

K21 RAILWAY ACCESS

2023

PROJECT TEAM

PRAKTIKUM PROGRAMMA KOMPUTER



KELOMPOK 21

- | | |
|----------------------------------|------------|
| 1. MUHAMAD FAHRIZA FAADHILAH | (10322079) |
| 2. MUHAMMAD YUSUF HABIBIE | (10322087) |
| 3. MUSYAFFA ABIDHAN RAHMAT PUTRA | (10322088) |
| 4. MUTIARA ANJELI ALDRIN | (10322089) |
| 5. NADYA KARTIKA APTARINI | (10322091) |





KATA PENGANTAR

Puji syukur kami ucapkan kehadiran Allah SWT atas segala rahmat-Nya sehingga kami dapat menyelesaikan Laporan Praktikum Program Komputer ini dengan lancar dan sesuai dengan jadwal yang telah ditentukan. Laporan ini dibuat untuk memenuhi tugas mata kuliah Praktikum Program Komputer pada program studi Teknik Industri Universitas Sebelas Maret Surakarta.

Penyusunan laporan ini tentu tidak luput dari dukungan banyak pihak, untuk itu kami ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang telah membantu kami dalam menyusun laporan ini sehingga dapat selesai dengan tepat waktu, terutama kepada:

1. Bapak Dr. Eko Liquiddanu S, T., M.T. selaku Kepala Program Studi Teknik Industri.
2. Bapak I Wayan Suletra M.T selaku Dosen pengampu Praktikum Program Komputer Tahun 2023 yang telah memberikan dukungan kepada kami.
3. Saudara Berlian Safri Prakoso, Halidya Siti Hanifah, Nurki Putra Mahardika, Pravanasta Rian Setiawan, Rizky Ammar Surya Saputra, Talitha Pranastiti, Aji Manarul Aziz, Alica Ayu Fitriani, serta Dendy Halim Kusuma selaku Asisten Laboratorium Perancangan dan Optomasi Sistem Industri (POSI) 2020 yang telah memberikan bimbingan kepada kami mahasiswa Teknik Industri UNS Angkatan 2022.
4. Teman-teman kelompok 21 yang sudah bekerjasama dengan baik untuk menyelesaikan laporan ini sehingga dapat menyelesaikan laporan praktikum kelompok dengan baik.

Dalam penyusunan laporan ini, kami menyadari bahwa hasil Laporan Praktikum Program Komputer masih jauh dari kata sempurna. Untuk itu kami sangat mengharapkan kritik dan saran membangun dari pembaca sekalian. Kami berharap dengan adanya laporan ini dapat menambah wawasan dan juga memberikan manfaat bagi kita semua.

Surakarta, 17 Juni 2023

Penulis





DAFTAR ISI

HALAMAN JUDUL	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	iv
BAB I DESKRIPSI MASALAH.....	K21-1
BAB II FLOWCHART PEMROGRAMAN	K21-3
BAB III KODE PROGRAM PYTHON.....	K21-5
BAB IV HASIL RUNNING PYTON.....	K21-9
BAB V PENUTUP.....	K21-13
5.1 Kesimpulan	K21-13





DAFTAR GAMBAR

BAB II : FLOWCHART PEMROGRAMAN

Gambar 2.1 Flowchart..... K21-5

BAB III : KODE PROGRAM PYTHON

Gambar 3.1 Fungsi *login*..... K21-7

Gambar 3.2 Fungsi Membuat Akun K21-7

Gambar 3.3 Fungsi *Main Page*..... K21-8

Gambar 3.4 Fungsi *Schedul Page*..... K21-8

Gambar 3.5 Fungsi Isi Identitas..... K21-9

Gambar 3.6 Fungsi *Payment Page*..... K21-9

Gamabr 3.7 Fungsi *Ticket Page*..... K21-10

BAB IV : HASIL RUNNING PYTHON

Gambar 4.1 Halaman *Login* K21-11

Gambar 4.2 Halaman *Sign Up*..... K21-11

Gambar 4.3 Halaman Main Page K21-12

Gambar 4.4 Halaman *Schedule Page* K21-12

Gambar 4.5 Halaman Isi Identitas..... K21-13

Gambar 4.6 Halaman Pembayaran..... K21-13

Gamabr 4.7 Halaman *Ticket Page*..... K21-14



BAB I

DESKRIPSI MASALAH

Dalam pengembangan program pemesanan tiket kereta api ini, kami mengidentifikasi beberapa masalah yang perlu diatasi dari program yang telah ada sebelumnya. Pertama, penggunaan kertas dalam mencetak tiket fisik menjadi masalah serius terkait keberlangsungan lingkungan. Setiap tahun ribuan tiket kereta dicetak hingga menghasilkan banyak tumpukan limbah kertas. Selain masalah lingkungan, penggunaan tiket fisik juga menimbulkan kendala bagi pengguna. Pengguna harus mencetak tiket sendiri atau mengambilnya di loket yang dapat memakan waktu dan tenaga. Selain itu, resiko kehilangan tiket fisik yang dapat menyebabkan kesulitan dalam mengakses dan menggunakan tiket saat bepergian.

Selain masalah penggunaan kertas, berdasarkan analisa kami program pemesanan tiket kereta api yang ada saat ini tidak memberikan tampilan yang optimal dan memadai pada desktop. Pengguna menghadapi kendala dalam melihat informasi tiket dengan jelas dan mudah dibaca. Tampilan yang terbatas dan tidak dioptimalkan pada program tersebut seringkali menyebabkan informasi yang penting terpotong atau sulit terbaca. Hal ini dapat mengakibatkan kesulitan dalam memahami rincian tiket, seperti jadwal, rute perjalanan, harga, dan ketersediaan tempat duduk. Maka dari itu, kami mengembangkan program pemesanan tiket kereta api yang tidak hanya mengurangi penggunaan kertas melalui *e-ticket*, tetapi juga memberikan tampilan yang lebih besar pada *desktop* dengan tujuan memudahkan pengguna dalam memesan tiket.

Dengan memperkenalkan solusi *e-ticket* melalui program pemesanan tiket kereta api ini, kami bertujuan untuk mengurangi dampak negatif terhadap lingkungan. Melalui penggunaan *e-ticket*, pengguna dapat menghindari pencetakan tiket fisik dan dengan mudah mengaksesnya melalui perangkat elektronik mereka. Solusi cetak *e-ticket* juga memberikan kemudahan bagi pengguna karena pengguna dapat mengakses tiket secara langsung serta memiliki akses yang mudah untuk melakukan perjalanan. Dengan demikian, pengembangan program pemesanan tiket kereta api yang mengutamakan *e-ticket* dapat memberikan manfaat signifikan dalam hal keberlangsungan lingkungan dan kenyamanan pengguna.

Disamping itu kami mengembangkan program ini dengan fokus pada tampilan yang lebih besar pada desktop. Program ini dirancang dengan mempertimbangkan ukuran layar yang lebih besar, sehingga informasi tiket dapat ditampilkan secara menyeluruh. Dengan begitu, pengguna akan lebih mudah dalam melihat jadwal kereta, rute perjalanan, dan harga tiket.





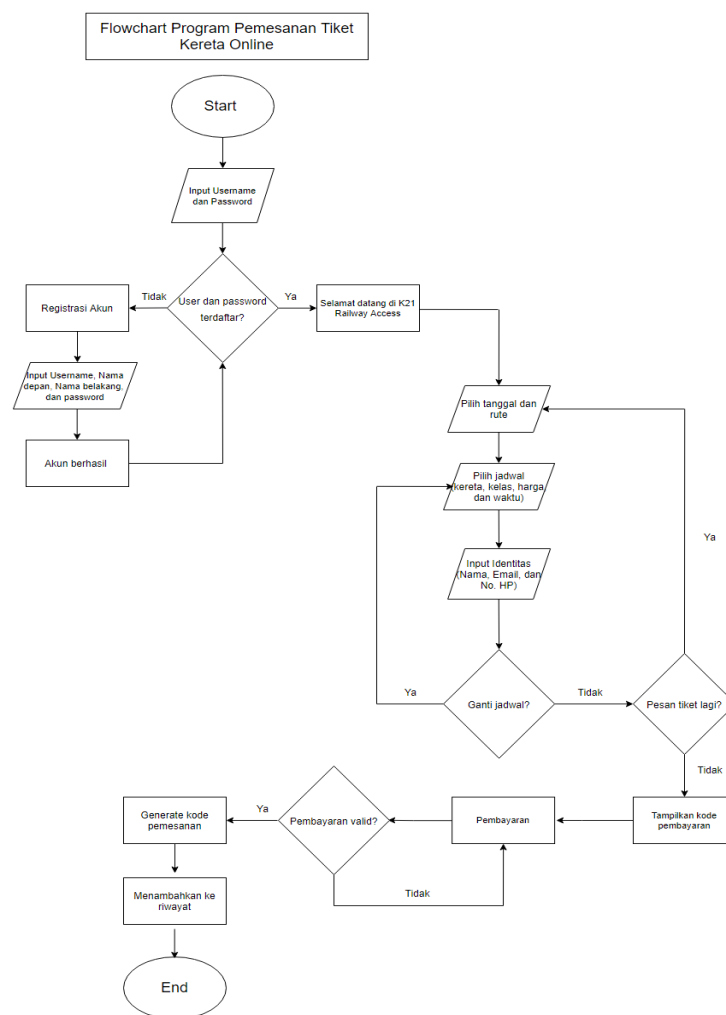
Tampilan *full screen* akan memastikan bahwa pengguna dapat melihat semua detail tiket serta meningkatkan kenyamanan dan pengalaman pengguna dalam melakukan pemesanan. Dengan mengatasi masalah tampilan yang kurang optimal pada program pemesanan tiket kereta api yang ada saat ini, kami berharap dapat memberikan solusi dalam memenuhi kebutuhan pengguna. Kami juga berusaha untuk meningkatkan kemudahan penggunaan program pemesanan tiket kereta api ini. Kami membuat desain program yang interaktif serta mudah dipahami, sehingga pengguna dapat dengan mudah memilih jadwal kereta, kelas pelayanan, dan harga tiket. Kami berharap dengan pengembangan yang kami buat pada program pemesanan tiket kereta api dapat meningkatkan pengalaman pengguna.



BAB II

FLOWCHART PEMROGRAMAN

Bab ini menjelaskan mengenai diagram alir atau *flowchart* dari program kami yaitu “K21 Railway Acces”. Pada *flowchart* tersebut, digunakan beberapa simbol untuk menjelaskan keberlangsungan program. Pembuatan *flowchart* ini dilakukan dalam rangka pemenuhan tugas Praktikum Program Komputer.



Pada program tersebut, diagram alir diawali dengan terminator yang menunjukkan bahwa program telah dimulai. Selanjutnya dilakukan suatu proses dimana pengguna diminta untuk meng-*input*-kan data berupa *username* dan *password*. Tahap ini menggunakan simbol *input-output*. Setelah data di-*input*-kan, dilakukan *decision* pemilihan proses berdasar kondisi yang ada. Apabila pengguna telah memiliki akun sebelumnya, maka akan muncul di halaman utama dan pengguna akan diarahkan menuju menu selanjutnya. Namun apabila pengguna belum memiliki akun, maka akan diarahkan untuk membuat akun (iterasi). Tahap selanjutnya adalah



proses pemilihan tanggal, rute, jadwal, dan *input* identitas. Pada tahap ini menggunakan simbol *input-output*. Kemudian kami menggunakan simbol *decision* untuk meyakinkan pengguna bahwa data yang telah informasi yang telah di-*input*-kan pada tahap sebelumnya sudah benar atau belum. Apabila data dirasa sudah benar, maka pengguna akan diarahkan menuju ke *decision* selanjutnya. Namun apabila data yang di-*input*-kan dirasa masih ada kesalahan, pengguna dapat mengubahnya dengan kembali ke tahap sebelumnya. *Decision* selanjutnya adalah sistem akan menanyakan kepada pengguna mengenai tiket. Ingin melakukan pemesanan tidak lagi atau tidak. Apabila ingin melakukan pemesanan lagi, maka pengguna akan diarahkan menuju tahap pemilihan tanggal, rute, jadwal, dan *input* identitas. Apabila tidak ingin melakukan pemesanan lagi, pengguna diarahkan menuju proses pembayaran. Pada tahap ini, digunakan simbol proses yang mana sistem akan melakukan data atau informasi. Pengguna akan memperoleh kode pembayaran yang nanti digunakan saat pembayaran tiket. Kemudian akan dilakukan verifikasi atau pengondisian apakah pembayaran tersebut valid atau tidak. Apabila valid, maka pengguna dapat menuju tahap *generate* kode pemesanan dan kemudian akan ditambahkan ke riwayat pembelian pengguna. Namun apabila pembayaran tidak valid, maka pengguna akan diarahkan untuk mengulangi (iterasi) proses pembayaran hingga pembayaran dilakukan secara valid. Kemudian pengguna dapat melakukan proses *generate* kode pembayaran.





BAB III

KODE PROGRAM PYTHON

Bab ini menjelaskan mengenai kode program python yang kami gunakan dalam program kami yaitu “K21 Railway Acces”. Pada program tersebut, digunakan beberapa fungsi, sebagai berikut:

```
def loginAkun():  
    global login  
    global tgl_banding  
    tgl_banding = ""  
    login = Tk()  
    login.title("K21 Railway Access")  
    login.configure(bg="#fff")  
    login.geometry("920x500+170+80")  
    login.resizable(False, False)  
  
    #####
```

Gambar 3.1 Fungsi *Login*

Pada awal program, terdapat fungsi *loginAkun()* yang berfungsi untuk tampilan awal dari program. Selanjutnya, pada bagian ini *users* diminta melakukan *login* (apabila sudah memiliki akun) atau registrasi akun (apabila belum memiliki akun). Setelah berhasil *login* atau registrasi akun, *Users* dapat melanjutkan program.

```
def bikinAkun():  
    global regis  
    regis = Tk()  
    regis.title("K21 Railway Access")  
    regis.geometry("920x500+170+80")  
    regis.configure(bg="#fff")  
    regis.resizable(False, False)  
  
    #####  
  
    frame = Frame(regis, width=350, height=350, bg="#fff")  
    frame.place(x=285, y=40)  
  
    #####
```

Gambar 3.2 Fungsi Membuat Akun





Pada bagian fungsi *bikinAkun()*, *users* akan diminta membuat akun dengan menginput nama dan *password* akun. Selanjutnya, akun akan tersimpan di data json dan *users* dapat melakukan *sign in*.

```
def mainPage():
    global main
    global Asal
    global Tujuan
    global Tanggal_pilih
    global banyakPenumpang
    system("cls")
    main = Tk()
    main.geometry("920x500+170+80")
    main.resizable(False, False)
    main.configure(bg="#dfdfde")
    main.title("K21 Railway Access")
```

Gambar 3.3 Fungsi *Main Page*

Pada fungsi *mainPage()*, program akan meminta *users* untuk memilih tempat asal, tempat tujuan, tanggal berangkat dan banyak penumpang. Kemudian, *Users* dapat melanjutkan ke program mencari jadwal.

```
def schedulPage():
    global schedul
    schedul = Toplevel()
    schedul.geometry("920x500+170+80")
    schedul.resizable(False, False)
    schedul.title("K21 Railway Access")
    schedul.configure(bg="#fff")
    style = ttk.Style()
    style.theme_use('default')
    style.configure("Vertical.TScrollbar", background="#57a1f8", bordercolor="#000000")

    wrapper = CTKFrame(schedul)
    wrapper.pack(side=BOTTOM, fill=BOTH, expand=YES, pady=30, padx=250)
```

Gambar 3.4 Fungsi *Schedul Page*

Pada Fungsi *schedulPpage()*, Program akan menampilkan pilihan kereta sesuai dengan tujuan dan tanggal berangkat sesuai yang dipilih *users* sebelumnya. Lalu, *users* diminta memilih kereta yang *users* inginkan.





```
def isiIdentitas1():  
    global isi  
    global nama_user  
    isi = Toplevel()  
    isi.geometry("920x500+170+80")  
    isi.resizable(False, False)  
    isi.configure(bg="#dfdfde")  
    isi.title("K21 Railway Access")
```

Gambar 3.5 Fungsi Isi Identitas

Pada fungsi *isiIdentitas()*, program menampilkan identitas yang perlu diisi. *Users* diminta mengisi identitas berupa nama lengkap, email, nomor handphone, dan tipe penumpang. Kemudian, *users* dapat melanjutkan ke proses pembayaran.

```
def paymentPage():  
    global payment  
    global bayar  
    def addHistory():  
        kode = []  
        def Addjson(new_data, filename='purchase.history.json'):  
            with open(filename, 'r+') as file:  
                file_data = json.load(file)  
                file_data[Username].append(new_data)  
                file.seek(0)  
                json.dump(file_data, file, indent = 4)
```

Gambar 3.6 Fungsi *Payment Page*

Pada fungsi *paymentPage()*, program akan menampilkan nama kereta, asal pemberangkatan, tujuan, durasi perjalanan, tanggal berangkat, tanggal tiba, jumlah penumpang, dan total harga yang perlu dibayarkan sesuai *users* pilih sebelumnya. Kemudian, *users* diminta membayar jumlah tagihan yang perlu dibayarkan sesuai dengan harga tiket, setelah pembayaran berhasil maka otomatis data akan tersimpan.





```
def ticketPage():  
    global ticket  
    main.destroy()  
    ticket = Tk()  
    ticket.geometry("920x500+170+80")  
    ticket.resizable(False, False)  
    ticket.title("K21 Railway Access")  
    ticket.configure(bg="#fff")  
    style = ttk.Style()  
    style.theme_use('default')  
    style.configure("Vertical.TScrollbar", background="#57a1f8", bordercolor="#000000",
```

Gambar 3.7 Fungsi *Ticket Page*

Pada fungsi *ticketPage()*, program menampilkan tiket users setelah berhasil melakukan pembayaran. Tiket berisi kode pemesanan, status pembayaran, asal pemberangkatan, waktu berangkat, tempat tujuan, dan waktu tiba.

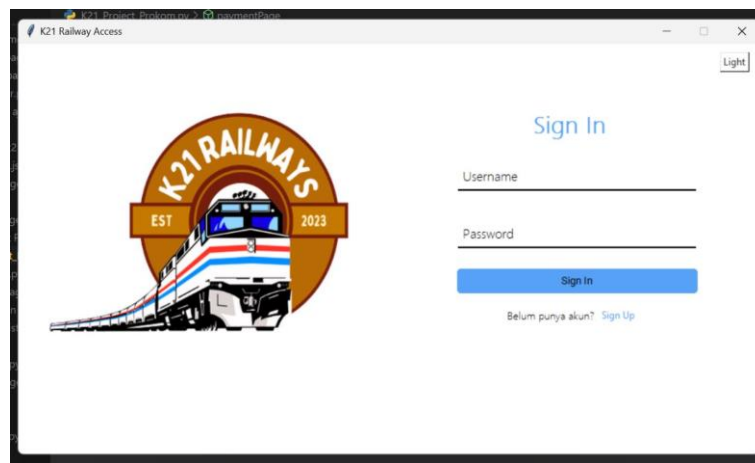




BAB IV

HASIL RUNNING PYTHON

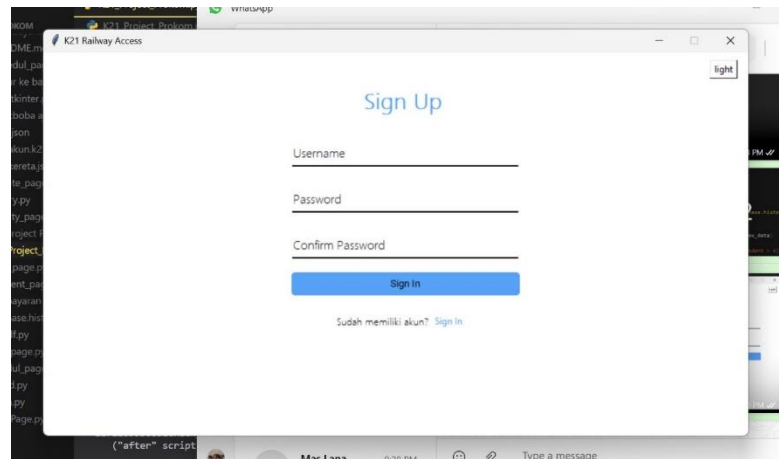
Bab ini menjelaskan mengenai hasil dari program kami yaitu “K21 Railway Acces”. Pada program tersebut, digunakan beberapa fungsi untuk keberlangsungan program ini. Pembuatan program ini dilakukan dalam rangka pemenuhan tugas Praktikum Program Komputer.



Gambar 4.1 Halaman *Login*

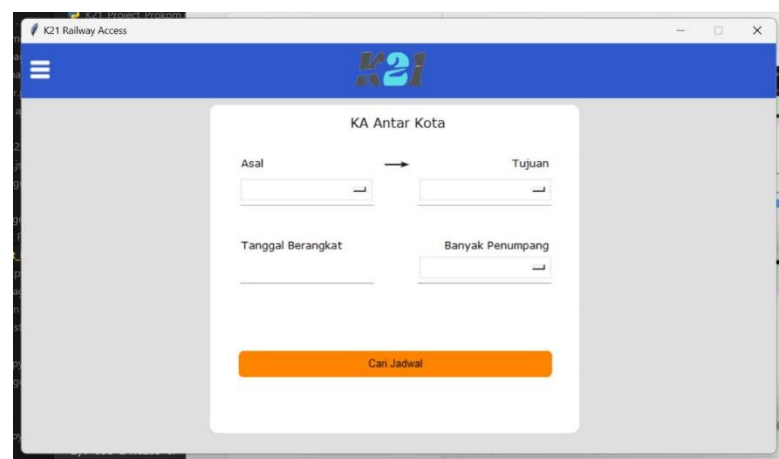
Gambar diatas merupakan hasil *running* fungsi *loginAkun()*. Gambar diatas menampilkan *username* dan *password* yang merupakan masukan dari pengguna. Apabila *username* dan *password* belum terdaftar, maka oleh sistem akan diarahkan untuk *sign in*. Setelah itu pengguna baru dapat *login* ke aplikasi.





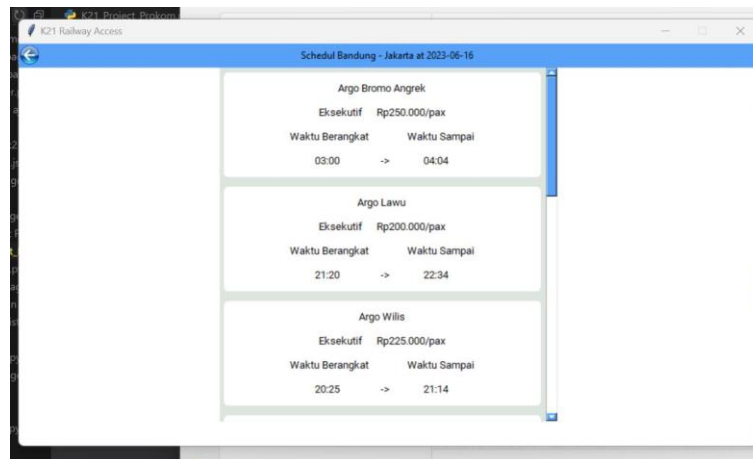
Gambar 4.2 Halaman *Sign Up*

Gambar diatas merupakan hasil *running* fungsi *bikinAkun()*. Gambar diatas menampilkan *username*, *password*, dan *confirm password*. Pengguna yang belum terdaftar perlu untuk melakukan pendaftaran pada fungsi ini agar dapat menggunakan program K21 Railway Acces.



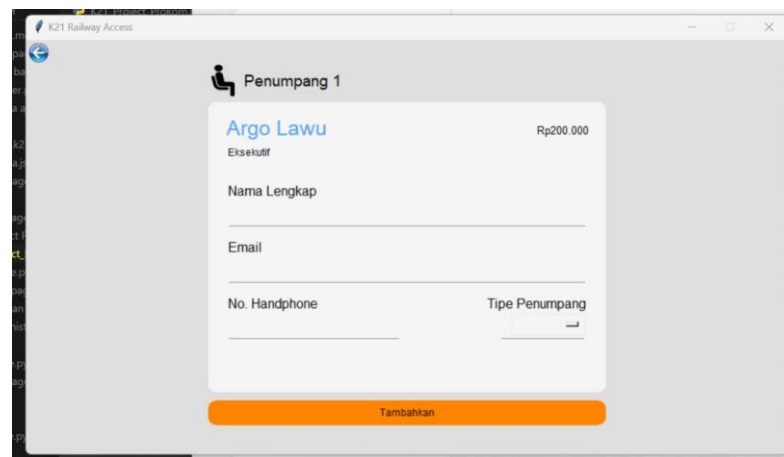
Gambar 4.3 Halaman Main Page

Gambar diatas merupakan hasil *running* fungsi *mainPage()*. Gambar diatas menunjukkan input pengguna mengenai asal keberangkatan dan tujuan akhir pada baris pertama. Pada baris kedua menunjukkan *input* tanggal keberangkatan dan banyak penumpang. Kemudian penumpang dapat melakukan pencarian jadwal.



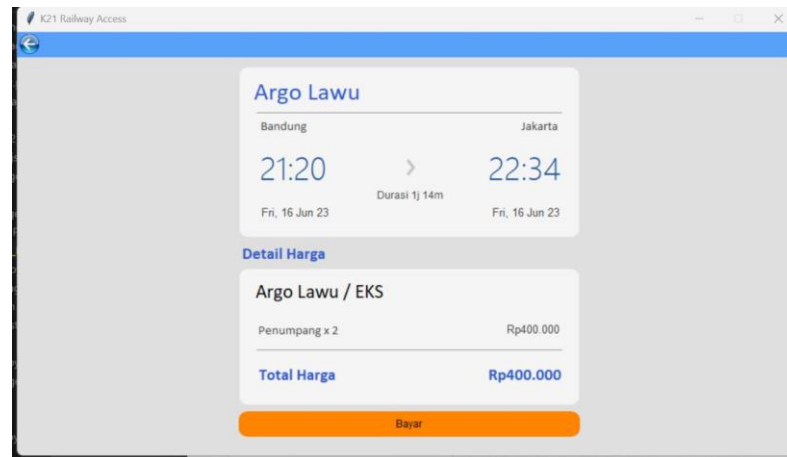
Gambar 4.4 Halaman *Schedule Page*

Gambar diatas merupakan hasil running fungsi *schedulePage()*. Pada gambar diatas terlihat jadwal beberapa kereta api sesuai dengan data yang telah di-*input*-kan pada fungsi sebelumnya. Pengguna dapat memilih jadwal sesuai dengan keinginan masing-masing.



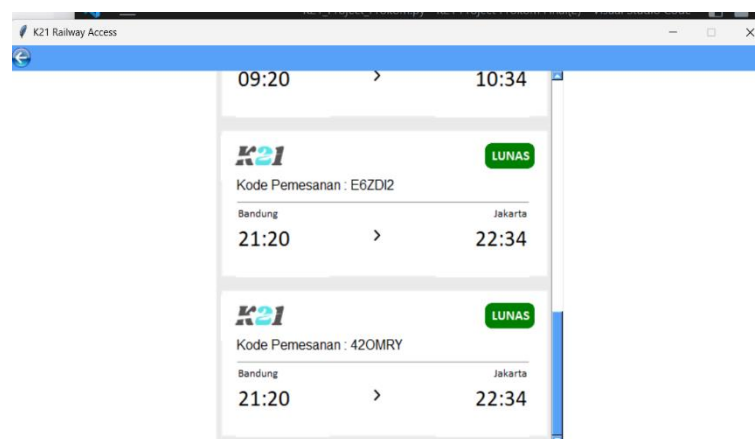
Gambar 4.5 Halaman fungsi identitas

Gambar diatas merupakan hasil dari fungsi *isiIdentitas()*. Pada gambar diatas, program menampilkan data atau identitas dari penumpang. Penumpang diarahkan untuk mengisi rubrik kosong tersebut. Setelah itu, pengguna memencet tombol tambahkan kemudian melakukan pembayaran.



Gambar 4.6 Halaman Pembayaran

Gambar diatas merupakan hasil running fungsi *paymentPage()*. Pada gambar diatas dapat dilihat nama kereta, asal pemberangkatan, tujuan, durasi perjalanan, tanggal berangkat, tanggal tiba, jumlah penumpang, dan total harga yang perlu dibayarkan.



Gambar 4.7 Halaman Ticket Page

Gambar diatas merupakan hasil running fungsi *ticketPage()*. Pada gambar tiket diatas terdapat kode pemesanan yang berbeda setiap tiket, asal pemberangkatan, tujuan pemberangkatan, jam berangkat, jam tiba dan status pembayaran.



BAB V

PENUTUP

5.1 Kesimpulan

Program pemesanan tiket kereta api “K21 Railways Access” ini didasarkan beberapa masalah yang perlu diatasi dari program yang ada sebelumnya. Masalah pertama, penggunaan kertas pada mencetak tiket fisik. Penggunaan kertas pada proses mencetak tiket menurut kami kurang ramah lingkungan. Masalah kedua, berdasarkan analisa kami, program pemesanan tiket kereta api yang ada saat ini tidak memberikan tampilan yang optimal dan memadai pada desktop.

Dengan program “K21 Railways Access”, kami memperkenalkan solusi *e-ticket* dalam program kami. Hal ini bertujuan untuk mengurangi dampak penggunaan kertas pada tiket kereta api. Selain itu, solusi *e-ticket* juga mempermudah pengguna untuk mengaksesnya dengan perangkat elektronik mereka. Disamping itu, Kami juga mengembangkan program ini fokus pada tampilan yang lebih besar, yaitu dapat ditampilkan pada dekstop.

Kami berharap adanya program kami “K21 Railways Access” dengan pengembangan dari program sebelumnya yang telah ada dapat meningkatkan pengalaman dan mempermudah pengguna dalam pemesanan tiket kereta api.

