanjut dan dapat disempurnakan berdasarkan masukan dari pengguna aktual di tahap implementasi selanjutnya.

4.6 PEMBAHASAN

Berdasarkan hasil pengujian sistem menggunakan metode Black Box Testing dan *User Acceptance Testing (UAT)*, dapat disimpulkan bahwa aplikasi penyewaan alat musik online yang dibangun telah berjalan dengan baik dan memenuhi sebagian besar ekspektasi fungsional.

Melalui Black Box Testing, seluruh fitur inti diuji berdasarkan input dan output tanpa memeriksa struktur internal program. Hasil pengujian menunjukkan bahwa sistem mampu memvalidasi input pengguna dengan benar, menangani kesalahan dengan baik, serta menampilkan respons yang sesuai pada setiap proses, seperti login, registrasi, pengelolaan alat musik, pengisian data KTP, dan form sewa. Seluruh skenario pengujian menghasilkan status “Berhasil”, yang berarti sistem secara fungsional sudah berjalan sesuai dengan rancangan.

Sementara itu, melalui User Acceptance Testing (UAT), aplikasi diuji dari perspektif pengguna menggunakan lima indikator penilaian berbasis skala Likert (1–5). Hasil menunjukkan bahwa seluruh aspek mendapatkan skor rata-rata di atas 4, yaitu: kemudahan penggunaan (4.4), kejelasan informasi (4.6), kelengkapan fitur penyewaan (4.2), kecepatan akses (4.4), dan kegunaan fitur *chatbot* (4.3). Skor tertinggi diperoleh pada aspek kejelasan informasi, menunjukkan bahwa desain antarmuka dan penyampaian informasi sudah cukup efektif. Sementara itu, nilai 4.3 pada fitur *chatbot* menandakan bahwa fitur ini dinilai cukup membantu meskipun masih bisa dikembangkan lebih lanjut agar lebih interaktif dan fleksibel dalam memahami pertanyaan pengguna.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem telah memenuhi kriteria fungsional dan usability dasar. Namun, beberapa perbaikan pada pengalaman pengguna, seperti peningkatan kemampuan *chatbot* dan penyesuaian antarmuka berdasarkan umpan balik nyata, dapat menjadi langkah pengembangan selanjutnya. Dengan demikian, aplikasi ini dinilai layak digunakan dan memiliki potensi untuk diterapkan secara lebih luas.