

## BAB 4

### HASIL PENELITIAN

#### 4.1 RINGKASAN HASIL PENELITIAN

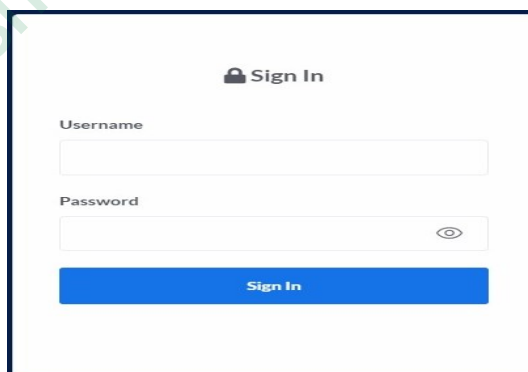
Sistem Laporan Evaluasi Kinerja Dosen Ftti Di Universitas Jenderal Achmad Yani Yogyakarta dirancang dengan menggunakan Python dan MySQL. Aplikasi yang dibuat akan memudahkan Anda untuk melacak dan memeriksa data historis karena data disimpan dalam database sistem.

#### 4.2 IMPLEMENTASI DESAIN ANTARMUKA

Berikut tampilan Sistem Pelaporan Evaluasi Kinerja Instruktur FTTI Universitas Jenderal Achmad Yani Yogyakarta.

##### 4.2.1 Login Pegawai

Halaman login pegawai digunakan pegawai untuk login ke sistem dan mengunggah data hasil penilaian kinerja guru. Untuk login, karyawan harus memasukkan username dan password yang benar. Jika salah, pegawai tidak dapat masuk ke sistem untuk mengunggah data hasil penilaian kinerja instruktur pengawas. Tampilan dapat dilihat pada Gambar 4.1.



The image shows a web interface for a sign-in page. At the top center, there is a lock icon followed by the text "Sign In". Below this, there are two input fields. The first is labeled "Username" and is empty. The second is labeled "Password" and is also empty, with a small eye icon to its right for toggling visibility. At the bottom of the form, there is a solid blue button with the text "Sign In" in white.

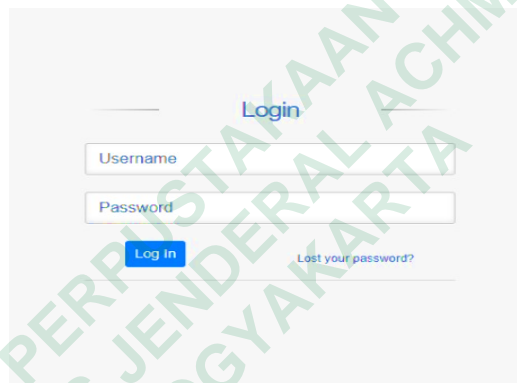
**Gambar 4.1** Login Pegawai

Berikut potongan kode untuk menampilkan halaman login pegawai.

```
1. @app.route('/staffLogin')
2. def staffLogin():
3.     session.pop('npp', None)
4.     return render_template("login.html");
```

#### 4.2.2 Login Dosen

Pada halaman ini, pengguna hanya perlu login karena sudah didaftarkan oleh administrator. Pengguna terdaftar dapat memeriksa peringkat pembicara yang dimasukkan oleh admin. Pengguna hanya perlu login dengan memasukkan username dan password yang diberikan oleh administrator. Halaman login pembicara dapat dilihat pada Gambar 4.2.



**Gambar 4.2** Login Dosen

Berikut potongan kode untuk menampilkan halaman login dosen.

```
1. @app.route('/login_dosen')
2. def login_dosen():
3.     session.pop('username', None)
4.     return render_template("admin_login.html")
```

#### 4.2.3 Input Data Penilaian

Input data penilaian digunakan menginputkan komponen-komponen data penilaian. Hasil penilaian dosen yang sudah di upload selanjutnya akan masuk ke hasil penilaian kinerja. Contoh halaman input data penilaian dilihat pada Gambar 4.3.

**Gambar 4.3** Input Data

Berikut potongan kode untuk menampilkan halaman input data penilaian

```

1. @app.route('/inputPenilaian')
2. def inputPenilaian():
3.     session.pop('npp', None)
4.     if session.get('username'):
5.         openDb()
6.
7.         container = []
8.         sql = "SELECT * FROM clustering_dosen"
9.         cursor.execute(sql)
10.        results = cursor.fetchall()
11.        for data in results:
12.            container.append(data)
13.        closeDb()
14.        return render_template('input_penilaian.html',
15.                               container=container,)
16.    else:
17.        return render_template('login.html')

```

#### 4.2.4 Laporan Dosen Perprodi

Halaman data dosen digunakan untuk melihat hasil dari data-data dosen pengampu yang telah di upload melalui data penilaian dosen. data penilaian yang sudah tercantum dapat dilakukan pengecekan kembali. halaman data dosen dapat dilihat pada Gambar 4.4.

No	Kode Dosen	Nama	NIP/NPP	Pendidikan	Prodi	Status	Mata Kuliah	Kode Mata Kuliah	Jam Mengajar	Kode Mengajar	Action
1	FTT001	Arif Wahyu Muryanto, S.Kom, M.Cs.	11234412	S2	Sistem Informasi	Tetap	Sistem Manajemen Basis Data	IY114	12 Jam	JUL2023	TAMBAH
2	FTT0101	Puji Winar Cahyo, S.Kom, M.Cs.	15112196	S2	Informatika	Tetap	Pengantar Teknologi Informasi dan Komunikasi	CE112	8 Jam	PTI2023	TAMBAH
3	IFK001	Choenun Anawi, S.Kom, M.Kom.	1234512	S2	Informatika	Tetap	Algoritma dan Struktur Data	CE314	12 Jam	JUL2023	TAMBAH

**Gambar 4.4** Laporan Dosen Perprodi

Berikut potongan kode untuk menampilkan halaman data dosen.

```

1. @app.route('/dataDosen')
2. def dataDosen():
3.     session.pop('npp', None)
4.     if session.get('username'):
5.         openDb()
6.
7.         container = []
8.         sql = "SELECT * FROM clustering_dosen"
9.         cursor.execute(sql)
10.        results = cursor.fetchall()
11.        for data in results:
12.            container.append(data)
13.
14.        container2 = []
15.        sql = "SELECT * FROM clustering_makul"
16.        cursor.execute(sql)
17.        results = cursor.fetchall()
18.        for data in results:
19.            container2.append(data)
20.        closeDb()
21.        return render_template('data_dosen.html', container=container,
22.                               container2=container2)
22.    else:
23.        return render_template('login.html')

```

#### 4.2.5 Laporan Kinerja Dosen

Halaman laporan kinerja dosen digunakan untuk melihat semua data kinerja dosen yang sudah di inputkan. Halaman laporan kinerja dosen dapat dilihat pada Gambar 4.5.

No	Nama	Matakuliah yang diampu	Total jam mengajar	Kesiapan Mengajar	Materi Pengajaran	Disiplin Mengajar	Evaluasi Mengajar	Kepribadian Dosen	Total Skor	Predikat
1	Aris Wahyu Muryanto, S.Kom., M.Cs.	Sistem Manajemen Basis Data	12 Jam	4,00	4,00	4,00	4,00	4,00	7200	Sangat Memuaskan
2	Puji Winar Cahyo, S.Kom., M.Cs.	Pengantar Teknologi Informasi dan Komunikasi	8 Jam	3,00	3,00	4,00	2,00	3,00	3520	Kurang Memuaskan

Showing 1 to 1 of 1 entries

Previous 1 Next

2023, made by Jenny Rombe Langsa

**Gambar 4.5** Laporan Kinerja Dosen

Berikut potongan kode yang digunakan untuk menampilkan halaman untuk menampilkan laporan kinerja dosen.

```

1. data = {
2.     'nilaikm': nilaikm,
3.     'nilaimp': nilaimp,
4.     'nilaidm': nilaidm,
5.     'nilaiemj': nilaiemj,
6.     'nilaikd': nilaikd,
7.     'total': total,
8.     'predikat': predikat,
9.     'namadosen': namadosen,
10.    'namamakul': namamakul,
11.    'totaljam': totaljam,
12. }
13. data_kinerja.append(data)
14. return render_template('pegawai/data_kinerja.html',
    data_kinerja=data_kinerja, container=container, title=title)

```

#### 4.2.6 Tambah Data Dosen

Halaman tambah data dosen digunakan untuk menginputkan data, status dan jabatan dosen. Hasil tambah data dosen akan tercantum ke dalam data dosen. halaman tambah data dosen dapat dilihat pada Gambar 4.6.

**Gambar 4.6** Tambah Data Dosen

Berikut potongan kode untuk menampilkan halaman tambah data dosen.

```

1. @app.route('/addDosen', methods=['POST', 'GET'])
2. def addDosen():
3.     if request.method == 'POST':
4.
5.         kode_dosen = request.form['kd_dosen']
6.         nama = request.form['nama']
7.         npp = request.form['npp']
8.         password = request.form['password']
9.         pendidikan = request.form['pendidikan']
10.        jabatan = request.form['jabatan']
11.        work_status = request.form['work_status']
12.        mata_kul = request.form['makul']
13.        kode_makul = request.form['kode_mk']
14.        jam_ampu = request.form['jam_ajar']
15.        kode_ampu = request.form['kd_ajar']
16.
17.        openDb()
18.        sql = "INSERT INTO clustering_dosen (kode_dosen, nama, npp,
19.        password, pendidikan, jabatan, work_status, mata_kul, kode_makul,
20.        jam_ampu, kode_ampu) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)"
21.        val = (kode_dosen, nama, npp, password, pendidikan, jabatan,
22.        work_status, mata_kul, kode_makul, jam_ampu, kode_ampu)
23.        cursor.execute(sql, val)
24.        conn.commit()
25.        try:
26.            container = []
27.            sql = "SELECT * FROM clustering_dosen"
28.            cursor.execute(sql)
29.            results = cursor.fetchall()
30.            for data in results:
31.                container.append(data)

```

```

29.         closeDb()
30.         return render_template('data_dosen.html', container=container,)
31.     except:
32.         return render_template('index.html')
33.     else:
34.         return render_template('data_dosen.html')

```

#### 4.2.7 Data Mata Kuliah

Halaman data mata kuliah digunakan untuk data dosen pengampu mengajar mata kuliah yang ditentukan. Dosen pengampu mengajar mata kuliah sesuai bidangnya masing-masing. halaman hasil data mata kuliah dapat dilihat pada Gambar 4.7.

No	Nama Mata Kuliah	Kode Mata Kuliah	Action
1	Praktikum Aplikasi Komputer I	CA121	[edit] [delete]
2	Pengantar Teknologi Informasi dan Komunikasi	CE122	[edit] [delete]
3	Praktikum Aplikasi Komputer II	CA131	[edit] [delete]
4	Pemrograman Lanjut	CE224	[edit] [delete]
5	Komunikasi Data dan Jaringan Komputer	NC224	[edit] [delete]
6	Bahasa Inggris II	SC353	[edit] [delete]
7	Pendidikan Kewarganegaraan	UN122	[edit] [delete]
8	Sistem Manajemen Basis Data	IY114	[edit] [delete]
9	Algoritma dan Struktur Data	CE114	[edit] [delete]
10	Pengembangan Aplikasi Mobile	CE114	[edit] [delete]

**Gambar 4.7** Data Mata Kuliah

Berikut potongan kode untuk menampilkan halaman data mata kuliah.

```

1. @app.route('/dataMakul')
2. def dataMakul():
3.     session.pop('npp', None)
4.     if session.get('username'):
5.         openDb()
6.
7.         container = []
8.         sql = "SELECT * FROM clustering_makul"
9.         cursor.execute(sql)
10.        results = cursor.fetchall()
11.        for data in results:
12.            container.append(data)
13.        closeDb()
14.        return render_template('data_makul.html', container=container,)
15.    else:
16.        return render_template('login.html')

```

#### 4.2.8 Input Data Mata Kuliah

Halaman tambah data mata kuliah digunakan untuk menginputkan dosen pengampuh dalam mengisi mata kuliah dengan bidang yang dikuasainya. Hasil tambah data dosen akan tercantum ke data mata kuliah. halaman tambah mata kuliah dapat dilihat pada Gambar 4.8

**Gambar 4.8** Input Mata Kuliah

Berikut potongan kode untuk melihat halaman input data mata kuliah.

```

1. @app.route('/addMakul', methods=['POST', 'GET'])
2. def addMakul():
3.     if request.method == 'POST':
4.         nama_makul = request.form['nama_makul']
5.         kode_makul = request.form['kode_makul']
6.
7.         openDb()
8.         sql = "INSERT INTO clustering_makul (nama_makul, kode_makul)
          VALUES (%s, %s)"
9.         val = (nama_makul, kode_makul)
10.        cursor.execute(sql, val)
11.        conn.commit()
12.
13.        try:
14.            container = []
15.            sql = "SELECT * FROM clustering_makul"
16.            cursor.execute(sql)
17.            results = cursor.fetchall()
18.            for data in results:
19.                container.append(data)
20.        closeDb()
21.        return render_template('data_makul.html', container=container,)
22.    except:

```



```

23.         return render_template('data_makul.html')
24.     else:
25.         return render_template('data_makul.html')

```

#### 4.2.9 Hasil Penilaian Kinerja

Halaman hasil penilaian kinerja digunakan untuk output dari input data penilaian dengan komponen-komponen data yang telah dihitung keseluruhan. halaman hasil penilaian kinerja dapat dilihat pada Gambar 4.9.

No	Nama	Matakuliah yang diampu	Total jam mengajar	Keselapan Mengajar	Materi Pengajaran	Disiplin Mengajar	Evaluasi Mengajar	Kepribadian Dosen	Total Skor
1	Andika Dayu Saputra	Pengembangan Aplikasi Web	12 Jam	3.00	2.00	3.00	4.00	4.00	5520

Predikat: Cukup

Showing 1 to 1 of 1 entries

**Gambar 4.9** Hasil Penilaian Kinerja

Berikut potongan kode untuk menampilkan halaman hasil penilaian kinerja.

```

1. @app.route('/tablePenilaianKinerja')
2. def tablePenilaianKinerja():
3.     session.pop('username', None)
4.     if session.get('npp'):
5.
6.         openDb()
7.         container = []
8.         ceksql = "SELECT * FROM clustering_dosen where
npp='"+session.get('npp')+"'"
9.         cursor.execute(ceksql)
10.        cekresults = cursor.fetchall()
11.        for data in cekresults:
12.            container.append(data)
13.
14.            nama = data['nama'];
15.
16.            result = []
17.            sql = "SELECT * FROM clustering_hasil where
dosen_name='"+nama+"'"
18.            cursor.execute(sql)
19.            res = cursor.fetchall()

```

#### 4.2.10 Ubah Password

Halaman ubah *password* digunakan untuk mengubah password dosen lama ke *password* baru sesuai yang gampang di ingat. halaman ubah *password* dapat dilihat pada gambar 4.10.

**Gambar 4.10** Ubah Password

Berikut potongan kode untuk menampilkan halaman ubah password.

```

1. @app.route('/ubah_password')
2. def ubah_password():
3.     session.pop('username', None)
4.     if session.get('npp'):
5.         openDb()
6.         container = []
7.         ceksql = "SELECT * FROM clustering_dosen where
npp='"+session.get('npp')+"'"
8.         cursor.execute(ceksql)
9.         cekresults = cursor.fetchall()
10.        for data in cekresults:
11.            container.append(data)
12.        closeDb()
13.        return render_template("ubahpassword.html", container=container)

```

### 4.3 PENGUJIAN SISTEM

Pengujian yang akan dilakukan dengan metode *black box*. Pengujian *black box* menitik beratkan pada fungsionalitas sistem. Metode ini digunakan untuk mengetahui apakah sistem berfungsi dengan benar.

#### 4.3.1 Tabel Pengujian *Login* Pegawai

Hasil pengujian dari *login* pegawai dapat dilihat pada Tabel 4.1.

**Tabel 4.1** Tabel Pengujian *Login* Pegawai

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Button <i>Login</i> , memasukan <i>username</i> dan <i>password</i> pada form yang disediakan.	Melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang valid	Masuk ke halaman utama pegawai	OK
2	Button <i>Login</i> , memasukan <i>username</i> dan <i>password</i> pada form yang disediakan.	Melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang tidak valid	Tetap di halaman login dengan di tunjukan <i>alert</i> "username atau password salah" dan tidak masuk ke halaman utama pegawai.	OK

#### 4.3.2 Tabel Pengujian *Logout* Pegawai

Pengujian dari fitur *logout* pegawai telah dilakukan dan dijelaskan pada Tabel 4.2.

**Tabel 4.2** Tabel Pengujian *Logout* Pegawai

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Button <i>Logout</i>	Mengklik logo user di kanan atas dan memilih tombol <i>logout</i> .	Berhasil kembali ke halaman <i>login</i> .	OK

#### 4.3.3 Tabel Pengujian *Login* dan *Logout* Dosen

Ujicoba terhadap fitur *login* dosen sudah dilakukan dan dijelaskan pada Tabel 4.3 dibawah.

**Tabel 4.3** Tabel Pengujian Login Dosen

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Button <i>Login</i> , memasukan <i>username</i> dan <i>password</i> pada form yang disediakan.	Melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang valid	Masuk ke halaman utama Dosen	OK
2	Button <i>Login</i> , memasukan <i>username</i> dan <i>password</i> pada form yang disediakan.	Melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang tidak valid	Tetap di halaman login dengan di tunjukan <i>alert</i> "username atau password salah" dan tidak masuk ke halaman utama dosen.	OK
3	Button <i>Logout</i>	Mengklik logo user di kanan atas dan memilih tombol <i>logout</i> .	Berhasil kembali ke halaman <i>login</i> .	OK

#### 4.3.4 Tabel Pengujian Input Data Penilaian

Fitur input data penilaian telah dilakukan pengujian dengan hasil seperti pada Tabel 4.4.

**Tabel 4.4** Tabel Pengujian Input Data Penilaian

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Menu input data penilaian	Memilih menu input data penilaian, memasukan parameter-parameter yang sudah disediakan	Penilaian berhasil di tambahkan dan dosen bisa melakukan pengecekan melalui dashboardnya	OK

#### 4.3.5 Tabel Pengujian Tambah Data Dosen

Menu tambah data dosen telah dilakukan pengujian dengan hasil pengujian akan dijelaskan pada Tabel 4.5.

**Tabel 4.5** Tabel Pengujian Tambah Data Dosen

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Menu tambah data dosen	Menambahkan data dosen per prodi dan menambah akun untuk login dosen.	Data dosen sudah tersimpan, dan dosen berhasil melakukan <i>login</i> dengan <i>username</i> dan <i>password</i> yang sudah dibuat	OK

#### 4.3.6 Tabel Pengujian Input Matakuliah

Proses pengujian input matakuliah telah dilakukan dan hasilnya dapat dilihat pada Tabel 4.6.

**Tabel 4.6** Tabel Pengujian Input Matakuliah

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Input matakuliah	Menambahkan mata kuliah yang sudah disediakan perprodi dan kode matakuliah.	Matakuliah telah tersimpan pada menu data matakuliah dan bisa diakses juga melalui menu tambah data dosen	OK

#### 4.3.7 Tabel Pengujian Ubah Password

Pada halaman dashboard untuk dosen juga disediakan fitur ubah password. Fitur ubah password telah dilakukan pengujian dan hasilnya bisa dilihat pada Tabel 4.7.

No	Fungsi yang di uji	Cara pengujian	Hasil yang diharapkan	Keterangan
1	Ubah Password	Dosen harus memasukan <i>password</i> lama yang sudah dibuatkan pegawai untuk identifikasi, lalu dosen diminta untuk memasukan <i>password</i> yang baru	<i>Password</i> sudah terubah di <i>database</i> dan saat login kembali dosen sudah bisa menggunakan <i>password</i> barunya.	OK

#### 4.4 PEMBAHASAN

Sistem yang dibuat terdiri dari dua aktor yaitu administrator atau karyawan dan user atau dosen. Karyawan bertanggung jawab untuk memasukkan dan menyimpan data dalam sistem, dan instruktur bertanggung jawab untuk menggabungkan data dari apa yang diunggah karyawan untuk ditinjau.

##### 4.4.1 Kekurangan

1. Belum bisa mencetak hasil laporan dari data penilaian.
2. Sistem tidak memiliki proses penyimpanan laporan.
3. Tidak adanya integrasi antara sistem dengan portal akademik di FTII unjani Yogyakarta.

##### 4.4.2 Kelebihan

1. Data yang diinputkan sudah tersimpan ke database sehingga jauh lebih aman.
2. Dengan adanya sistem penilaian ini dosen bias mengukur hasil dari kinerjanya untuk mendapatkan rekomendasi yang kedepannya lebih baik.
3. Memudahkan pengecekan data penilaian kinerja dosen.