

BAB 4

HASIL PENELITIAN

4.1 Ringkasan Hasil Penelitian

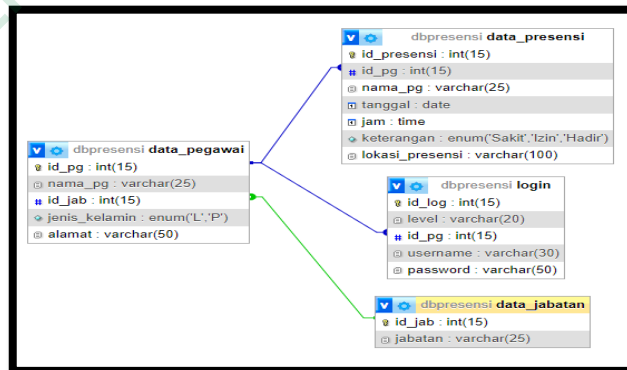
Pengembangan sistem presensi dengan *fingerprint* berbasis web ini menggunakan metode pengembangan perangkat lunak *waterfall*. Sistem presensi membantu dalam pencatatan informasi presensi menjadi lebih praktis. Berikut pembahasan lebih lanjut mengenai hasil penelitian.

4.2 Implementasi

Berikut uraian implementasi desain yang diterapkan pada pengembangan sistem presensi dengan *fingerprint* berbasis web.

4.2.1 Implementasi Desain Database

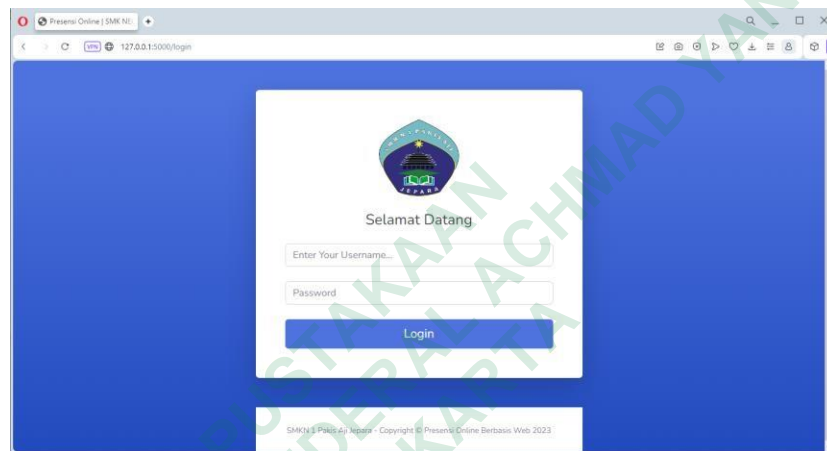
Hasil implementasi desain *database* yang telah dirancang dapat dilihat pada Gambar 4.1. terdapat empat tabel pada desain *database*. Tabel data Pegawai berisi id pegawai, nama pegawai, id jabatan, jenis kelamin dan alamat pegawai. Pada tabel data presensi berisi id presensi, id pegawai, nama pegawai, tanggal, jam, keterangan, dan lokasi presensi. Pada tabel login berisi id login, lever user, id pegawai, nama pegawai, password.



Gambar 4.1 Desain Database

4.2.2 Implementasi Desain Antarmuka Login

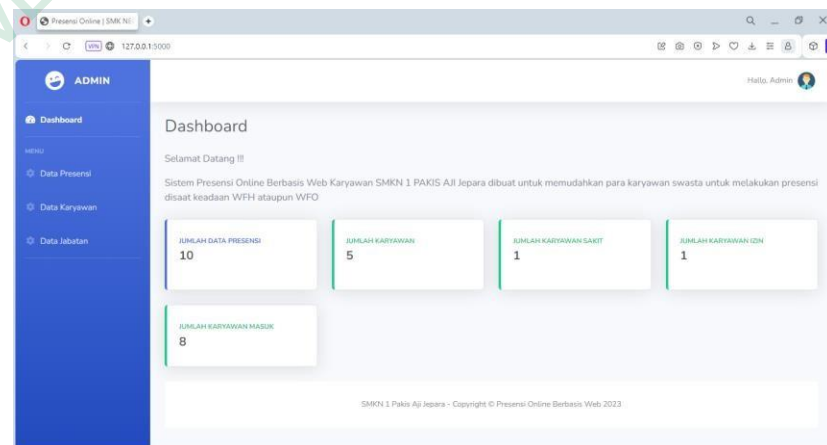
Implementasi dari halaman login yang dapat dilihat pada Gambar 4.2 digunakan untuk masuk kedalam sistem informasi. Pada halaman ini terdapat dua tabel yang berisi *username* dan *password* sebagai syarat untuk melanjutkan kehalaman berikutnya dan juga terdapat 3 level pengguna yaitu admin, kepala sekolah dan pegawai honorer yang terdaftar.



Gambar 4.2 Halaman Log In

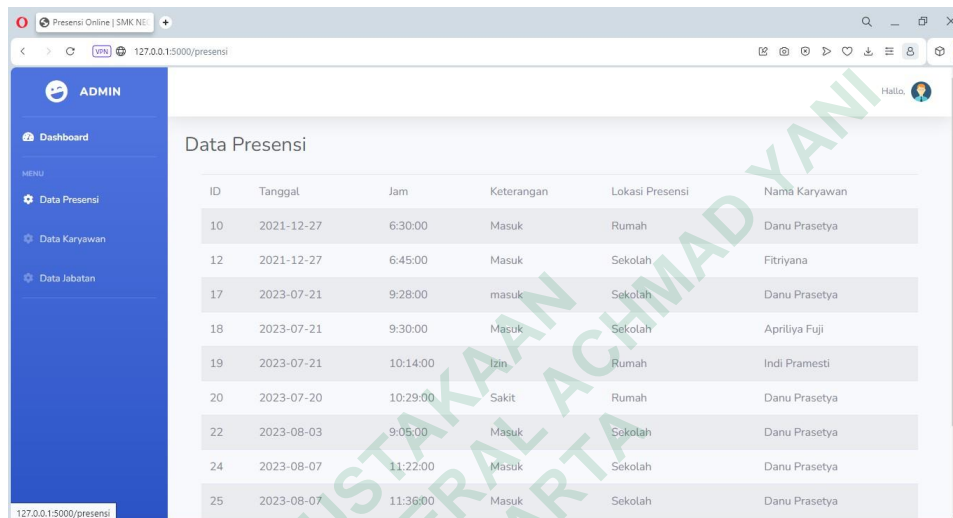
4.2.3 Implementasi Desain Antarmuka Admin

Tampilan halaman *dashboard* admin akan terbuka jika berhasil login. Halaman *dashboard* pada Gambar 4.3 berisi ringkasan dari beberapa menu yang ada yaitu data presensi, data pegawai dan data jabatan.



Gambar 4.3 Halaman Dashboard Admin

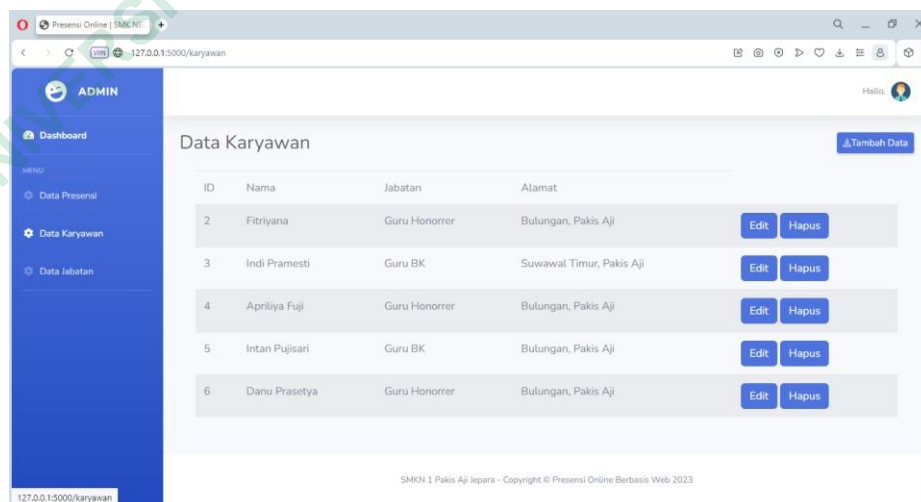
Halaman data presensi yang dapat dilihat pada Gambar 4.4 merupakan halaman data presensi pegawai yang telah berhasil disubmit. Pada halaman data presensi berisi informasi tanggal, jam, keterangan, lokasi dilakukannya presensi beserta nama pegawai yang terdaftar.



ID	Tanggal	Jam	Keterangan	Lokasi Presensi	Nama Karyawan
10	2021-12-27	6:30:00	Masuk	Rumah	Danu Prasetya
12	2021-12-27	6:45:00	Masuk	Sekolah	Fitriyana
17	2023-07-21	9:28:00	masuk	Sekolah	Danu Prasetya
18	2023-07-21	9:30:00	Masuk	Sekolah	Apriliya Fuji
19	2023-07-21	10:14:00	Izin	Rumah	Indi Pramesti
20	2023-07-20	10:29:00	Sakit	Rumah	Danu Prasetya
22	2023-08-03	9:05:00	Masuk	Sekolah	Danu Prasetya
24	2023-08-07	11:22:00	Masuk	Sekolah	Danu Prasetya
25	2023-08-07	11:36:00	Masuk	Sekolah	Danu Prasetya

Gambar 4.4 Halaman Data Presensi Admin

Halaman data pegawai seperti yang terlihat pada Gambar 4.5 merupakan halaman data pegawai yang terdaftar di dalam database, yang berisikan informasi nama pegawai, jabatan serta alamat tempat tinggal.



ID	Nama	Jabatan	Alamat	
2	Fitriyana	Guru Honorer	Bulungan, Pakis Aji	Edit Hapus
3	Indi Pramesti	Guru BK	Suwawal Timur, Pakis Aji	Edit Hapus
4	Apriliya Fuji	Guru Honorer	Bulungan, Pakis Aji	Edit Hapus
5	Intan Pujisari	Guru BK	Bulungan, Pakis Aji	Edit Hapus
6	Danu Prasetya	Guru Honorer	Bulungan, Pakis Aji	Edit Hapus

Gambar 4.5 Halaman Data Pegawai Admin

Halaman tambah data pegawai pada Gambar 4.6 berisi informasi nama pegawai, jabatan, dan alamat. Penambahan data pegawai hanya bisa dilakukan admin, pegawai hanya menambahkan data *fingerprint* melalui menu profil pegawai.

The screenshot shows a web browser window with the URL '127.0.0.1:5000/insertkaryawan'. The page title is 'ADMIN' and the main heading is 'Tambah Data Karyawan'. The form contains three input fields: 'Nama', 'Jabatan', and 'Alamat', each with a corresponding label. Below the fields is a 'Simpan' button. The sidebar on the left lists 'Dashboard', 'Data Presensi', 'Data Karyawan', and 'Data Jabatan'. The footer text reads 'SMKN 1 Pakis Aji Jepara - Copyright © Presensi Online Berbasis Web 2023'.

Gambar 4.6 Tambah Data Pegawai Admin

Halaman data jabatan pada Gambar 4.7 merupakan informasi data jabatan yang berisi nama-nama jabatan yang dijabat oleh pegawai honorer yang terdaftar di SMK Negeri 1 Pakis Aji, Jepara.

The screenshot shows a web browser window with the URL '127.0.0.1:5000/jabatan'. The page title is 'ADMIN' and the main heading is 'Data Jabatan'. A 'Tambah Data' button is in the top right. The table below has the following data:

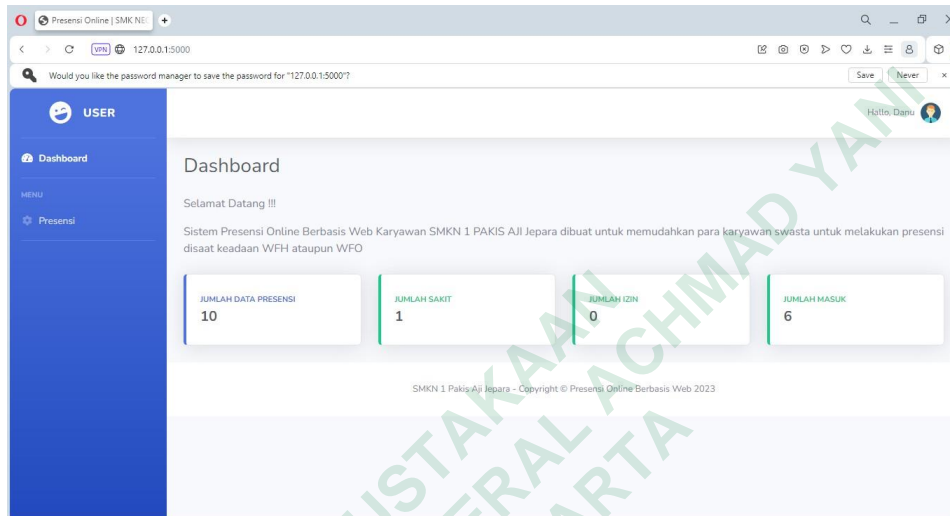
ID	Nama Jabatan	
1	Staff TU	Hapus
3	Guru BK	Hapus
4	Guru Honorer	Hapus

The sidebar on the left lists 'Dashboard', 'Data Presensi', 'Data Karyawan', and 'Data Jabatan'. The footer text reads 'SMKN 1 Pakis Aji Jepara - Copyright © Presensi Online Berbasis Web 2023'.

Gambar 4.7 Halaman Data Jabatan Admin

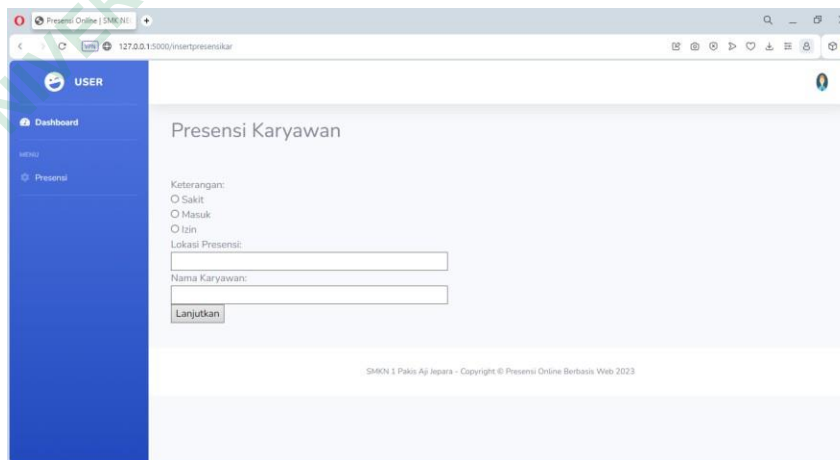
4.2.4 Implementasi Desain Antarmuka Pegawai

Tampilan halaman *dashboard* pegawai yang berhasil login dapat dilihat pada Gambar 4.8. Halaman *dashboard* berisi ringkasan dari beberapa menu yang ada yaitu data presensi.



Gambar 4.8 Halaman *Dasboard* Pegawai

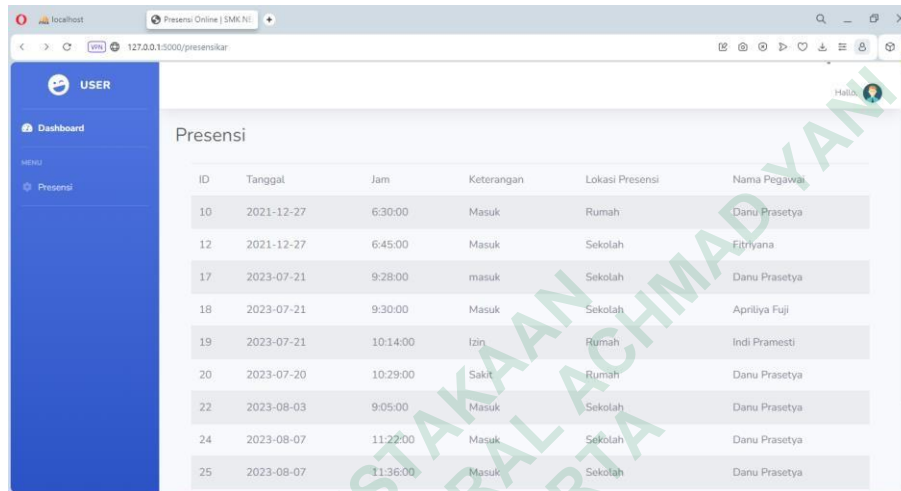
Halaman form presensi yang berisi data keterangan, lokasi dilakukannya presensi, dan mana pegawai yang ingin melakukan presensi, setelah pengisian akan muncul menu pop up untuk dilanjutkan dengan verifikasi *fingerprint* yang sudah terdaftar, yang dapat dilihat pada Gambar 4.9.



Gambar 4.9 Halaman Input Presensi Pegawai

4.2.5 Implementasi Desain Antarmuka Kepala Sekolah

Tampilan yang muncul setelah kepala sekolah melakukan login adalah tampilan data presensi pegawai yang telah melakukan presensi seperti yang ada pada Gambar 4.10.



ID	Tanggal	Jam	Keterangan	Lokasi Presensi	Nama Pegawai
10	2021-12-27	6:30:00	Masuk	Rumah	Danu Prasetya
12	2021-12-27	6:45:00	Masuk	Sekolah	Fitriyana
17	2023-07-21	9:28:00	masuk	Sekolah	Danu Prasetya
18	2023-07-21	9:30:00	Masuk	Sekolah	Apriliya Fuji
19	2023-07-21	10:14:00	Izin	Rumah	Indi Pramesti
20	2023-07-20	10:29:00	Sakit	Rumah	Danu Prasetya
22	2023-08-03	9:05:00	Masuk	Sekolah	Danu Prasetya
24	2023-08-07	11:22:00	Masuk	Sekolah	Danu Prasetya
25	2023-08-07	11:36:00	Masuk	Sekolah	Danu Prasetya

Gambar 4.10 Halaman Data Presensi Pegawai bagi Kepala Sekolah

4.3 Pengkodean

Pengkodean yang digunakan untuk pengembangan sistem informasi presensi yaitu menggunakan bahasa pemrograman Python dengan *framework* Flask. Berikut ringkasan penjelasan beberapa kode penting yang digunakan.

4.3.1 Kode bagian Login

Kode sistem informasi bagian login di gunakan para aktor untuk masuk ke halaman dashboard, jika *username* dan *password* yang dimasukkan tidak sesuai maka akan tetap di halam login dan ada pesan bahwa *username/password* salah. Berikut kode bagian login yang di simpan dengan nama *file* models.py.

```
def __init__(self, username=None, password=None):
    self.username = username
    self.password = password

def openDB(self):
    global db, cursor
    db = pymysql.connect(
        host=config.DB_HOST,
        user=config.DB_USER,
```

```

        password=config.DB_PASSWORD,
        database=config.DB_NAME,
        port=config.DB_PORT)
    cursor = db.cursor()
def authenticate(self):
    self.openDB()
    cursor.execute("SELECT COUNT(*) FROM login WHERE username = '%s'
AND password = '%s' " % (self.username, self.password))
    count_account = (cursor.fetchone())[0]
    self.closeDB()
    return True if count_account>0 else False

def accountType(self):
    self.openDB()
    cursor.execute("SELECT * FROM login WHERE username = '%s' AND
password = '%s' " % (self.username, self.password))
    account = (cursor.fetchone())
    self.closeDB()
    if account[1] == 'admin':
        return 'admin'
    elif account[1] == 'user':
        return 'user'
    elif account[1] == 'kepsek':
        return 'kepsek'

```

Berikut tampilan kode bagian login yang di simpan dengan nama *file* app.py.

```

@app.application.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        pengguna = MDatabase(username, password)
        if pengguna.authenticate():
            session['username'] = username
            session['level'] = pengguna.accountType()
            return redirect(url_for('index'))
        msg = 'Username/Password salah.'
        return render_template('login.html', msg=msg)
    return render_template('login.html')

```

4.3.2 Kode bagian Dashboard

Kode bagian *dashboard* berfungsi untuk menampilkan tampilan *dashboard* seperti yang dapat dilihat pada Gambar 4.3 dan Gambar 4.8. Berikut kode yang di simpan dengan nama *file* models.py.

```

def selectjumlah_data_presensi(self):
    self.openDB()

```

```

        cursor.execute("SELECT count(id_presensi) FROM presensi")
        jumlah_data_presensi = cursor.fetchone()[0]
        return jumlah_data_presensi
    def selectjumlah_karyawan(self):
        self.openDB()
        cursor.execute("SELECT count(nama_kar) FROM karyawan")
        jumlah_karyawan = cursor.fetchone()[0]
        return jumlah_karyawan

    def selectjumlah_masuk(self):
        self.openDB()
        cursor.execute("SELECT count(id_presensi) FROM presensi WHERE
            keterangan='masuk' ")
        jumlah_masuk = cursor.fetchone()[0]
        return jumlah_masuk

    def selectjumlah_izin(self):
        self.openDB()
        cursor.execute("SELECT count(id_presensi) FROM presensi WHERE
            keterangan='izin' ")
        jumlah_izin = cursor.fetchone()[0]
        return jumlah_izin

    def selectjumlah_sakit(self):
        self.openDB()
        cursor.execute("SELECT count(id_presensi) FROM presensi WHERE
            keterangan='sakit' ")
        jumlah_sakit = cursor.fetchone()[0]
        return jumlah_sakit

    def selectjumlah_mkar(self):
        self.openDB()
        cursor.execute("SELECT count(id_presensi) FROM presensi WHERE
            keterangan='masuk' AND nama_kar='Danu Prasetya' ")
        jumlah_mkar = cursor.fetchone()[0]
        return jumlah_mkar

    def selectjumlah_ikar(self):
        self.openDB()
        cursor.execute("SELECT count(id_presensi) FROM presensi WHERE
            keterangan='izin' AND nama_kar='Danu Prasetya' ")
        jumlah_ikar = cursor.fetchone()[0]
        return jumlah_ikar

    def selectjumlah_skar(self):
        self.openDB()
        cursor.execute("SELECT count(id_presensi) FROM presensi WHERE
            keterangan='sakit' AND nama_kar='Danu Prasetya' ")
        jumlah_skar = cursor.fetchone()[0]
        return jumlah_skar

```


Berikut tampilan kode bagian *dashboard* yang di simpan dengan nama *file* app.py.

```
@application.route('/')
def index():
    models = MDatabase()
    if 'username' in session and session['level'] == 'admin':
        username = session['username']
        level = session['level']
        jumlah_data_presensi = models.selectjumlah_data_presensi()
        jumlah_karyawan = models.selectjumlah_karyawan()
        jumlah_masuk = models.selectjumlah_masuk()
        jumlah_izin = models.selectjumlah_izin()
        jumlah_sakit = models.selectjumlah_sakit()
        print(jumlah_data_presensi)
        return render_template('index.html', username=username,
                               level=level, jumlah_data_presensi=jumlah_data_presensi,
                               jumlah_karyawan=jumlah_karyawan, jumlah_masuk=jumlah_masuk,
                               jumlah_izin=jumlah_izin, jumlah_sakit=jumlah_sakit)
    elif 'username' in session and session['level'] == 'user':
        username = session['username']
        level = session['level']
        jumlah_data_presensi = models.selectjumlah_data_presensi()
        jumlah_mkar = models.selectjumlah_mkar()
        jumlah_ikar = models.selectjumlah_ikar()
        jumlah_skar = models.selectjumlah_skar()
        print(jumlah_data_presensi)
        return render_template('index1.html', username=username,
                               level=level, jumlah_data_presensi=jumlah_data_presensi,
                               jumlah_mkar=jumlah_mkar, jumlah_ikar=jumlah_ikar,
                               jumlah_skar=jumlah_skar)
    elif 'username' in session and session['level'] == 'kepek':
        username = session['username']
        level = session['level']
        jumlah_data_presensi = models.selectjumlah_data_presensi()
        jumlah_karyawan = models.selectjumlah_karyawan()
        jumlah_masuk = models.selectjumlah_masuk()
        jumlah_izin = models.selectjumlah_izin()
        jumlah_sakit = models.selectjumlah_sakit()
        print(jumlah_data_presensi)
        return render_template('index2.html', username=username,
                               level=level, jumlah_data_presensi=jumlah_data_presensi,
                               jumlah_karyawan=jumlah_karyawan, jumlah_masuk=jumlah_masuk,
                               jumlah_izin=jumlah_izin, jumlah_sakit=jumlah_sakit)
    return redirect(url_for('login'))
```

4.3.3 Kode bagian Presensi Pegawai

Kode bagian presensi pegawai berfungsi untuk menampilkan tampilan form presensi yang diisi oleh pegawai seperti yang dapat dilihat pada Gambar 4.9. Berikut kode yang di simpan dengan nama *file* models.py.

```
def insertpresensikarDB(self, data):
    self.openDB()
    cursor.execute("INSERT INTO presensi (id_presensi, tanggal, jam,
keterangan, lokasi, nama_kar) VALUES ('%s', '%s', '%s', '%s', '%s',
'%s')" % data)
    db.commit()
    self.closeDB
```

Berikut tampilan kode bagian presensi pegawai yang di simpan dengan nama *file* app.py.

```
@application.route('/presensikar', methods=['GET', 'POST'])
def presensikar():
    models = MDatabase()
    container = []
    container = models.selectpresensikarDB()
    return render_template('presensikar.html', container=container)

@application.route('/insertpresensikar', methods=['GET', 'POST'])
def insertpresensikar():
    now = datetime.now()
    if request.method == 'POST':
        id_presensi = request.form['id_presensi']
        tanggal = now.strftime("%Y/%m/%d")
        jam = now.strftime("%H:%M")
        keterangan = request.form['keterangan']
        lokasi = request.form['lokasi']
        nama_kar = request.form['nama_kar']
        data = (id_presensi,tanggal,jam,keterangan,lokasi,nama_kar)
        models = MDatabase()
        models.insertpresensikarDB(data)
        return redirect(url_for('presensikar'))
    else:
        return render_template('insert_kar.html')
```

4.3.4 Kode bagian Data Presensi

Kode bagian data presensi pegawai berfungsi untuk menampilkan tampilan form data presensi yang disimpan seperti yang dapat dilihat pada Gambar 4.4 dan Gambar 4.10. Berikut kode yang di simpan dengan nama *file* models.py.

```

def selectpresensiDB(self):
    self.openDB()
    cursor.execute("SELECT * FROM presensi")
    container = []
    for id_presensi,tanggal,jam, keterangan,lokasi,nama_kar in
        cursor.fetchall():
            container.append((id_presensi,tanggal,jam,keterangan,lokasi,na
                ma_kar))
    self.closeDB()
    return container

```

Berikut tampilan kode bagian presensi pegawai yang di simpan dengan nama *file* app.py.

```

@application.route('/presensi', methods=['GET', 'POST'])
def presensi():
    models = MDatabase()
    container = []
    container = models.selectpresensiDB()
    return render_template('presensi.html', container=container)

```

4.4 Pembahasan

Sistem presensi pegawai ini digunakan untuk mempermudah pegawai dalam melakukan kegiatan presensi saat hadir di sekolah. Sistem presensi yang digunakan sebelumnya adalah menggunakan sistem presensi secara manual yaitu menuliskan data diri yang ditandatangani ke dalam buku besar untuk menandakan bahwa pegawai tersebut hadir.

Sistem informasi yang dibangun ini memiliki tiga level aktor yaitu admin, kepala sekolah dan pegawai honorer, disetiap aktor memiliki fungsinya masing-masing yaitu aktor admin dapat melakukan tambah, edit, dan hapus data pegawai honorer, melihat data pegawai yang telah melakukan presensi. Aktor kepala sekolah hanya dapat melihat data pegawai yang telah melakukan presensi. Aktor pegawai dapat melihat data pegawai yang telah melakukan presensi dan melakukan presensi.

Dari hasil pengujian menggunakan *blackbox testing* yang akan dijabarkan pada sub bab selanjutnya mendapatkan *presentase* sebesar 75%. Kekurangan dari

sistem presensi yang dibangun iyalah belum bisa dijalankan fungsi lokasi secara otomatis.

4.4.1 Pengujian bagian Admin

Pengujian bagian admin dilakukan menggunakan metode *blackbox testing* data pengujian dapat dilihat dalam *test case* tabel 4.1.

Tabel 4.1 *Test Case* bagian Admin

No.	Fungsi	Kasus Uji	Hasil yang diharapkan	Berhasil	Gagal
1.	Login	Memasukkan Username dan password yang sudah terdaftar di database	Login berhasil masuk ke halaman dashboard sesuai level user	√	
		Memasukkan Username dan password yang salah atau tidak terdaftar di database	Login gagal atau tidak berhasil masuk ke halaman dashboard sesuai level user	√	
2.	Tambah data pegawai	Mengisi form dengan data sesuai data pegawai	Pengisian berhasil tersimpan kedalam database	√	
		Mengisi form dengan data yang tidak sesuai	Pengisian gagal tersimpan ke data base dan tetap pada halaman tambah data	√	
3.	Edit data pegawai	Mengisi form dengan data sesuai data pegawai	Pengisian berhasil tersimpan kedalam database	√	
		Mengisi form dengan data yang tidak sesuai	Pengisian gagal tersimpan ke data base dan tetap pada halaman tambah data	√	
4.	Hapus data pegawai	Menghapus data yang tersimpan di database	Data berhasil dihapus dari database	√	
5.	Melihat Presensi	Melihat data presensi para pegawai	Memuat tampilan data presensi pegawai	√	

4.4.2 Pengujian bagian Kepala Sekolah

Pengujian bagian kepala sekolah dilakukan menggunakan metode *blackbox testing* data dapat dilihat dalam *test case* tabel 4.2.

Tabel 4.2 *Test Case* bagian Kepala Sekolah

No.	Fungsi	Kasus Uji	Hasil yang diharapkan	Berhasil	Gagal
1.	Login	Memasukkan Username dan password yang sudah terdaftar di database	Login berhasil masuk ke halaman dashboard sesuai level user	√	
		Memasukkan Username dan password yang salah atau tidak terdaftar di database	Login gagal atau tidak berhasil masuk ke halaman dashboard sesuai level user	√	
2.	Melihat Presensi	Melihat data presensi para pegawai	Memuat tampilan data presensi pegawai	√	

4.4.3 Pengujian bagian Pegawai

Pengujian bagian kepala sekolah dilakukan menggunakan metode *blackbox testing* data dapat dilihat dalam *test case* tabel 4.3.

Tabel 4.3 *Test Case* bagian Pegawai

No.	Fungsi	Kasus Uji	Hasil yang diharapkan	Berhasil	Gagal
1.	Login	Memasukkan Username dan password yang sudah terdaftar di database	Login berhasil masuk ke halaman dashboard sesuai level user	√	
		Memasukkan Username dan password yang salah atau tidak terdaftar di database	Login gagal atau tidak berhasil masuk ke halaman dashboard sesuai level user	√	

2.	Melihat Presensi	Melihat data presensi para pegawai	Memuat tampilan data presensi pegawai	v
3.	Melakukan Presensi	Memasukkan fingerprint yang terdaftar	Presensi berhasil data masuk ke database	v
		Memasukkan fingerprint yang tidak terdaftar	Presensi gagal dan tetap pada halaman presensi	v

PERPUSTAKAAN
UNIVERSITAS JENDERAL ACHMAD YANI
YOGYAKARTA