



HOLIYAY

Laporan Project Team Kelompok 13

I0322039 / Diodorus Charis Alfiano
I0322047 / Flavia Fayyazza A A
I0322059 / Josephine Karin R Br K
I0322061 / Karisa Arfina Ratri

BAB I

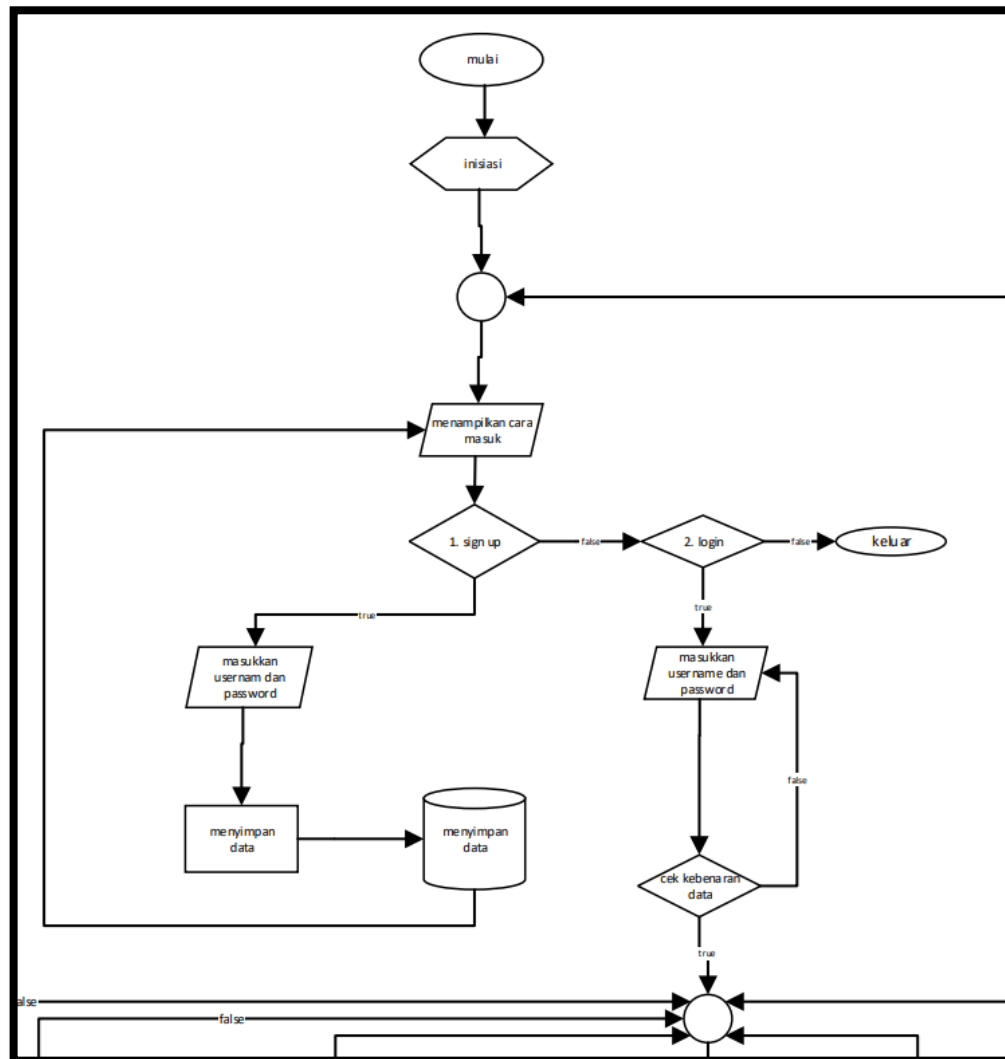
DESKRIPSI MASALAH

Kegiatan wisata merupakan hal yang umum dilakukan oleh hampir semua orang, bahkan banyak yang sangat menikmati aktivitas tersebut. Wisata dapat diartikan sebagai perjalanan bersama atau pergi keluar untuk tujuan rekreasi. Di Indonesia, wisata bukanlah sesuatu yang asing, karena negara ini memiliki banyak tempat tujuan wisata yang terkenal baik di dalam negeri maupun di luar negeri salah satunya Bali. Pariwisata Bali merupakan salah satu tujuan wisata yang sudah tidak diragukan lagi oleh wisatawan asing maupun wisatawan domestik. Pariwisata Bali sudah menjadi tujuan wisata dunia yang terkenal di seluruh manca negara.

Bali memiliki banyak sekali tempat tujuan wisata mulai dari wisata alam, bahari, sejarah, kuliner, maupun yang lainnya. Jumlah wisatawan dalam negeri pun dewasa ini mengalami peningkatan yang cukup pesat. Perkembangan teknologi yang pesat telah mengubah gaya hidup masyarakat pada masa kini, termasuk dalam kegiatan pariwisata. Data dari Badan Pusat Statistik (BPS) pada bulan April 2023 menunjukkan peningkatan jumlah wisatawan mancanegara yang mengunjungi Indonesia sebanyak 411.510 orang dibandingkan dengan April tahun 2022.

Secara keseluruhan, program pariwisata yang menyediakan layanan pemesanan hotel, destinasi wisata, dan pemandu wisata di Bali sangat bermanfaat karena menyediakan informasi yang lengkap, kemudahan akses, dan keamanan. Hal ini membantu wisatawan merencanakan perjalanan mereka dengan lebih baik, meningkatkan pengalaman mereka, dan memastikan mereka mendapatkan layanan yang sesuai dengan kebutuhan dan preferensi mereka. Secara keseluruhan, aplikasi wisata khusus untuk suatu daerah dapat meningkatkan visibilitas pariwisata, mempermudah akses informasi, dan meningkatkan pengalaman wisatawan, yang pada gilirannya dapat berkontribusi pada pertumbuhan sektor pariwisata dan ekonomi lokal.

BAB II FLOWCHART

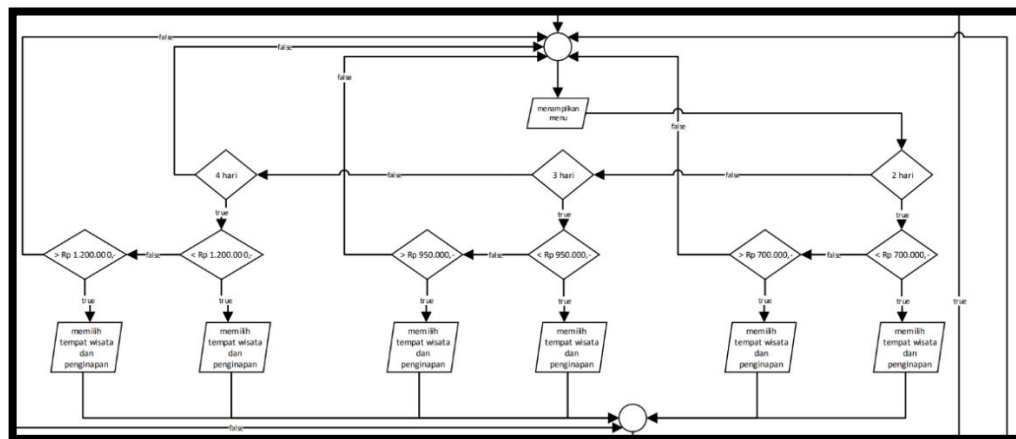


Gambar 2.1

Program ini diawali dengan terminator untuk memulai proses transaksi aplikasi holiyay. Selanjutnya terjadi proses inisiasi pengguna dengan menampilkan cara masuk. Pengguna diberikan dua pilihan, yaitu sign up bagi pengguna yang belum memiliki akun, atau login jika pengguna sudah memiliki akun.

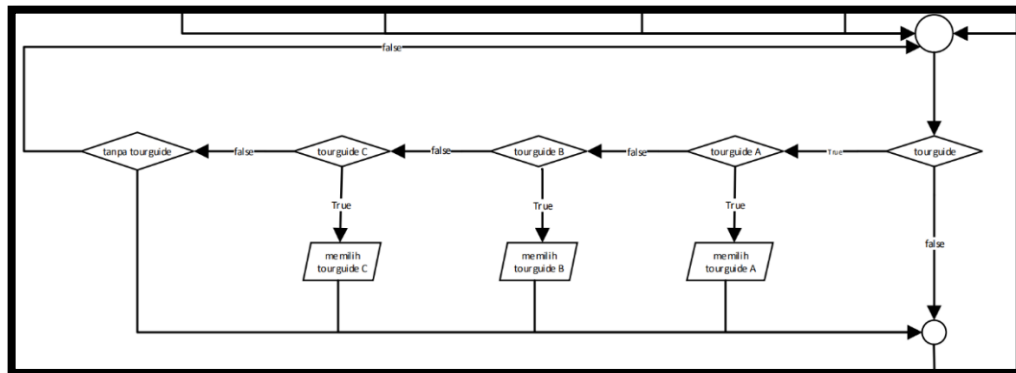
Pada proses sign up, pengguna diminta untuk memasukkan username dan password. Setelah itu, data pengguna akan disimpan di database. Setelah selesai sign up, pengguna akan diarahkan kembali ke tahap cara masuk. Karena pengguna sudah memiliki akun, pengguna dapat langsung melakukan login dengan menggunakan username dan password yang telah disimpan pada tahap sign up.

Jika pengguna memasukkan data dengan benar, aplikasi akan mengarahkan pengguna ke tampilan menu. Namun, jika pengguna memasukkan data dengan salah, pengguna akan diminta untuk melakukan login ulang.



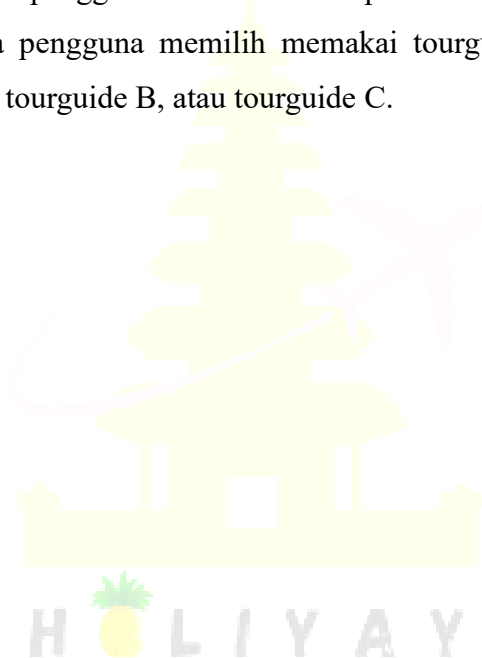
Gambar 2.2

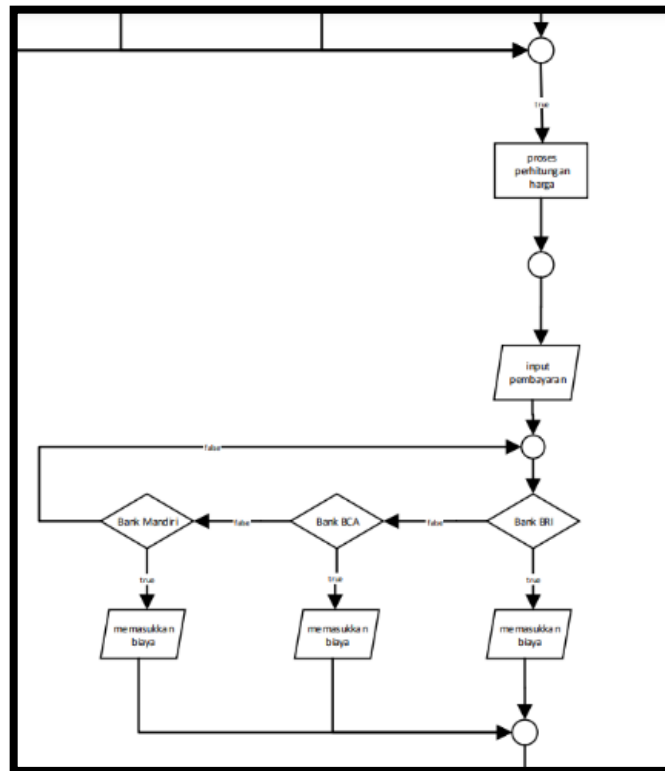
Pada bagian ini, pengguna masuk ke menu utama. Pengguna akan diberikan beberapa pilihan rentang waktu yaitu 2 hari, 3 hari, dan 4 hari. Setelah pengguna memilih rentang hari yang diinginkan, program akan memberikan pilihan budget sesuai jumlah hari yang dipilih. Jika pengguna memilih rentang waktu 2 hari maka pilihan budget terdiri dari kurang dari Rp700.000 atau lebih dari Rp700.000. Jika pengguna memilih rentang waktu 3 hari maka pilihan budget terdiri dari kurang dari Rp950.000 atau lebih dari Rp950.000. Jika pengguna memilih rentang waktu 4 hari maka pilihan harga terdiri dari Rp1.200.000 atau lebih dari Rp1.200.000. Setelah user menentukan pilihan rentang waktu dan pilihan budget, user dapat memilih tempat wisata dan penginapan sesuai dengan list yang ada.



Gambar 2.3

Pada bagian ini pengguna akan diberikan pilihan ingin memakai tourguide atau tanpa tourguide. Jika pengguna memilih memakai tourguide, maka pengguna dapat memilih tourguide A, tourguide B, atau tourguide C.

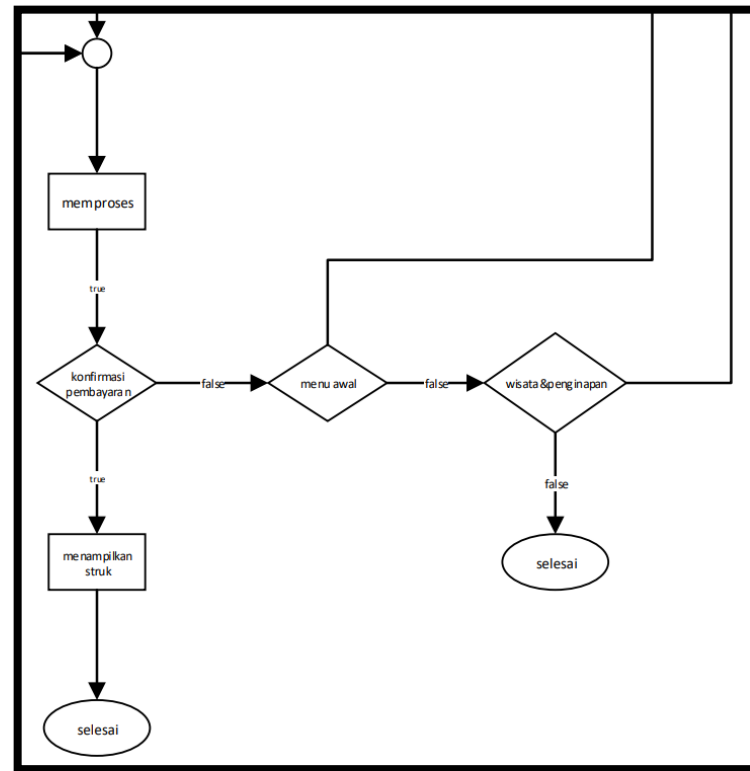




Gambar 2.4

Pada bagian ini, pilihan yang sudah masuk akan di proses untuk menghitung harga total. Kemudian pengguna dapat memilih metode pembayaran yang tersedia seperti bank BRI, bank BCA, maupun bank Mandiri. Setelah memilih salah satu dari bank tersebut, pengguna diminta memasukkan nominal biaya yang sesuai dengan perhitungan harga total.

HOLIYAY



Gambar 2.5

Pada bagian ini program akan memproses lalu mengkonfirmasi pembayaran. Jika pengguna menyetujui pembayaran, maka program akan mencetak struk pembayaran. Namun jika pengguna pengguna tidak menyetujui pembayaran, program akan kembali ke menu awal.

HOLIYAY

BAB III

PROGRAM DAN PENJELASAN

Pada Bab ini menjelaskan tentang kode program yang digunakan dalam program Holiyay.

A. Def menu_awal()

Bagian ini adalah fungsi untuk menampilkan pilihan masuk. Pada fungsi ini ditampilkan beberapa pilihan kepada pengguna dan menggunakan eksepsi untuk menanggulangi kesalahan *input* dari pengguna. Berikut adalah gambar dari kode program def menu_awal().

```
while True:
    try:
        print(
            '\nSilakan memilih opsi:
1. Sign up
2. Sign in
3. Keluar')
        pilih = input("Masukkan pilihan (1/2/3): ")
        if pilih == "1":
            sign_up()
            menu_awal()
        elif pilih == "2":
            sign_in()
            lama_libur()
            break
        elif pilih == "3":
            keluar()
        else:
            raise ValueError("Pilihan yang dimasukkan tidak valid, silakan pilih 1, 2, atau 3.")
    except ValueError as e:
        print("\n=====")
        print(f"{'#Kesalahan pada akun anda atau kesalahan input#':^60}")
        print(f"{str(e):^60}")
        print("=====")
        continue
menu_awal()
```

Gambar 3.1

Eksepsi pada **Gambar 3.1** menanggapi kesalahan pengguna berupa kesalahan memasukkan pilihan. Jika kesalahan tersebut terjadi, maka akan muncul seperti gambar

dibawah.

```
print("\n=====")
print(f"{'#Kesalahan pada akun anda atau kesalahan input#':^60}")
print(f"{str(e):^60}")
print("=====")
```

Gambar 3.2

Lalu diikuti oleh kalimat seperti gambar dibawah.

```
raise ValueError("Pilihan yang dimasukkan tidak valid, silakan pilih 1, 2, atau 3.")
```

Gambar 3.3

B. Def sign_up()

Apabila pengguna belum memiliki akun, maka pengguna dapat memilih pilihan “1” untuk mendaftarkan diri. Pilihan “1” akan membawa pengguna menuju def sign_up(). Berikut adalah kode program pada fungsi sign_up().

```
def sign_up():
    global username
    print("\n=====")
    username = input("Masukkan username: ")
    password = input("Masukkan password: ")
    print("=====")
    file = open("data_base.txt", "a")
    file.write(f"{username}:{password}\n")
    print(f"{'Sign up berhasil!':^28}\n")
```

Gambar 3.4

Penjelasan dari kode program di atas adalah sebagai berikut,

1. Global username digunakan agar username dapat dipanggil tanpa perlu memanggil fungsi sign_up().
2. Pengguna diminta untuk mengisi username dan password yang mana akan disimpan pada file txt Bernama “data_base.txt” dengan format penulisan adalah “username:password”.
3. Kemudian pengguna akan kembali ke menu awal seperti yang ditampilkan pada **Gambar 3.1**.

C. Def sign_in()

Apabila pengguna telah memiliki akun, maka pengguna dapat memilih pilihan “2” untuk masuk ke dalam program. Pilihan “2” akan membawa pengguna menuju def sign_in(). Berikut adalah kode program pada fungsi sign_in().

```
def sign_in():
    global username
    print("\n=====")
    username = input("Masukkan username: ")
    password = input("Masukkan password: ")
    print("=====")
    file = open("data_base.txt", "r")
    lines = file.readlines()
    for line in lines:
        stored_username, stored_password = line.strip().split(":")
        if username == stored_username and password == stored_password:
            print(f'{Sign in berhasil!':^28}")
            lama_libur()
            return
    print("\nUsername atau password salah.")
    sign_in()
```

Gambar 3.5

Penjelasan dari program di atas adalah sebagai berikut,

1. Global username digunakan agar username dapat dipanggil tanpa perlu memanggil fungsi sign_in().
2. Pengguna diminta untuk mengisi username dan password
3. Kemudian program akan membuka data_base.txt dan membaca semua baris yang ada pada file tersebut
4. Program akan mencari kecocokan data yang dimasukkan oleh pengguna dengan data yang telah tersimpan dalam data_base.txt
5. Jika data benar maka pengguna akan diarahkan menuju fungsi lama_libur()
6. Jika data salah maka akan muncul “Username atau password salah” dan pengguna diminta untuk mengisi kembali username dan password.

D. Def keluar()

Apabila pengguna tidak jadi menggunakan program Holiyay, maka pengguna dapat memilih pilihan “3”. Berikut adalah kode program pada fungsi keluar().

```
def keluar():
    print("\nTerima Kasih sudah menggunakan Holiyay")
    exit()
```

Gambar 3.6

Pada fungsi keluar(), berisi *output* seperti pada **Gambar 3.6** dan program akan keluar dengan fungsi exit() pada bawaan Python.

E. Def get_filtered_tempat_wisata_dan_hotel(cek_total_harga)

Fungsi ini berguna untuk mengambil daftar tempat wisata dan hotel yang memenuhi kriteria total harga tertentu. Berikut adalah penjelasan lebih detail mengenai program tersebut,

```
def get_filtered_tempat_wisata_dan_hotel(cek_total_harga):
    global tempat_wisata_dan_hotel_dict
    global tempat_wisata_list
    global hotel_list
    global filtered_tempat_wisata_list
    global filtered_hotel_list

    tempat_wisata_dan_hotel_dict = {}
    tempat_wisata_list = []
    hotel_list = []
    filtered_tempat_wisata_list = []
    filtered_hotel_list = []
```

Gambar 3.7

1. Menginisialisasi Variabel dan Daftar Kosong:

Membuat beberapa variabel global, seperti tempat_wisata_dan_hotel_dict, tempat_wisata_list, hotel_list, filtered_tempat_wisata_list, dan filtered_hotel_list, untuk menyimpan data yang diperlukan seperti pada **Gambar 3.7**.

```
with open("tempat_wisata.txt", "r") as file:
    tempat_wisata_reader = csv.reader(file, delimiter=',')
    next(tempat_wisata_reader)
```

Gambar 3.8

```
with open("hotel.txt", "r") as file:
    hotel_reader = csv.reader(file, delimiter=',')
    next(hotel_reader)
```

Gambar 3.9

2. Membaca Data dari File:

Program membuka file "tempat_wisata.txt" dan "hotel.txt" menggunakan perintah open() dengan mode "r" (read) untuk membaca data dari file tersebut. Menggunakan modul csv untuk membaca data CSV dengan menggunakan csv.reader(). Delimiter = ',' digunakan untuk menentukan pemisah antar kolom dalam file CSV. Mengabaikan baris header pertama menggunakan next() untuk memajukan iterator ke baris selanjutnya seperti pada Gambar 3.8 dan Gambar 3.9.

```
for row in tempat_wisata_reader:
    tempat_wisata_list.append(row)
```

Gambar 3.10

```
for row in hotel_reader:
    hotel_list.append(row)
```

Gambar 3.11

3. Membaca Data Tempat Wisata dan Hotel:

Melakukan iterasi pada setiap baris data dalam file "tempat_wisata.txt" menggunakan loop for dan menyimpannya dalam tempat_wisata_list (**Gambar 3.10**). Melakukan hal yang sama untuk file "hotel.txt" dan menyimpannya dalam hotel_list (**Gambar 3.11**).

```
tempat_wisata_list.sort(key=lambda element: int(element[0]))
hotel_list.sort(key=lambda element: int(element[0]))
```

Gambar 3.12

4. Pengurutan Data:

Menggunakan metode `sort()` untuk mengurutkan `tempat_wisata_list` dan `hotel_list` berdasarkan harga (indeks pertama pada setiap baris).

Pengurutan dilakukan dengan mengubah harga menjadi tipe data integer menggunakan `int()` dan menggunakan `key = lambda element: int(element[0])` untuk memperoleh elemen pertama dari setiap baris sebagai kunci pengurutan. Fungsi “key” bertujuan untuk menentukan kunci pengurutan atau kriteria pengurutan yang akan digunakan (**Gambar 3.12**). Penjelasan tentang “`key=lambda element: int(element[0])`” sebagai fungsi kunci pengurutan:

- “`Lambda element: int(element[0]):`” adalah ekspresi lambda atau fungsi anonim yang mengambil satu parameter yaitu `element`. Ekspresi lambda ini menerapkan konversi `int()` pada elemen pertama (`element[0]`) dari setiap baris dalam daftar.
- “`element[0]:`” mengacu pada elemen pertama dari setiap baris dalam daftar. Dalam konteks ini, elemen pertama adalah harga tempat wisata atau hotel.
- “`int(element[0]):`” Melalui ekspresi `int()`, harga tempat wisata atau hotel diubah menjadi tipe data integer. Hal ini diperlukan agar pengurutan dapat dilakukan dengan benar, karena secara default data dalam daftar dianggap sebagai string saat dilakukan pengurutan.

```
for tempat_wisata in tempat_wisata_list:
    for hotel in hotel_list:
        if cek_total_harga(int(tempat_wisata[0]) + int(hotel[0])):
            if not tempat_wisata in filtered_tempat_wisata_list:
                filtered_tempat_wisata_list.append(tempat_wisata)

            if not hotel in filtered_hotel_list:
                filtered_hotel_list.append(hotel)
        else:
            if int(hotel[0]) > int(tempat_wisata[0]):
                if hotel in filtered_hotel_list:
                    filtered_hotel_list.remove(hotel)
            else:
                if tempat_wisata in filtered_tempat_wisata_list:
                    filtered_tempat_wisata_list.remove(tempat_wisata)
```

Gambar 3.13

5. Memfilter Tempat Wisata dan Hotel:
 - a. Dilakukan nested loop dengan iterasi pada setiap tempat wisata dalam tempat_wisata_list dan setiap hotel dalam hotel_list.
 - b. Memeriksa apakah total harga tempat wisata dan hotel memenuhi kriteria dengan memanggil fungsi cek_total_harga() dengan total harga sebagai argumennya.
 - c. Jika memenuhi kriteria, tempat wisata dan hotel tersebut ditambahkan ke dalam daftar filtered_tempat_wisata_list dan filtered_hotel_list, masing-masing dengan menggunakan metode append().
 - d. Jika tidak memenuhi kriteria, tempat wisata dan hotel tersebut dihapus dari filtered_tempat_wisata_list dan filtered_hotel_list menggunakan metode remove().
 - e. Selama proses, mengelola pemilihan tempat wisata dan hotel berdasarkan perbandingan harga.

Seperti yang tertera pada **Gambar 3.13**.

```
tempat_wisata_dan_hotel_dict ["tempat_wisata"] = filtered_tempat_wisata_list
tempat_wisata_dan_hotel_dict ["hotel"] = filtered_hotel_list

return tempat_wisata_dan_hotel_dict
```

Gambar 3.14

6. Membentuk Dictionary Hasil:

Membuat dictionary tempat_wisata_dan_hotel_dict yang berisi dua kunci, yaitu "tempat_wisata" dan "hotel", yang masing-masing berisi daftar tempat wisata dan hotel yang telah difilter. Menggunakan assignment untuk menyimpan filtered_tempat_wisata_list dalam kunci "tempat_wisata" dan filtered_hotel_list dalam kunci "hotel" (**Gambar 3.14**). Kemudian Mengembalikan dictionary tempat_wisata_dan_hotel_dict yang berisi tempat wisata dan hotel yang telah difilter.

F. Def lama_libur()

Fungsi ini mengatur pemilihan jumlah hari liburan dan rentang budget pengguna untuk berwisata. Berikut adalah penjelasan dari program tersebut:

```
def lama_libur():
    while True:
        try:
            print(
                '''\nSilakan memilih jumlah hari:
                2 hari
                3 hari
                4 hari'''
            )
            global hari
            hari = int(input("Masukkan berapa lama waktu berwisata (2/3/4): "))
            if hari not in [2, 3, 4]:
                raise ValueError("Pilihan jumlah hari yang dimasukkan tidak valid, silakan memilih jumlah hari 2,
                break
            except ValueError as e:
                print("\n=====")
                print(f'#{404 Not Found#':^85}")
                print(f"{str(e):^85}")
                print("=====")
```

Gambar 3.15

1. Fungsi lama_libur memiliki loop while True yang akan terus berjalan hingga input yang valid diberikan oleh pengguna.
 - a. Di dalam loop, program akan mencetak daftar pilihan jumlah hari liburan yang tersedia (2, 3, atau 4) dan meminta pengguna untuk memasukkan pilihan mereka.
 - b. Input pengguna kemudian diubah menjadi bilangan bulat menggunakan fungsi int(input()) dan disimpan dalam variabel global hari.
 - c. Jika input pengguna tidak memasukkan pilihan yang valid (2, 3, atau 4), maka program akan mengangkat *ValueError* dengan pesan yang sesuai.
 - d. Jika input pengguna valid, loop akan dihentikan dengan menggunakan *break*.
- Seperti yang termuat pada **Gambar 3.15**.

```
global cek_total_harga
cek_total_harga = None

global kategori_budget
```

Gambar 3.16

2. Setelah loop, variabel cek_total_harga diinisialisasi dengan nilai None dan di-globalkan. Variabel ini akan digunakan nanti untuk memfilter tempat wisata dan hotel berdasarkan kriteria harga. Membuat variable kategori_budget menjadi global agar dapat dipanggil kembali (**Gambar 3.16**).

```

if hari == 2:
    print(
'''
Silahkan memilih rentang budget:
1. Kurang dari Rp 700.000,-
2. Lebih dari Rp 700.000,-'''
    )
    print("\n=====")
    kategori_budget = int(input("masukkan jumlah nominal budget anda (1/2): "))
    print("=====\\n")
    if kategori_budget == 1:
        cek_total_harga = lambda total_harga: total_harga <= 400000
    elif kategori_budget == 2:
        cek_total_harga = lambda total_harga: total_harga >= 400000
    else:
        lama_libur()

```

Gambar 3.17

3. Seperti yang telah termuat pada **Gambar 3.15**, pengguna diminta untuk memasukkan hari mereka berlibur. Jika hari adalah 2, program akan meminta pengguna untuk memilih rentang budget. Pilihan tersebut dicetak dan pengguna diminta untuk memasukkan pilihan mereka menggunakan `int(input())`. Pilihan pengguna disimpan dalam variabel `kategori_budget` (**Gambar 3.17**).

- Jika `kategori_budget` adalah 1, variabel `cek_total_harga` diinisialisasi dengan lambda fungsi yang mengembalikan `True` jika `total_harga <= 400000`.
- Jika `kategori_budget` adalah 2, variabel `cek_total_harga` diinisialisasi dengan lambda fungsi yang mengembalikan `True` jika `total_harga >= 400000`.
- Jika `kategori_budget` tidak valid, fungsi `lama_libur` akan dipanggil kembali untuk mengulangi pemilihan.

Hal yang serupa untuk hari yang berbeda dengan perbedaan pada kriteria rentang budget.

```

tempat_wisata_dan_hotel_dict = get_filtered_tempat_wisata_dan_hotel(cek_total_harga)
tempat_wisata_list = tempat_wisata_dan_hotel_dict["tempat_wisata"]
hotel_list = tempat_wisata_dan_hotel_dict["hotel"]

```

Gambar3.18

- fungsi `get_filtered_tempat_wisata_dan_hotel` dipanggil dengan argumen `cek_total_harga` untuk mendapatkan tempat wisata dan hotel yang sesuai dengan kriteria harga yang telah ditentukan. Hasil dari pemanggilan fungsi `get_filtered_tempat_wisata_dan_hotel` disimpan dalam variabel `tempat_wisata_dan_hotel_dict`, yang berisi dictionary dengan dua kunci:

5. "tempat_wisata" dan "hotel". Daftar tempat wisata dan hotel kemudian diambil dari dictionary tersebut dan disimpan dalam variabel tempat_wisata_list dan hotel_list (Gambar 3.18).

```
print("\nberikut tempat wisata yang dapat anda pilih")
print("="*77)
for tempat_wisata in tempat_wisata_list:
    print(f"| {tempat_wisata[1]:^53} | {tempat_wisata[0]:^8} | {tempat_wisata[2]:^8} |")
    print("="*77)
print("\nberikut tempat penginapan yang dapat anda pilih")
print("="*61)
for hotel in hotel_list:
    print(f"| {hotel[1]:^35} | {hotel[0]:^8} | {hotel[2]:^8} |")
    print("="*61)
```

Gambar 3.19

6. fungsi `get_filtered_tempat_wisata_dan_hotel` dipanggil dengan argumen `cek_total_harga` untuk mendapatkan tempat wisata dan hotel yang sesuai dengan kriteria harga yang telah ditentukan. Hasil dari pemanggilan fungsi `get_filtered_tempat_wisata_dan_hotel` disimpan dalam variabel `tempat_wisata_dan_hotel_dict`, yang berisi dictionary dengan dua kunci: "tempat_wisata" dan "hotel". Daftar tempat wisata dan hotel kemudian diambil dari dictionary tersebut dan disimpan dalam variabel `tempat_wisata_list` dan `hotel_list` (Gambar 3.19).

G. Def memilih()

Fungsi ini bertugas untuk mengambil input dari pengguna berupa pilihan tempat wisata dan hotel. Berikut adalah penjelasan detail dari program tersebut:

```
def memilih():
    input_tempat_wisata = input("\nsilahkan memilih tempat wisata: ")
    input_hotel = input("silahkan memilih tempat penginapan: ")

    total_harga = 0
```

Gambar 3.20

1. Fungsi `memilih` dimulai dengan meminta input dari pengguna untuk memilih tempat wisata, input tersebut disimpan dalam variabel `input_tempat_wisata`. Selanjutnya, pengguna diminta untuk memilih tempat penginapan (hotel), input tersebut disimpan

dalam variabel `input_hotel`. Variabel `total_harga` diinisialisasi dengan nilai 0. Variabel ini akan digunakan untuk menghitung total harga dari tempat wisata, hotel, dan pemandu yang dipilih.

```
for tempat_wisata in tempat_wisata_list:
    if tempat_wisata[1].lower() == input_tempat_wisata.lower():
        harga_tempat_wisata = int(tempat_wisata[0])
        total_harga += harga_tempat_wisata
        tempat_wisata_pilihan = tempat_wisata[1]
        break
else:
    print("\n=====")
    print(f"{'Input anda tidak valid':^43}")
    print("=====")
    memilih()
```

Gambar 3.2

```
for hotel in hotel_list:
    if hotel[1].lower() == input_hotel.lower():
        harga_per_malam = int(hotel[0])
        total_harga += harga_per_malam*hari
        hotel_pilihan = hotel[1]
        break
else:
    print("=====")
    print(f"{'input anda tidak valid':^43}")
    print("=====")
    memilih()
```

Gambar 3.22

- Program menggunakan loop for untuk mencari tempat wisata yang sesuai dengan input pengguna di dalam `tempat_wisata_list`. Jika tempat wisata ditemukan, harga tempat wisata ditambahkan ke `total_harga`, dan nama tempat wisata dipilih disimpan dalam variabel `tempat_wisata_pilihan`. Loop berhenti menggunakan `break` (**Gambar 3.21**). Jika loop for selesai tanpa menemukan tempat wisata yang sesuai dengan input pengguna, program mencetak pesan kesalahan dan memanggil fungsi `memilih()` kembali untuk meminta input ulang.

Setelah itu, program menggunakan loop for yang serupa untuk mencari hotel yang sesuai dengan input pengguna di dalam hotel_list. Jika hotel ditemukan, harga per malam dikalikan dengan jumlah hari (harga_per_malam * hari) dan ditambahkan ke total_harga, dan nama hotel dipilih disimpan dalam variabel hotel_pilihan. Loop berhenti menggunakan break (**Gambar 3.22**). Jika loop for selesai tanpa menemukan hotel yang sesuai dengan input pengguna, program mencetak pesan kesalahan dan memanggil fungsi memilih() kembali untuk meminta input ulang.

```
global pemandu_dict
pemandu_dict = pilih_pemandu()
total_harga += pemandu_dict["harga_pemandu"]
print("\n=====")
print("Total harga (tempat wisata, hotel, dan pemandu): ", total_harga)
print("="*60)
```

Gambar 3.23

3. pemandu_dict = pilih_pemandu(): Memanggil fungsi pilih_pemandu() untuk memilih pemandu wisata dan menyimpan hasilnya dalam variabel pemandu_dict. Fungsi ini akan mengembalikan sebuah kamus (dictionary) yang berisi informasi pemandu wisata terpilih dan harga pemandu.

total_harga += pemandu_dict["harga_pemandu"]: Menambahkan harga pemandu yang terdapat dalam pemandu_dict ke variabel total_harga. Harga pemandu diakumulasi ke total harga yang telah dihitung sebelumnya.

print("Total harga (tempat wisata, hotel, dan pemandu): ", total_harga): Mencetak total harga yang terdiri dari harga tempat wisata, hotel, dan pemandu. Nilai total harga diambil dari variabel total_harga yang telah diakumulasi sepanjang program.

```
while True:
    try:
        input_biaya = int(input("\nMasukkan biaya dengan nominal pas: "))

        # Lanjutkan dengan program selanjutnya
        if input_biaya == total_harga:
            break
        elif input_biaya > total_harga:
            raise ValueError(f"Nominal yang dimasukkan lebih dari total harga, silakan memasukkan nominal {total_harga}.")
        elif input_biaya < total_harga:
            raise ValueError(f"Nominal yang dimasukkan kurang dari total harga, silakan memasukkan nominal {total_harga}.")

    except ValueError as e:
        print("\n=====")
        print(f"{'Kesalahan Jumlah biaya':^80}")
        print(f"{str(e):^80}")
        print("=====")

konfirmasi_pembayaran = input("\nApakah Anda setuju dengan pembayaran tersebut? (ya/tidak) ")
```

Gambar 3.24

4. Program di atas bertujuan untuk meminta pengguna memasukkan jumlah biaya dengan nominal yang pas untuk melakukan pembayaran. Berikut adalah penjelasan dari program tersebut:

- a. Menggunakan input untuk meminta pengguna memasukkan jumlah biaya dengan nominal yang pas pada variable input_biaya.
Program melakukan pengecekan kondisi sebagai berikut:

- b. Jika input_biaya sama dengan total_harga, artinya pengguna memasukkan jumlah biaya yang pas, maka loop dihentikan menggunakan break.
- c. Jika input_biaya lebih besar dari total_harga, program menghasilkan sebuah ValueError dengan pesan yang menyatakan bahwa nominal yang dimasukkan lebih dari total harga yang harus dibayar. Jika program menghasilkan ValueError, pesan kesalahan beserta penjelasannya akan dicetak ke layar.
- d. Jika input_biaya lebih kecil dari total_harga, program menghasilkan sebuah ValueError dengan pesan yang menyatakan bahwa nominal yang dimasukkan kurang dari total harga yang harus dibayar. Jika program menghasilkan ValueError, pesan kesalahan beserta penjelasannya akan dicetak ke layar.

Setelah mencapai break dan keluar dari loop, program lanjut ke baris selanjutnya.

Program meminta pengguna untuk mengonfirmasi pembayaran dengan memasukkan "ya" atau "tidak" menggunakan input pada variable konfirmasi_pembayaran. Program berlanjut ke baris selanjutnya setelah pengguna memberikan konfirmasi pembayaran (**Gambar 3.24**).

```
while True:
    konfirmasi_pembayaran = input("Apakah Anda ingin melanjutkan pembayaran? (ya/tidak): ")

    if konfirmasi_pembayaran.lower() == "ya":
        print()
        cetak_struk_holiyay(
            username, tempat_wisata_pilihan, harga_tempat_wisata, hotel_pilihan, harga_per_malam, hari,
            pemandu_dict["pemandu_terpilih"], pemandu_dict["harga_pemandu"],
            total_harga, input_biaya)
        exit()
```

Gambar 3.25

5. Jika konfirmasi_pembayaran yang dimasukkan pengguna adalah "ya" (baik huruf besar maupun huruf kecil), program akan mencetak struk pembayaran menggunakan fungsi cetak_struk_holiyay(), kemudian program berakhir dengan fungsi exit() (**Gambar 3.25**).

```
elif konfirmasi_pembayaran.lower() == "tidak":
    print('\nSilakan memilih opsi:
    1. Menu Awal
    2. Pilih Wisata dan Penginapan
    3. Keluar')
    kembali = input("Masukkan pilihan anda (1/2/3): ")
    if kembali == "1":
        menu_awal()
    elif kembali == "2":
        lama_libur()
    else:
        keluar()
```

Gambar 3.26

7. Jika konfirmasi_pembayaran yang dimasukkan pengguna adalah "tidak" (baik huruf besar maupun huruf kecil), program akan menampilkan opsi menu kepada pengguna untuk memilih tindakan selanjutnya (kembali ke Menu Awal, memilih Wisata dan Penginapan, atau keluar) (**Gambar 3.26**).

```
else:
    print("Input yang Anda masukkan tidak valid. Silakan masukkan 'ya' atau 'tidak'.")
    konfirmasi_pembayaran = input("Apakah Anda ingin melanjutkan pembayaran? (ya/tidak): ")
```

Gambar 3.27

8. Jika konfirmasi_pembayaran yang dimasukkan pengguna tidak valid (bukan "ya" atau "tidak"), program akan mencetak pesan kesalahan dan meminta pengguna untuk mengulangi mengonfirmasi pembayaran. Jika pengguna memasukkan "ya", maka

pengguna diarahkan kembali pada proses pembayaran. Jika pengguna memasukkan “tidak” maka pengguna diberikan menampilkan opsi menu kepada pengguna untuk memilih tindakan selanjutnya (kembali ke Menu Awal, memilih Wisata dan Penginapan, atau keluar) (**Gambar 3.27**).

F. Def pilih_pemandu(

```
def pilih_pemandu():
    print('')
    Silahkan pilih pemandu Anda:

    1. Pemandu A
    Berikut Kriteria Pemandu A :
    1. Memiliki kemampuan 2 bahasa asing (bahasa inggris dan bahasa Italia)
    2. Sudah pernah menjelajahi keseluruhan Pulau Bali
    3. Mengetahui tempat kuliner yang enak di Bali
    4. Biaya Tourguide A adalah Rp 100.000,-

    2. Pemandu B
    Berikut Kriteria Pemandu B :
    1. Fasih dalam berbahasa daerah Bali
    2. Mengetahui sejarah dan adat istiadat Bali
    3. Tahu tempat diving yang bagus di Bali
    4. Biaya Tourguide B adalah Rp 80.000,-

    3. Pemandu C
    Berikut Kriteria Pemandu C :
    1. Menguasai 3 jenis bahasa asing (bahasa inggris, bahasa chinese, bahasa spanyol)
    2. Sudah kenal dekat dengan beberapa pemilik wisata di Bali
    3. Tahu tempat berbelanja oleh-oleh di Bali
    4. Biaya Tourguide C adalah Rp 120.000,-

    4. Tidak menggunakan pemandu wisata''')
```

Gambar 3.28

1. Pada fungsi pilih_pemandu(), pengguna akan ditampilkan kriteria untuk setiap pemandu yang ada.

```
global pemandu_dict
global pemandu_terpilih
global harga_pemandu

pemandu_dict = {}
pemandu_terpilih = None
harga_pemandu = None
```

Gambar 3.29

2. Membuat pemandu_dict, pemandu_terpilih, dan harga_pemandu menjadi global. Membuat dictionary pemandu_dict kosong, membuat variabel kosong pemandu_terpilih, dan harga_pemandu.

```
while pemandu_terpilih is None:
    pilihan = input("\nMasukkan pilihan Anda (1/2/3/4): ")

    if pilihan == "1":
        print("Anda memilih pemandu A.")
        pemandu_terpilih = "A"
        harga_pemandu = 100000

    elif pilihan == "2":
        print("Anda memilih pemandu B.")
        pemandu_terpilih = "B"
        harga_pemandu = 80000

    elif pilihan == "3":
        print("Anda memilih pemandu C.")
        pemandu_terpilih = "C"
        harga_pemandu = 120000

    elif pilihan == "4":
        print("Anda tidak menggunakan jasa pemandu wisata")
        pemandu_terpilih = "tanpa pemandu"
        harga_pemandu = 0

    else:
        print("Pilihan tidak valid. Silakan pilih antara 1, 2, 3, atau 4.")
```

Gambar 3.30

3. while pemandu_terpilih is None:: Memulai loop while yang akan berjalan selama pemandu_terpilih masih bernilai None, artinya pengguna belum memilih pemandu. pilihan = input("\nMasukkan pilihan Anda (1/2/3/4): "): Meminta pengguna untuk memasukkan pilihan pemandu dengan angka 1, 2, 3, atau 4. Pada bagian berikutnya, terdapat percabangan if-elif-else untuk menentukan pemandu terpilih dan harga pemandu berdasarkan pilihan pengguna:

- a. Jika pilihan sama dengan "1", maka pemandu A dipilih. Variabel pemandu_terpilih diisi dengan "A" dan harga_pemandu diisi dengan 100000.
- b. Jika pilihan sama dengan "2", maka pemandu B dipilih. Variabel pemandu_terpilih diisi dengan "B" dan harga_pemandu diisi dengan 80000.
- c. Jika pilihan sama dengan "3", maka pemandu C dipilih. Variabel pemandu_terpilih diisi dengan "C" dan harga_pemandu diisi dengan 120000.
- d. Jika pilihan sama dengan "4", maka pengguna memilih tanpa pemandu. Variabel pemandu_terpilih diisi dengan "tanpa pemandu" dan harga_pemandu diisi dengan 0.
- e. Jika pilihan bukan 1/2/3/4, maka pengguna diminta mengisi dengan benar.

```
pemandu_dict["pemandu_terpilih"] = pemandu_terpilih
pemandu_dict["harga_pemandu"] = harga_pemandu

return pemandu_dict
```

Gambar 3.31

4. pemandu_dict["pemandu_terpilih"] = pemandu_terpilih: Menyimpan nilai pemandu_terpilih ke dalam kamus pemandu_dict dengan kunci "pemandu_terpilih".

pemandu_dict["harga_pemandu"] = harga_pemandu: Menyimpan nilai harga_pemandu ke dalam kamus.

G. Def bank()

Fungsi ini bertujuan untuk meminta pengguna memilih jenis bank sebagai metode pembayaran. Berikut adalah penjelasan detail dari program tersebut:


```
def bank():
    print('\nSilahkan memilih jenis bank:
        1. Bank BRI
        2. Bank BCA
        3. Bank Mandiri')
    global jenis_pembayaran
    jenis_pembayaran = (input("Masukkan jenis bank (1/2/3): "))

    while jenis_pembayaran not in ["1", "2", "3"]:
        print("\n=====")
        print("Jenis bank yang Anda masukkan tidak valid. Silakan masukkan 1/2/3.")
        print("=====\\n")
        jenis_pembayaran = (input("Masukkan jenis bank (1/2/3): "))
```

Gambar 3.32

1. global jenis_pembayaran: Mendeklarasikan variabel jenis_pembayaran sebagai variabel global, yang akan digunakan untuk menyimpan pilihan jenis bank yang dipilih oleh pengguna.
2. jenis_pembayaran = (input("Masukkan jenis bank (1/2/3): ")): Meminta pengguna untuk memasukkan pilihan jenis bank dengan memasukkan angka 1, 2, atau 3 sesuai dengan opsi bank yang disediakan.
3. while jenis_pembayaran not in ["1", "2", "3"]:: Memulai loop while yang akan berjalan selama jenis_pembayaran tidak ada di dalam list ["1", "2", "3"], artinya pengguna memasukkan jenis bank yang tidak valid.
Di dalam loop while, program akan mencetak pesan kesalahan dan meminta pengguna untuk memasukkan jenis bank yang valid jika jenis bank yang dimasukkan tidak valid.

H. Def cetak_struk_Holiyay()

Fungsi ini digunakan untuk mencetak struk pembayaran atau kuitansi setelah proses pembayaran selesai. Berikut adalah penjelasan dari program tersebut:

```
def cetak_struk_holiyay(
    username, tempat_wisata, harga_tempat_wisata, hotel_pilihan,
    harga_per_malam, hari, pemandu_terpilih, harga_pemandu, harga_total, input_biaya):
    print("="*50)
    print(f"|          {'STRUK HOLIIYAY':^25}          |")
    print("="*50)
    print(f"| Username           : {username:^25}|")
    print(f"| Tempat Wisata      : {tempat_wisata:^25}|",)
    print(f"| Harga Tempat Wisata : {harga_tempat_wisata:^25}|")
    print(f"| Nama Hotel         : {hotel_pilihan:^25}|")
    print(f"| Harga per Malam    : {harga_per_malam:^25}|")
    print(f"| Jumlah hari        : {hari:^25}|")
    print(f"| Tour Guide         : {pemandu_terpilih:^25}|")
    print(f"| Harga Tour Guide   : {harga_pemandu:^25}|")
    print("-"*50)
    print(f"| Total Biaya        : {harga_total:^25}|")
    print(f"| Biaya yang dibayar : {input_biaya:^25}|")
    print("="*50)
    print("\nTerima kasih sudah menggunakan program Holiyay")
```

Gambar 3.34

1. cetak_struk_holiyay(...): Mendefinisikan fungsi cetak_struk_holiyay dengan parameter sebagai berikut:
 - a. username: Nama pengguna.
 - b. tempat_wisata: Nama tempat wisata yang dipilih.
 - c. harga_tempat_wisata: Harga tempat wisata.
 - d. hotel_pilihan: Nama hotel yang dipilih.
 - e. harga_per_malam: Harga per malam penginapan di hotel.
 - f. hari: Jumlah hari yang dipilih untuk berwisata.
 - g. pemandu_terpilih: Nama pemandu yang dipilih.
 - h. harga_pemandu: Harga pemandu wisata.
 - i. harga_total: Total biaya keseluruhan (termasuk tempat wisata, hotel, dan pemandu).
 - j. input_biaya: Jumlah biaya yang dibayar oleh pengguna.

Parameter di atas diambil dari berjalannya proses program dan *input* dari pengguna.

Setelah sign up telah berhasil maka pengguna akan dikembalikan pada tampilan menu awal yaitu pilihan masuk. Pengguna dapat memilih sign in dan memasukkan data username dan password yang telah dibuat pada menu sign up sebelumnya, setelah itu pengguna dapat masuk ke akun yang telah dibuat. Berikut ini merupakan contoh gambar dari penjelasan.

```
Silakan memilih opsi:
1. Sign up
2. Sign in
3. Keluar
Masukkan pilihan (1/2/3): 2

=====
Masukkan username: karisa
Masukkan password: kelompok13
=====
Sign in berhasil!
```

Gambar 4.3 Sign In

B. Menu Utama

1. Pilihan Hari

Setelah pengguna berhasil masuk ke menu utama maka program akan menampilkan pilihan hari yang terdiri dari 3 opsi yaitu 2 hari, 3 hari, dan 4 hari. pengguna dapat memasukkan pilihan atau jawaban pada kolom yang telah disediakan. Berikut ini merupakan contoh pilihan yang dapat dipilih oleh pengguna.

```
Silakan memilih jumlah hari:
2 hari
3 hari
4 hari
Silakan memilih berapa lama waktu berwisata (2/3/4): 2
```

Gambar 4.4 Pilihan Lama Berwisata

2. Pilihan Budget

Setelah pengguna memilih hari, kemudian program akan memberikan pilihan estimasi harga sesuai hari yang telah dipilih. Berikut ini merupakan gambar contoh ketika pengguna menginputkan 2 hari dengan masing-masing rentang budget.

```
Silakan memilih jumlah hari:
  2 hari
  3 hari
  4 hari
Silakan memilih berapa lama waktu berwisata (2/3/4): 2

Silahkan memilih rentang budget:
  1. Kurang dari Rp 600.000,-
  2. Lebih dari Rp 600.000,-
```

Gambar 4.5 Pilihan Budget dengan 2 Hari Lama Wisata

Berikut ini merupakan gambar contoh ketika pengguna memilih 3 hari dengan masing-masing rentang budget.

```
Silakan memilih jumlah hari:
  2 hari
  3 hari
  4 hari
Silakan memilih berapa lama waktu berwisata (2/3/4): 3

Silahkan memilih rentang budget:
  1. Kurang dari Rp 800.000,-
  2. Lebih dari Rp 800.000,-
```

Gambar 4.6 Pilihan Budget dengan 3 Hari Lama Wisata

Berikut ini merupakan gambar contoh ketika pengguna memilih 4 hari dengan masing-masing rentang budget.

```
Silakan memilih jumlah hari:
  2 hari
  3 hari
  4 hari
Silakan memilih berapa lama waktu berwisata (2/3/4): 4

Silahkan memilih rentang budget:
  1. Kurang dari Rp 1.000.000,-
  2. Lebih dari Rp 1.000.000,-
```

Gambar 4.7 Pilihan Budget dengan 4 Hari Lama Wisata

3. Pilihan Tempat Wisata dan Tempat Penginapan

Setelah pengguna memilih rentang budget, maka kemudian program akan menampilkan list tempat wisata dan tempat penginapan yang sesuai dengan harga yang telah ditentukan pengguna sehingga pengguna dapat memilih tempat wisata dan tempat penginapan yang akan mereka gunakan. Berikut ini merupakan contoh list tempat wisata dari semua rentang budget.

berikut tempat wisata yang dapat anda pilih

Pantai Kuta	0	4.5
Pantai Jimbaran	0	4.4
Pantai Muaya	0	4.5
Pantai Sanur	0	4.5
Pantai Tanjung Benoa	5000	4.4
Nusa Penida	10000	4.7
Danau Batur	15000	4.6
Desa Penglipuran	15000	4.8
Air Terjun Sekumpul	15000	3.9
Pantai Pandawa	20000	4.6
Tanah Lot	20000	4.7
Pura Tirta Empul Tampak Siring	30000	4.6
Danau Beratan Bedugul	30000	4.7
Pura Luhur Uluwatu	40000	4.6
Pura Ulun Siwi	50000	4.6
Dream Museum Zone	52000	4.0
Monkey Forest Ubud	60000	4.5
GWK	90000	4.5
Taman Budaya Garuda Wisnu Kencana dan New Kuta Golf	115000	4.5

Gambar 4.8 List Tempat Wisata

Setelah rentang budget ditentukan oleh pengguna, maka program akan memilih tempat penginapan yang sesuai dengan rentang budget yang telah ditentukan diatas. Berikut ini merupakan contoh list tempat penginapan pada rentang budget di atas Rp1.000.000,- yang dapat dipilih oleh pengguna.

```

berikut tempat penginapan yang dapat anda pilih
=====
|          Easton Kuta          | 110000 | 3.7 |
=====
|        Restu Bali Hotel        | 120000 | 4.1 |
=====
|        Mirah Hostel           | 120000 | 4.2 |
=====
|    Hotel Segara Sadhu Inn      | 130000 | 4   |
=====
|        Sugi Gede Homestay      | 180000 | 4.8 |
=====
|        Hotel Nuraini          | 190000 | 4.3 |
=====
|    Hotel The Hill Ungasan     | 200000 | 4   |
=====
|        Pondok Agung           | 230000 | 4.8 |
=====

```

Gambar 4.9 List Tempat Penginapan

Pengguna dapat memasukkan data tempat wisata dan data tempat penginapan yang dipilih pada kolom yang telah disediakan. Berikut ini merupakan contoh data yang dapat dimasukkan oleh pengguna.

```

silahkan memilih tempat wisata: tanah lot
silahkan memilih tempat penginapan: hotel nuraini
=====

```

Gambar 4.10 Input Wisata dan Penginapan dari User

4. Pilihan Pemandu Wisata

Selanjutnya pengguna akan diminta untuk memilih pemandu wisata, terdapat 4 opsi dengan beberapa kriteria tertentu. Opsi 1 yaitu pemandu A, opsi 2 yaitu pemandu B, opsi 3 yaitu pemandu C, dan opsi 4 yaitu tanpa pemandu. Berikut ini merupakan contoh kriteria pemandu yang tersedia.

Silahkan pilih pemandu Anda:

1. Pemandu A
Berikut Kriteria Pemandu A :
 1. Memiliki kemampuan 2 bahasa asing (bahasa inggris dan bahasa Italia)
 2. Sudah pernah menjelajahi keseluruhan Pulau Bali
 3. Mengetahui tempat kuliner yang enak di Bali
 4. Biaya Tourguide A adalah Rp 100.000,-
2. Pemandu B
Berikut Kriteria Pemandu B :
 1. Fasih dalam berbahasa daerah Bali
 2. Mengetahui sejarah dan adat istiadat Bali
 3. Tahu tempat diving yang bagus di Bali
 4. Biaya Tourguide B adalah Rp 80.000,-
3. Pemandu C
Berikut Kriteria Pemandu C :
 1. Menguasai 3 jenis bahasa asing (bahasa inggris, bahasa chinese, bahasa spanyol)
 2. Sudah kenal dekat dengan beberapa pemilik wisata di Bali
 3. Tahu tempat berbelanja oleh-oleh di Bali
 4. Biaya Tourguide C adalah Rp 120.000,-
4. Tidak menggunakan pemandu wisata

Gambar 4.11 List Kriteria Pemandu Wisata

Setelah menetapkan pilihan pemandu, pengguna dapat memasukkan data pemandu tersebut pada kolom yang telah dijelaskan. Berikut merupakan contoh pilihan pengguna.

Masukkan pilihan Anda (1/2/3/4): 2
Anda memilih pemandu B.

Gambar 4.12 Input pemandu oleh User

C. Total Harga dan Pembayaran

Bagian selanjutnya yaitu total harga, setelah pengguna memilih jumlah hari, tempat wisata dan tempat penginapan, maka program akan merekap semua data yang telah dimasukkan pengguna menjadi bentuk total harga.


```
=====
Total harga (tempat wisata, hotel, dan pemandu): 480000
=====
```

Gambar 4.13 Total Harga

Setelah pengguna mengetahui total harga, maka pengguna akan diminta untuk menentukan metode pembayaran dengan menggunakan 3 pilihan bank yaitu BRI, BCA dan Mandiri. pengguna dapat memilih salah satu dari ketiga opsi bank tersebut. Setelah itu, pengguna diminta untuk memasukkan biaya dengan nominal yang pas dan sesuai dengan harga total yang telah ditentukan. Berikut ini merupakan contoh data yang dapat dimasukkan oleh pengguna.

```
Silahkan memilih jenis bank:
1. Bank BRI
2. Bank BCA
3. Bank Mandiri
Masukkan jenis bank (1/2/3): 1
Masukkan biaya dengan nominal pas: 480000
Apakah Anda setuju dengan pembayaran tersebut? (ya/tidak) ya
```

Gambar 4.14 Pilihan Jenis Bank

Kemudian program akan meminta persetujuan dari pengguna terhadap pembayaran yang dilakukan, apabila pengguna bersedia maka program akan dilanjutkan. Namun, apabila pengguna tidak bersedia maka program akan meminta pengguna untuk memilih 3 pilihan opsi sebagai berikut.

```
Apakah Anda setuju dengan pembayaran tersebut? (ya/tidak) tidak
Silakan memilih opsi:
1. Menu Awal
2. Pilih Wisata dan Penginapan
3. Keluar
Masukkan pilihan anda (1/2/3): █
```

Gambar 4.15 Persetujuan User

D. Struk Pembayaran

Setelah pengguna menyetujui pembayaran, maka pengguna akan mendapatkan struk pembayaran sehingga pengguna dapat mengecek total bayar dengan harga yang tertera. Berikut ini merupakan contoh dari struk pembayaran.

```
Silahkan memilih jenis bank:
1. Bank BRI
2. Bank BCA
3. Bank Mandiri
Masukkan jenis bank (1/2/3): 1
Masukkan biaya dengan nominal pas: 480000
Apakah Anda setuju dengan pembayaran tersebut? (ya/tidak) ya
=====
|                               |
|          STRUK HOLIYAY       |
|                               |
|-----|-----|
| Username      :      karisa  |
| Tempat Wisata :      Tanah Lot |
| Harga Tempat Wisata :      20000 |
| Nama Hotel    :      Hotel Nuraini |
| Harga per Malam :      190000 |
| Jumlah hari   :      2 |
| Tour Guide    :      B |
| Harga Tour Guide :      80000 |
|-----|-----|
| Total Biaya   :      480000 |
| Biaya yang dibayar :      480000 |
|-----|-----|
Terima kasih sudah menggunakan program Holiyay
```

Gambar 4.16 Struk Pembayaran

E. Kondisi Kesalahan Input User

Pada program-program diatas, pengguna telah diminta untuk menginputkan data secara manual, oleh karena itu terdapat kemungkinan pengguna salah dalam menuliskan atau typo. Oleh karena itu terdapat beberapa cek error yang digunakan untuk meminimalisir terjadinya error tersebut. Berikut merupakan beberapa contoh penggunaannya dalam program tersebut.

1. Kesalahan Masuk

Kesalahan masuk dapat terjadi apabila terdapat ketidakcocokan antara database pengguna dengan data yang diinputkan oleh pengguna, sehingga hal tersebut dapat menyebabkan error. Program tersebut akan menampilkan peringatan

atau notifikasi apabila pengguna salah dalam penginputan data sehingga program akan kembali meminta pengguna untuk memasukkan data yang sesuai Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan data username atau password.

```
Silakan memilih opsi:
1. Sign up
2. Sign in
3. Keluar
Masukkan pilihan (1/2/3): 2

=====
Masukkan username: 123
Masukkan password: 345
=====

Username atau password salah.

=====
Masukkan username: 
```

Gambar 4.17 Kesalahan Input Masuk

2. Kesalahan input jumlah hari

Kesalahan input jumlah hari dapat terjadi apabila pengguna salah dalam dalam penginputan angka pilihan hari, atau jumlah hari yang diinginkan tidak terdapat dalam pilihan, sehingga hal tersebut dapat menyebabkan error. Program tersebut akan menampilkan peringatan atau notifikasi apabila pengguna salah dalam penginputan data jumlah hari sehingga program akan kembali meminta pengguna untuk menginputkan kembali jumlah hari yang sesuai. Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan data jumlah hari.

```
Silakan memilih jumlah hari:
  2 hari
  3 hari
  4 hari
Silakan memilih berapa lama waktu berwisata (2/3/4): 5

=====
#404 Not Found#
Pilihan jumlah hari yang dimasukkan tidak valid, silakan memilih jumlah hari 2, 3, atau 4.
=====
```

Gambar 4.18 Kesalahan Pemilihan Lama Wisata

3. Kesalahan memilih input rentang budget

Kesalahan dalam memilih input rentang budget dapat terjadi apabila pengguna memasukkan pilihan selain angka yang telah ditampilkan sehingga hal tersebut dapat menyebabkan program menjadi error. Program tersebut akan menampilkan peringatan atau notifikasi apabila pengguna salah dalam penginputan data rentang budget yang telah ditentukan. Kemudian program akan menampilkan ulang pilihan sehingga pengguna dapat memasukkan rentang budget yang sesuai.

Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan data rentang budget.

```
Silahkan memilih rentang budget:
  1. Kurang dari Rp 800.000,-
  2. Lebih dari Rp 800.000,-

=====
masukkan jumlah nominal budget anda (1/2): 3
=====

Silakan memilih jumlah hari:
  2 hari
  3 hari
  4 hari
Silakan memilih berapa lama waktu berwisata (2/3/4):
```

Gambar 4.19 Kesalahan Input Rentang Budget

4. Kesalahan input tempat wisata atau tempat penginapan

Kesalahan dalam memilih input tempat wisata dan tempat penginapan dapat terjadi apabila pengguna memasukkan pilihan selain list tempat wisata atau list tempat penginapan yang telah ditampilkan, sehingga hal tersebut dapat menyebabkan program menjadi error. Program tersebut akan menampilkan peringatan atau notifikasi apabila pengguna salah dalam penginputan data tempat wisata dan tempat penginapan yang telah ditentukan. Kemudian program akan menampilkan ulang pilihan sehingga pengguna dapat memasukkan tempat wisata atau tempat penginapan yang sesuai. Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan data tempat penginapan.

```
silahkan memilih tempat wisata: pantai kuta
silahkan memilih tempat penginapan: pondok agun
=====
input anda tidak valid
silahkan memilih tempat wisata: █
```

Gambar 4.20 Kesalahan Input Tempat Wisata dan Penginapan

5. Kesalahan memilih input pemandu

Kesalahan dalam memilih input pemandu dapat terjadi apabila pengguna memasukkan pilihan selain angka yang telah ditampilkan sehingga hal tersebut dapat menyebabkan program menjadi error. Program tersebut akan menampilkan peringatan atau notifikasi apabila pengguna salah dalam penginputan data pemandu yang telah ditentukan. Kemudian program akan menampilkan ulang pilihan sehingga pengguna dapat memasukkan pilihan pemandu yang sesuai.

Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan pilihan pemandu.

```
Masukkan pilihan Anda (1/2/3/4): 5
Pilihan tidak valid. Silakan pilih antara 1, 2, 3, atau 4.
Masukkan pilihan Anda (1/2/3/4): █
```

Gambar 4.21 Kesalahan Input Pemandu

6. Kesalahan input jenis bank

Kesalahan dalam memilih input jenis bank dapat terjadi apabila pengguna memasukkan pilihan selain angka yang telah ditampilkan sehingga hal tersebut dapat menyebabkan program menjadi error. Program tersebut akan menampilkan peringatan atau notifikasi apabila pengguna salah dalam penginputan jenis bank yang telah ditentukan. Kemudian program akan menampilkan ulang pilihan sehingga pengguna dapat memasukkan jenis bank yang sesuai. Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan pilihan jenis bank.

```
Silahkan memilih jenis bank:
1. Bank BRI
2. Bank BCA
3. Bank Mandiri
Masukkan jenis bank (1/2/3): 4
Jenis bank yang Anda masukkan tidak valid. Silakan coba lagi.
Masukkan jenis bank (1/2/3):
```

Gambar 4.22 Kesalahan Input Jenis Bank

7. Kesalahan input nominal pembayaran

Kesalahan dalam input nominal pembayaran dapat terjadi apabila pengguna salah dalam penginputan nominal, atau nominal yang diinputkan tidak pas dan tidak sesuai dengan harga total yang sudah ditentukan. Program tersebut akan menampilkan peringatan atau notifikasi apabila pengguna salah dalam penginputan data nominal pembayaran dengan harga total yang telah ditentukan. Kemudian program akan menampilkan ulang kolom jawaban sehingga pengguna dapat memasukkan nominal pembayaran yang sesuai. Berikut ini merupakan contoh tampilan apabila pengguna salah dalam penginputan data nominal pembayarant.

```
Masukkan biaya dengan nominal pas: 680000
Kesalahan Jumlah biaya
Nominal yang dimasukkan kurang dari total harga, silakan memasukkan nominal 690000.
Masukkan biaya dengan nominal pas:
```

Gambar 4.23 Kesalahan Input Nominal Bayar

DAFTAR PUSTAKA

Pentingnya Penggunaan Aplikasi Mobile untuk Strategi Promosi Pariwisata, n.d..

Retrieved from : Gamatechnoblog

<https://blog.gamatechno.com/pentingnya-penggunaan-aplikasi-mobile-untuk-strategi-promosi-pariwisata/>

Sekilas Bali. Retrieved from : Sistem Informasi dan Tata Ruang Bali

<https://tarubali.baliprov.go.id/sekilas-bali/>

Banyaknya Wisatawan Mancanegara Bulanan ke Bali Menurut Pintu Masuk (Orang),

2022. Retrieved from : Badan Pusat Statistik Provinsi Bali

<https://bali.bps.go.id/indicator/16/106/2/banyaknya-wisatawan-mancanegara-bulanan-ke-bali-menurut-pintu-masuk.html>

