
Laporan Dokumentasi Aplikasi

Tugas UAS Semester 3

Dosen : Mohamad Firdaus, M.Kom

Nama : Ardiyansyah

NPM : 623C0009

1. Deskripsi Aplikasi dan Tema

Aplikasi ini adalah aplikasi formulir pengajuan untuk pemohon yang membutuhkan layanan perizinan, dengan tema "Pengelolaan Layanan Perizinan Digital". Mekanisme aplikasi ini adalah tampilan awal adalah login dan register untuk pengguna dengan menggunakan validasi username tidak boleh sama, enkripsi password hash bcrypt dengan menerapkan panjang minimal karakter 8 (**pengembangan selanjutnya**: mengandung huruf besar angka dan simbol) dan terhubung pada database. Kemudian setelah login berhasil pengguna akan diarahkan ke layar formulir. Pada class formulir menggunakan filechoosericonviewer dan file path yang diupload akan disimpan pada database (**pengembangan selanjutnya**: file path disimpan pada host, dan di database hanya menampilkan jalur). Ketika pengguna melakukan submit, inputan data formulir disimpan pada database.

2. Perancangan Desain Antarmuka (UI/UX)

Pada aplikasi ini, desain antarmuka menggunakan beberapa layout utama untuk menyusun elemen-elemen UI secara responsif dan interaktif:

- **BoxLayout**: Digunakan untuk mengatur layout vertikal dan horizontal pada elemen seperti tombol dan input.
- **GridLayout**: Digunakan untuk menyusun elemen-elemen seperti form input data dengan rapi.
- **FloatLayout**: Digunakan pada bagian-bagian dengan penataan elemen lebih fleksibel sesuai ukuran jendela aplikasi.
- **ScreenManager**: Digunakan untuk menangani aplikasi yang kemungkinan semakin kompleks.

Elemen-elemen interaktif yang digunakan:

- **Button**: Untuk interaksi pengguna (login, submit, pilih file).
 - **TextInput**: Untuk memasukkan data pengguna (username, password, email, dll).
 - **Spinner**: Untuk memilih opsi layanan.
 - **Label**: Untuk menampilkan informasi statis.
 - **Popup**: Untuk memberikan umpan balik atau error.
-

3. Implementasi Konsep Object-Oriented Programming (OOP)

Aplikasi ini mengimplementasikan tiga pilar utama OOP:

- **Encapsulation**: Setiap fungsionalitas aplikasi ditempatkan dalam kelas tertentu dengan atribut dan metode. Sebagai contoh, LoginScreen dan FormulirScreen adalah dua kelas yang memiliki metode untuk login dan pengelolaan formulir pengajuan.

```
class LoginScreen(BoxLayout):
```

```
    def login(self):
```

```
        username = self.ids.username.text
```

```
        password = self.ids.password.text
```

```
        if username == "admin" and password == "admin":
```

```
            App.get_running_app().switch_to_form()
```

```
        else:
```

```
popup = Popup(title="Login Error", content=Label(text="Username atau Password salah!"))
```

```
popup.open()
```

- **Inheritance:** Kelas utama App digunakan sebagai induk dari semua layar (screen) yang diatur dalam aplikasi. Misalnya, kelas LoginScreen dan FormulirScreen turunan dari BoxLayout untuk menangani layout dasar dan elemen UI.

```
class FormulirScreen(BoxLayout):
```

```
    def update_layanan(self, permohonan):
```

```
    ...
```

- **Polymorphism:** Metode yang sama digunakan untuk berinteraksi dengan berbagai jenis form input dan tombol dengan parameter berbeda. Metode submit_data() digunakan untuk mengirim data, meskipun aplikasi bisa mengubah logika tergantung pada jenis data yang diinputkan oleh pengguna.

```
def submit_data(self):
```

```
    popup = Popup(title="Info", content=Label(text="Data berhasil dikirim!"))
```

```
    popup.open()
```

4. Fitur CRUD (Create, Read, Update, Delete)

- **Create:** Pengguna dapat mengisi data dalam form dan mengirimnya untuk disimpan.
- **Read:** Pengguna dapat melihat data mereka melalui pratinjau yang disediakan sebelum pengiriman.
- **Update:** Pengguna dapat memilih layanan atau memperbarui file yang diunggah pada formulir.
- **Delete:** Fitur ini belum diimplementasikan dalam aplikasi, tetapi dapat ditambahkan di masa depan dengan menambahkan metode untuk menghapus data yang telah diajukan.

Berikut adalah **workflow aplikasi** yang menggambarkan alur penggunaannya:

Workflow Aplikasi Formulir Permohonan Layanan

1. Login Screen (Layar Login)

- **Deskripsi:** Pengguna membuka aplikasi dan diminta untuk login dengan memasukkan **username** dan **password**.
- **Aksi:**
 - Jika **username** dan **password** yang dimasukkan sesuai (misalnya, admin dan admin), aplikasi akan melanjutkan ke layar formulir pengajuan.
 - Jika **username** atau **password** salah, aplikasi akan menampilkan **popup error** dengan pesan "Username atau Password salah!".
 - Jika pengguna memilih opsi **Register**, aplikasi akan menampilkan **popup info** dengan pesan "Fitur register belum aktif".
- **Diagram Workflow:**
- [User] --(enter username and password)--> [Login Screen]
- [If username and password are correct]--> [Go to Formulir Screen]
- [If username or password is incorrect]--> [Show Error Popup]

2. Formulir Permohonan Layanan (Formulir Screen)

- **Deskripsi:** Setelah login berhasil, pengguna diarahkan ke halaman formulir pengajuan layanan.
- **Aksi:**
 - **Permohonan:** Pengguna memilih jenis permohonan yang diinginkan (misalnya, "Legalitas Perusahaan", "Jasa NIB & OSS", dll.) melalui **Spinner**.
 - **Layanan:** Berdasarkan jenis permohonan yang dipilih, aplikasi akan menampilkan produk layanan yang sesuai di dalam **Spinner** (misalnya, untuk

"Legalitas Perusahaan", produk layanan seperti "Pendirian PT" dan "Pendirian CV").

- **Data Pribadi:** Pengguna memasukkan data pribadi seperti **Nama**, **Email**, dan **No HP** melalui **TextInput**.
- **File Upload:** Pengguna memilih file untuk diunggah menggunakan dua tombol **Pilih File 1** dan **Pilih File 2**. Aplikasi menampilkan nama file yang dipilih.
- **Preview:** Pengguna dapat melihat pratinjau data yang dimasukkan beserta file yang dipilih. Aplikasi menampilkan data dalam **Popup Preview**.
- **Submit:** Pengguna mengirimkan data melalui tombol **Submit**. Aplikasi akan menampilkan **Popup Info** dengan pesan "Data berhasil dikirim!".

- **Diagram Workflow:**

- [User] --(choose permohonan)--> [Update Layanan Options]
- [User] --(fill name, email, phone)--> [Fill Data Fields]
- [User] --(upload file)--> [Select File 1, File 2]
- [User] --(preview data)--> [Popup Preview Data]
- [User] --(submit data)--> [Show Submit Success Popup]

3. File Upload

- **Deskripsi:** Pengguna dapat memilih file untuk diunggah.

- **Aksi:**

- **Pilih File 1 dan File 2:** Pengguna memilih file menggunakan **FileChooserIconView**. Jika file dipilih, nama file akan ditampilkan di tombol.
- Jika pengguna memilih file, aplikasi akan menampilkan nama file yang dipilih di tombol **Pilih File 1** dan **Pilih File 2**.

- **Diagram Workflow:**

- [User] --(click Choose File 1)--> [FileChooser Icon View]
- [User] --(choose a file)--> [File Selected]
- [File Name displayed on button]--> [File1 or File2 updated]

Kesimpulan Workflow Aplikasi:

1. Pengguna memulai dengan layar login untuk memverifikasi identitas.
2. Setelah login berhasil, pengguna mengisi formulir permohonan dengan memilih jenis permohonan, produk layanan, dan memasukkan data pribadi.
3. Pengguna mengunggah dokumen yang diperlukan untuk pengajuan, mempratinjau data, dan kemudian mengirimkannya.
4. Setelah pengiriman, aplikasi memberikan konfirmasi bahwa data telah berhasil dikirimkan.

5. Dokumentasi dan Refleksi

Tantangan yang Dihadapi dan Solusi yang Diterapkan:

- **Tantangan:** Mengintegrasikan file chooser untuk memilih file PDF tidak berjalan sebagaimana mestinya karena masalah dalam menangani event file yang dipilih.

Solusi: Menerapkan perbaikan dengan menambahkan penanganan file secara eksplisit melalui **FileChooserIconView**, serta memastikan logika pengambilan file tepat dengan menggunakan objek **filechooser** secara benar.

Debug: Masih dalam proses debug, karena pemahaman tentang **filechooser** belum mendalam, dan sedang dalam tahap pengembangan untuk menerapkan file tersebut disimpan pada jalur (file path) host, dan pada database menampilkan file path.

- **Tantangan:** Menyusun login yang dapat menghubungkan aplikasi ke database tidak sepenuhnya selesai karena masalah dengan library yang digunakan.

Solusi: Fokus pada antarmuka dan logika aplikasi terlebih dahulu, dan merencanakan pengintegrasian database dalam versi berikutnya, yaitu penggunaan Screen Manager dan pemisahan beberapa class python untuk meminimalisir konflik error.

Kesimpulan dan Refleksi Pembelajaran: Pengembangan aplikasi ini memperkenalkan saya pada penggunaan modul-modul Kivy untuk membuat aplikasi dengan antarmuka grafis yang interaktif dan berbasis objek. Saya sedang dalam tahap pembelajaran bagaimana merancang antarmuka pengguna yang mudah digunakan, serta cara menerapkan prinsip OOP dalam pengembangan aplikasi untuk memisahkan logika dan tampilan. Meskipun masih ada beberapa tantangan dalam pengintegrasian fitur seperti koneksi database, upload file, dan pemahaman logika serta modul-modul lainnya, untuk menciptakan aplikasi yang sesuai harapan.
