

**LAPORAN**  
**PENGEMBANGAN APLIKASI FRONT OFFICE HOTEL UNTUK INPUT**  
**DATA KAMAR**  
**DJANGO DAN BOOTSTRAP**



**Disusun oleh**  
**DAFA RIZQY\_3337230010\_KELAS B**

**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS SULTAN AGENG TIRTAYASA**  
**2024**

## DAFTAR ISI

### DAFTAR ISI

LAPORAN.....	i
DAFTAR ISI.....	ii
BAB I.....	2
1.1    Latar Belakang .....	2
1.2    Tujuan Praktikum .....	2
1.3    Batasan Praktikum .....	3
BAB II .....	3
2.1    Sejarah dan Evolusi Django dan Bootstrap .....	3
2.2    Struktur Pemograman.....	4
2.3    Teori Pemograman .....	5
2.4    Perintah Dasar Python Django dan Bootstrap .....	6
BAB III.....	8
3.1    Perancangan Algoritma Kalimat Deskriptif .....	8
3.2    Perancangan Algoritma Pseudocode .....	9
3.3    Perancangan Flowchart.....	12
3.4    Perancangan Database.....	13
3.5    Perancangan Desain Interface .....	15
BAB IV .....	15
4.1    Coding Program .....	15
4.2    Hasil Coding .....	23
BAB V .....	24
5.1    Kesimpulan .....	24
5.2    Saran .....	24
DAFTAR PUSTAKA.....	26

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Sekarang ini, teknologi telah menjadi bagian yang tak terpisahkan dari bisnis perhotelan. Untuk meningkatkan efisiensi dan pengalaman pelanggan, manajemen hotel perlu memperhatikan sistem front office hotel yang baik. Salah satu fitur penting yang harus ada dalam sistem tersebut adalah kemampuan untuk input data kamar hotel.

Pasti Anda pernah mengalami kesulitan dalam mendata informasi kamar hotel, seperti nomor kamar, tipe kamar, harga per malam, dan ketersediaan kamar. Dengan menggunakan laporan aplikasi front office hotel, Anda bisa lebih mudah mengelola informasi tersebut dan mengoptimalkan proses manajemen kamar hotel.

Dengan bantuan Django, Anda dapat membuat form input data kamar yang sesuai dengan kebutuhan hotel Anda. Django dapat membantu Anda dalam mengatur database, melakukan validasi data, dan memudahkan proses penyimpanan informasi kamar hotel. Sedangkan Bootstrap akan membantu Anda dalam desain tampilan agar terlihat menarik dan responsif di berbagai perangkat.

Dengan laporan aplikasi front office hotel untuk input data kamar, Anda bisa lebih efisien dalam mengelola informasi kamar hotel dan memberikan pelayanan yang lebih baik kepada tamu.

### **1.2 Tujuan Praktikum**

#### **A. Mempelajari Django Framework:**

- Memahami dasar-dasar Django sebagai framework web berbasis Python.
- Menguasai cara menginstal dan mengkonfigurasi proyek Django.
- Mengetahui cara mengatur URL, views, dan models dalam Django.

#### **B. Implementasi Front-End dengan Bootstrap:**

- Memahami prinsip-prinsip dasar desain responsif menggunakan Bootstrap.
- Mampu mengintegrasikan Bootstrap dengan proyek Django untuk menciptakan antarmuka yang menarik dan mudah digunakan.

#### **C. Pengembangan Fungsionalitas Aplikasi Pengembangan Aplikasi Front Office Hotel untuk Input data Kamar:**

- Mengembangkan fitur-fitur dasar aplikasi Aplikasi Pengembangan Aplikasi Front Office Hotel untuk Input data Kamar seperti menambahkan, menghapus, dan menyunting tugas.
- Memahami cara mengelola data menggunakan Django ORM (Object-Relational Mapping).

#### **D. Pengujian dan Debugging:**

- Mampu melakukan pengujian terhadap aplikasi untuk memastikan semua fitur berfungsi dengan baik.
- Mengetahui teknik-teknik debugging untuk mengidentifikasi dan memperbaiki kesalahan dalam kode.

#### **E. Dokumentasi dan Pelaporan:**

- Mampu mendokumentasikan setiap tahapan pengembangan aplikasi secara sistematis.

- Menyusun laporan yang komprehensif dan jelas mengenai proses pengembangan, tantangan yang dihadapi, dan solusi yang diterapkan.

**F. Pengembangan Keterampilan Kerja Tim dan Manajemen Proyek:**

- Meningkatkan keterampilan bekerja dalam tim melalui kolaborasi dalam pengembangan proyek.
- Menerapkan prinsip-prinsip manajemen proyek untuk mengatur waktu dan sumber daya secara efektif.

### 1.3 Batasan Praktikum

- Praktikum ini hanya akan fokus pada pembuatan laporan aplikasi front office hotel untuk input data kamar menggunakan Django dan Bootstrap.
- Tidak akan membahas implementasi fitur-fitur tambahan seperti reservasi kamar, check-in/check-out, atau manajemen user.
- Penggunaan framework Django dan Bootstrap dalam pembuatan aplikasi merupakan keharusan, namun tidak akan ada pendalaman mendalam terhadap konsep-konsep yang kompleks.
- Bahan praktikum akan disiapkan dalam bentuk modul-modul yang sudah tersedia sehingga peserta fokus pada implementasi aplikasi.
- Tidak akan ada integrasi dengan sistem front office hotel yang sebenarnya, praktikum ini akan berfokus pada pembelajaran konsep dan teknologi yang digunakan.

## BAB II DASAR TEORI PRAKTIKUM

### 2.1 Sejarah dan Evolusi Django dan Bootstrap



*Gambar 1.1 Gambar Django dan Bootstrap*

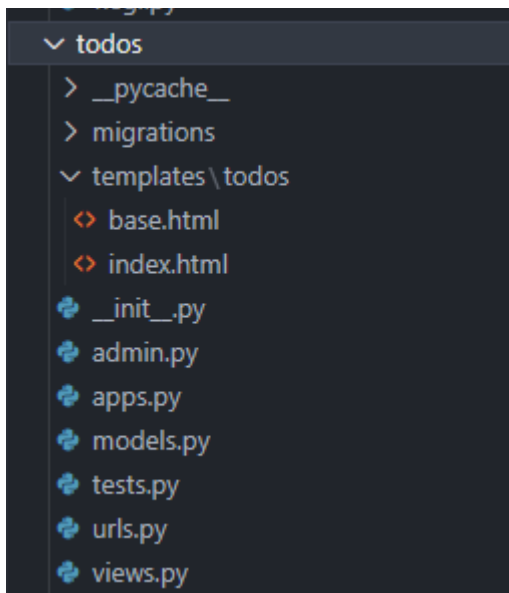
Sejarah dan evolusi Django serta Bootstrap memiliki peran penting dalam pengembangan web modern. Django, sebuah framework web berbasis Python, pertama kali dikembangkan oleh tim di Lawrence Journal-World pada tahun 2003. Dirancang untuk menyederhanakan proses pembuatan aplikasi web, Django menjadi proyek open-source pada Juli 2005. Dikenal dengan filosofi "batteries included",

Django menyediakan beragam fitur bawaan seperti ORM, antarmuka admin, dan sistem autentikasi, membuatnya populer di kalangan pengembang untuk menangani proyek-proyek yang kompleks.

Sementara itu, Bootstrap adalah framework front-end yang dikembangkan oleh Mark Otto dan Jacob Thornton di Twitter. Dirilis sebagai proyek open-source pada Agustus 2011, Bootstrap dirancang untuk mempermudah pengembangan desain web responsif dan mobile-first. Dengan sistem grid yang fleksibel dan berbagai komponen siap pakai, Bootstrap memungkinkan pengembang untuk menciptakan antarmuka pengguna yang menarik tanpa perlu menulis banyak kode CSS dari awal. Sejak itu, Bootstrap telah menjadi salah satu framework front-end yang paling populer di dunia, digunakan oleh jutaan situs web.

Kombinasi Django dan Bootstrap memberikan kekuatan penuh dari backend yang solid dan frontend yang menarik dalam pengembangan web. Django menangani logika bisnis, manajemen data, dan keamanan, sementara Bootstrap menyediakan alat untuk membuat antarmuka pengguna yang responsif dan estetis. Bersama-sama, mereka memungkinkan pengembang untuk membangun aplikasi web yang kuat, efisien, dan ramah pengguna.

## 2.2 Struktur Pemograman



Gambar 1.2 Gambar Codingan Django

### A. Model-View-Template (MVT) dalam Django:

- Django punya sistem kerja bernama Model-View-Template (MVT), yang pisahin tugas aplikasi jadi tiga bagian utama:
  - **Model:** Urus data dan hubungan dengan database.
  - **View:** Atur logika bisnis dan jadi penghubung antara model dan template.
  - **Template:** Atur tampilan antarmuka pengguna dan tunjukkan data dari view.

### B. Organisasi Kode yang Rapi:

- Struktur MVT Django bantu atur kode dengan bagus, pisahin tugas, dan gampang diurus.
- Tiap bagian punya peran yang jelas, bikin kerjasama tim dan perbaikan masalah jadi lebih lancar.

#### **C. Grid Fleksibel di Bootstrap:**

- Bootstrap punya grid yang bisa diatur, berdasarkan konsep 12 kolom.
- Dengan grid ini, pengembang bisa susun tata letak halaman dengan mudah, sesuai kebutuhan desain

#### **D. Komponen Siap Pakai untuk Desain Responsif:**

- Bootstrap punya beragam komponen siap pakai seperti tombol, formulir, navigasi, dll.
- Komponen ini sudah disiapkan untuk buat desain yang responsif dan menarik, tanpa harus mulai dari nol.

#### **E. Pendekatan Terstruktur dalam Pengembangan:**

- Gabungan Django dan Bootstrap bikin pendekatan pengembangan jadi lebih teratur.
- Django atur bisnis dan data, Bootstrap atur tampilan dan interaksi pengguna.
- Dengan pendekatan ini, kompleksitas berkurang, produktivitas meningkat, dan konsistensi dalam pengembangan aplikasi terjaga.

## **2.3 Teori Pemrograman**



*Gambar 1.3 Gambar Kamar Hotel*

Pemrograman adalah proses menciptakan, mendesain, mengimplementasikan, dan memelihara program komputer menggunakan bahasa pemrograman dan algoritma tertentu. Pemrograman adalah keterampilan yang penting dalam pengembangan perangkat lunak dan aplikasi, karena melibatkan pemahaman logika dan struktur data untuk menciptakan solusi permasalahan.

Dalam konteks pengembangan aplikasi Front Office Hotel untuk input data kamar dengan Django dan Bootstrap, teori pemrograman menjadi landasan penting. Django adalah sebuah framework web open-source yang ditulis dalam bahasa pemrograman Python, yang memudahkan pembuatan aplikasi web kompleks dengan pendekatan yang ramah pengguna. Sementara Bootstrap adalah library CSS dan JavaScript yang populer untuk memperindah tampilan aplikasi web dengan desain responsif.

Dengan memahami teori pemrograman, pengembang dapat merancang struktur aplikasi yang efisien, membuat algoritma pengolahan data yang tepat, serta mengoptimalkan kinerja aplikasi Front Office Hotel.

Pemrograman juga memungkinkan pengembang untuk mengimplementasikan fitur-fitur interaktif, validasi data, dan pengelolaan informasi kamar hotel secara praktis dan efektif.

Dengan penguasaan teori pemrograman, pengembang dapat menciptakan aplikasi Front Office Hotel yang handal, aman, dan berkualitas tinggi dalam mengelola data kamar hotel. Teori pemrograman menjadi dasar yang penting dalam mengembangkan aplikasi yang memenuhi kebutuhan pengguna dan memastikan keberhasilan proyek pengembangan perangkat lunak.

## 2.4 Perintah Dasar Python Django dan Bootstrap

- A. **{% extends 'apphotel /base.html' %}**: Ini adalah perintah template Django yang digunakan untuk mewarisi atau memperluas template dasar yang ada. Dalam hal ini, template ini memperluas atau mewarisi template ``base.html`` yang terletak di direktori ``apphotel``.
- B. **{% block title %} ... {% endblock %}**: Ini adalah perintah template Django yang mendefinisikan blok yang dapat diisi dengan konten spesifik dalam template anak. Blok ini, dengan nama ``title``, kemungkinan akan diisi dengan judul halaman spesifik oleh template anak.
- C. **{% load static %}**: Ini adalah perintah template Django yang digunakan untuk memuat tag-tag statis dalam template. Ini memungkinkan penggunaan tag seperti ``{% static %}`` untuk merujuk ke file statis dalam aplikasi.
- D. **{% block content %} ... {% endblock %}**: Ini adalah perintah template Django yang mendefinisikan blok yang akan diisi dengan konten spesifik dalam template anak. Blok ini, dengan nama ``content``, kemungkinan akan diisi dengan konten utama dari halaman spesifik.
- E. **<link rel="stylesheet" href="...">**: Ini adalah tag HTML yang digunakan untuk menyertakan file CSS eksternal dalam halaman HTML. Dalam konteks ini, digunakan untuk menyertakan file CSS Bootstrap dari sumber eksternal.
- F. **<script src="..."></script>**: Ini adalah tag HTML yang digunakan untuk menyertakan file JavaScript eksternal dalam halaman HTML. Dalam konteks ini, digunakan untuk menyertakan file JavaScript Bootstrap dari sumber eksternal.

- G. **{% url ' apphotel:add' %}**: Ini adalah perintah template Django yang digunakan untuk menghasilkan URL absolut untuk pola URL yang diberikan. Dalam hal ini, menghasilkan URL untuk pola URL bernama `'add'` yang terdaftar dalam aplikasi `' apphotel '`
- H. **{% csrf\_token %}**: Ini adalah perintah template Django yang digunakan untuk menyertakan token CSRF dalam formulir. Ini adalah langkah keamanan yang diperlukan ketika mengirimkan formulir di Django untuk melindungi dari serangan CSRF.
- I. **{% for apphotel in apphotel \_list %} ... {% endfor %}**: Ini adalah perintah template Django yang digunakan untuk melakukan iterasi melalui daftar objek yang diberikan. Dalam hal ini, melakukan iterasi melalui daftar objek `' apphotel '`.
- J. **<a href="..." title="Delete">...</a>**: Ini adalah tag HTML `<a>` yang digunakan untuk membuat tautan ke URL tertentu. Dalam hal ini, digunakan untuk membuat tautan untuk menghapus entri `apphotel`.
- K. **<form method="post" action="..."> ... </form>**: Ini adalah tag HTML `<form>` yang digunakan untuk membuat formulir HTML. Dalam hal ini, digunakan untuk membuat formulir untuk menambahkan entri `apphotel` dengan metode `'POST'` dan aksi yang ditentukan.
- L. **<input type="hidden" name="id" id=" apphotel -id" value="">**: Ini adalah tag HTML `<input>` yang digunakan untuk membuat input tersembunyi dalam formulir. Dalam hal ini, digunakan untuk menyimpan ID entri `apphotel` yang akan diedit atau dihapus.
- M. **<button type="submit" class="btn btn-primary">Save changes</button>**: Ini adalah tag HTML `<button>` yang digunakan untuk membuat tombol formulir. Dalam hal ini, digunakan untuk menyimpan perubahan yang dilakukan pada entri `apphotel`.
- N. **{% csrf\_token %}**: Ini adalah perintah template Django yang digunakan untuk menyertakan token CSRF dalam formulir. Ini adalah langkah keamanan yang diperlukan ketika mengirimkan formulir di Django untuk melindungi dari serangan CSRF.

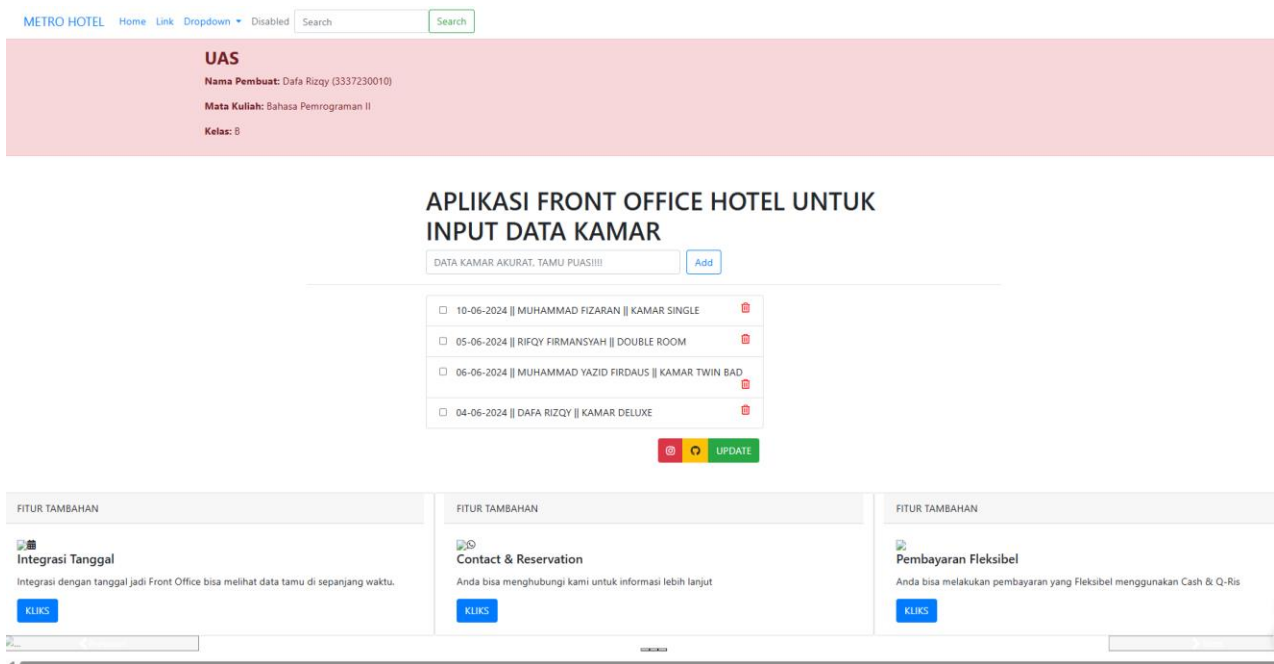


- O. `<div class="list-group-item ..." > ... </div>`: Ini adalah tag HTML `<div>` yang digunakan untuk membuat item daftar dalam daftar grup. Dalam hal ini, digunakan untuk menampilkan entri apshotel dalam daftar.

## BAB III

### PERANCANGAN PROYEK PRAKTIKUM

#### 3.1 Perancangan Algoritma Kalimat Deskriptif



Gambar 2.1 Gambar Interface

#### Proses Utama: Aplikasi Pengembangan Front Office Hotel untuk Input data Kamar:

Entitas Eksternal:

- Pengguna

Data Aliran:

##### A. Inisialisasi:

- Program dimulai dengan menampilkan pesan selamat datang.

##### B. Menu Utama:

- Program langsung menampilkan menu utama:

- Pilihan 1: Tampilkan daftar

- Pilihan 2: Tambahkan baru.
- Pilihan 3: Tandai sebagai selesai.
- Pilihan 4: Hapus
- Pilihan 5: Update
- Pilihan 6: Tombol untuk social media dan github

#### **C. Pemrosesan Menu:**

- Pengguna diminta memilih opsi dari menu.
- Program memeriksa pilihan pengguna dan menjalankan fungsi yang sesuai untuk menangani operasi yang dipilih.

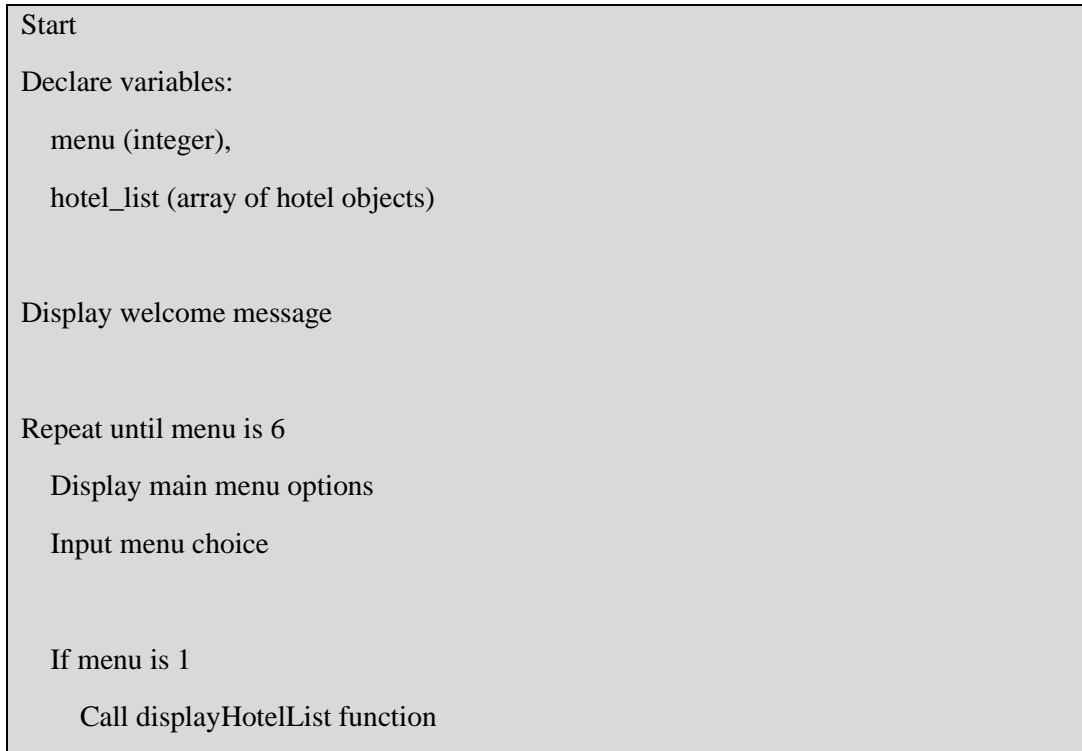
#### **D. Pencetakan Rincian Aplikasi Pengembangan Front Office Hotel untuk Input data Kamar::**

- Setelah pengguna memilih opsi, program mencetak daftar APPHotel beserta tombol-tombol untuk menandai selesai atau menghapus.

#### **E. Ulangi Proses:**

- Program memberikan opsi kepada pengguna untuk kembali ke menu utama atau memulai baru menambahkan APPHotel yang lainnya

### **3.2 Perancangan Algoritma Pseudocode**



```

Else If menu is 2
    Call addNewHotel function
Else If menu is 3
    Call markHotelAsCompleted function
Else If menu is 4
    Call deleteHotel function
Else If menu is 5
    Call updateHotel function
Else If menu is 6
    Display social media and GitHub buttons
Else
    Display "Invalid input! Please enter a valid menu option."

End Repeat

displayHotelList function:
    If hotel_list is empty
        Display "No hotel items found."
    Else
        Display "Hotel List:"
        For each hotel in hotel_list
            Display hotel title and completion status with buttons to mark as completed or
            delete
        Prompt user to go back to main menu or add a new hotel

addNewHotel function:
    Prompt user to enter a title for the new hotel
    Create a new hotel object with the entered title and add it to hotel_list
    Display "Hotel added successfully."
    Prompt user to go back to main menu or add another hotel

```

markHotelAsCompleted function:

Prompt user to select a hotel item to mark as completed

Input the index of the hotel item to mark as completed

Mark the selected hotel item as completed

Display "Hotel marked as completed."

Prompt user to go back to main menu or perform another action

deleteHotel function:

Prompt user to select a hotel item to delete

Input the index of the hotel item to delete

Delete the selected hotel item from hotel\_list

Display "Hotel deleted successfully."

Prompt user to go back to main menu or perform another action

updateHotel function:

Prompt user to select a hotel item to update

Input the index of the hotel item to update

Prompt user to enter a new title for the selected hotel item

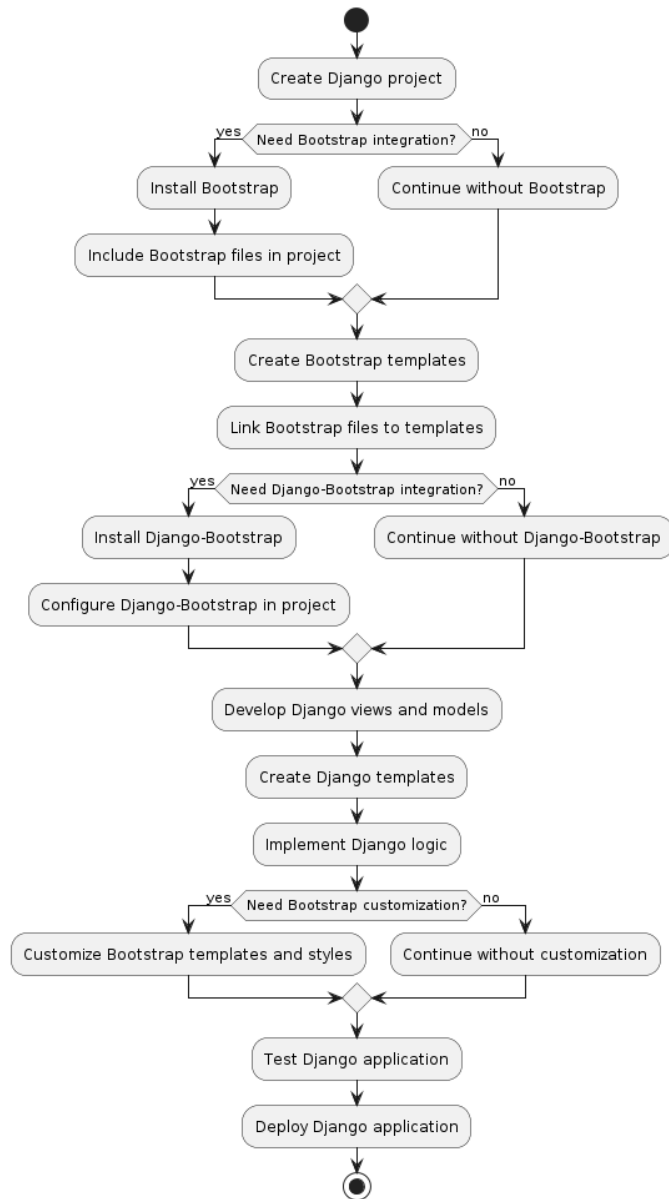
Update the title of the selected hotel item

Display "Hotel updated successfully."

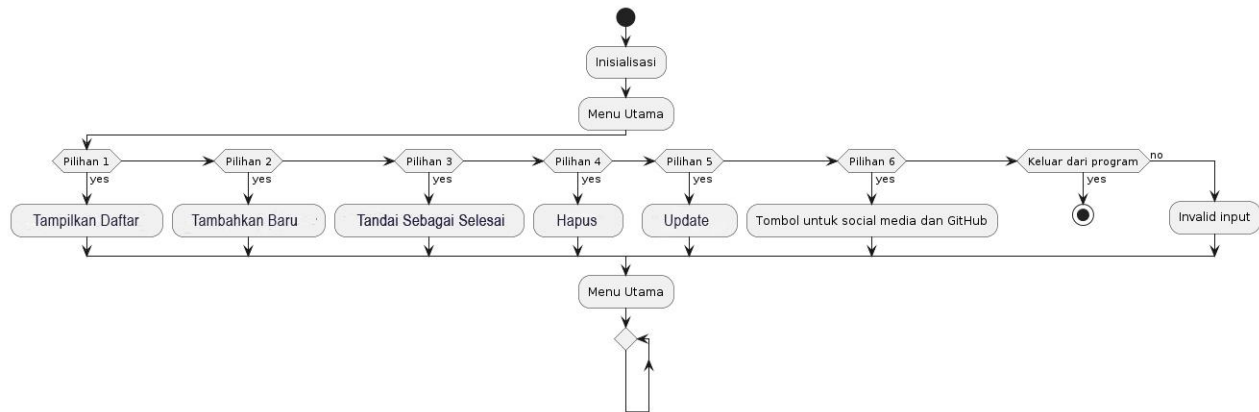
Prompt user to go back to main menu or perform another action

End

### 3.3 Perancangan Flowchart



*Gambar 2.2 Gambar Flowchart Django dan Bootstrap*



Gambar 2.3 Gambar Flowchart To do list Django dan Bootstrap

### 3.4 Perancangan Database

```

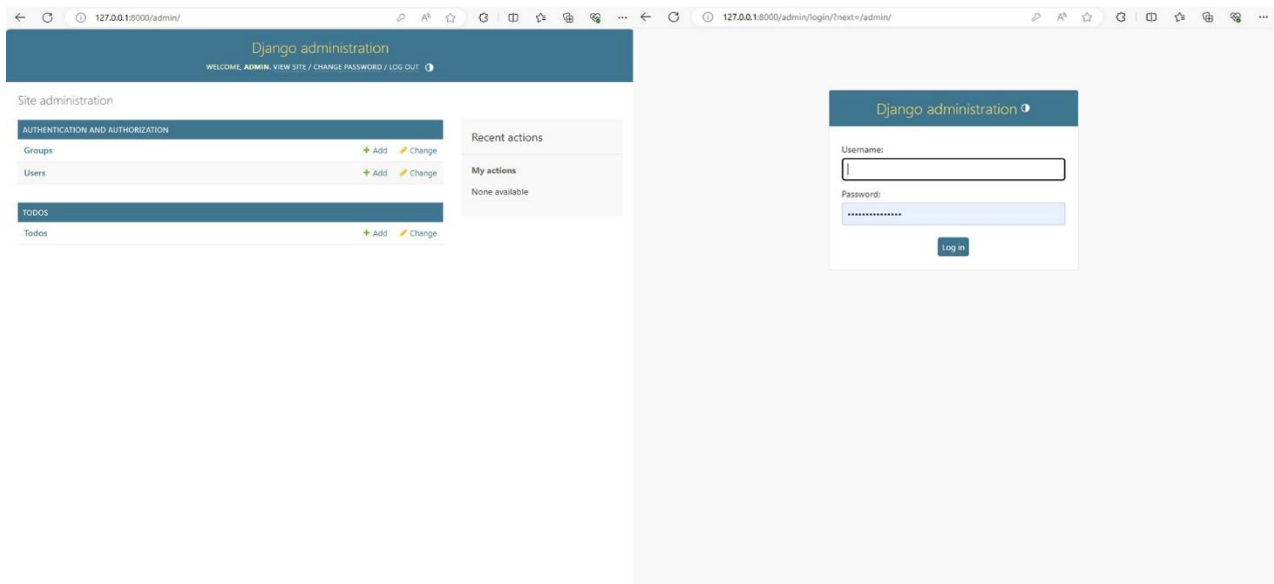
C:\Dev\Tutorial Django\django-todo-develop>python manage.py migrate
new_class._meta.apps.register_model(new_class._meta.app_label, new_class)
Migrations for 'todos':
  todos/migrations/0009_remove_todo_created_at_remove_todo_iscompleted_and_more.py
  - Remove field created_at from todo
  - Remove field isCompleted from todo
  - Remove field title from todo
  - Remove field update_at from todo
  - Alter field id on todo

C:\Dev\Tutorial Django\django-todo-develop>python manage.py migrate
C:\Users\DAFA RIZQY\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCac
he\local-packages\Python311\site-packages\django\db\models\base.py:366: RuntimeWarning: Model 'todos.t
odo' was already registered. Reloading models is not advised as it can lead to inconsistencies, most n
otably with related models.
  new_class._meta.apps.register_model(new_class._meta.app_label, new_class)
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions, todos
Running migrations:
  Applying todos.0009_remove_todo_created_at_remove_todo_iscompleted_and_more... OK

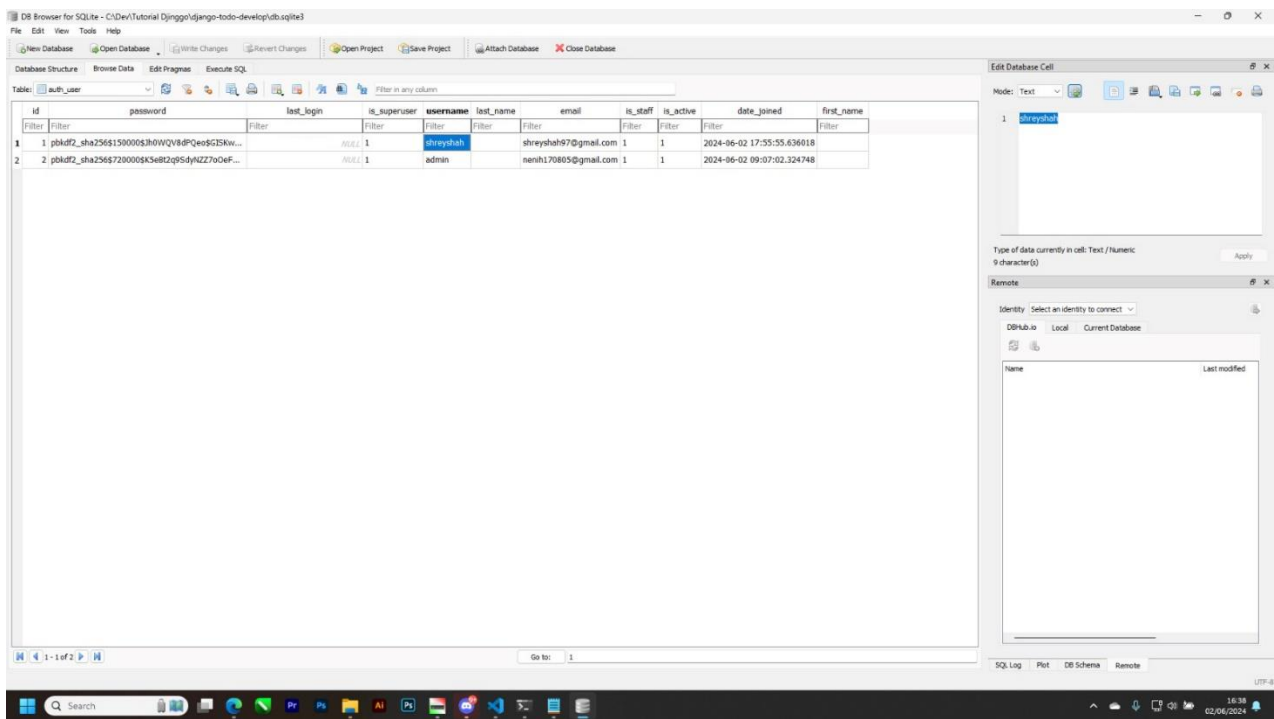
C:\Dev\Tutorial Django\django-todo-develop>python manage.py migrate
C:\Users\DAFA RIZQY\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCac
he\local-packages\Python311\site-packages\django\db\models\base.py:366: RuntimeWarning: Model 'todos.t
odo' was already registered. Reloading models is not advised as it can lead to inconsistencies, most n
otably with related models.
  new_class._meta.apps.register_model(new_class._meta.app_label, new_class)
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions, todos
Running migrations:
  No migrations to apply.

C:\Dev\Tutorial Django\django-todo-develop>python manage.py createsuperuser
C:\Users\DAFA RIZQY\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCac
he\local-packages\Python311\site-packages\django\db\models\base.py:366: RuntimeWarning: Model 'todos.t
odo' was already registered. Reloading models is not advised as it can lead to inconsistencies, most n
otably with related models.
  new_class._meta.apps.register_model(new_class._meta.app_label, new_class)
Username (leave blank to use 'dafarizqy'): admin
Email address: nenih178805@gmail.com
Password:
Password (again):
Superuser created successfully.

C:\Dev\Tutorial Django\django-todo-develop>
  
```

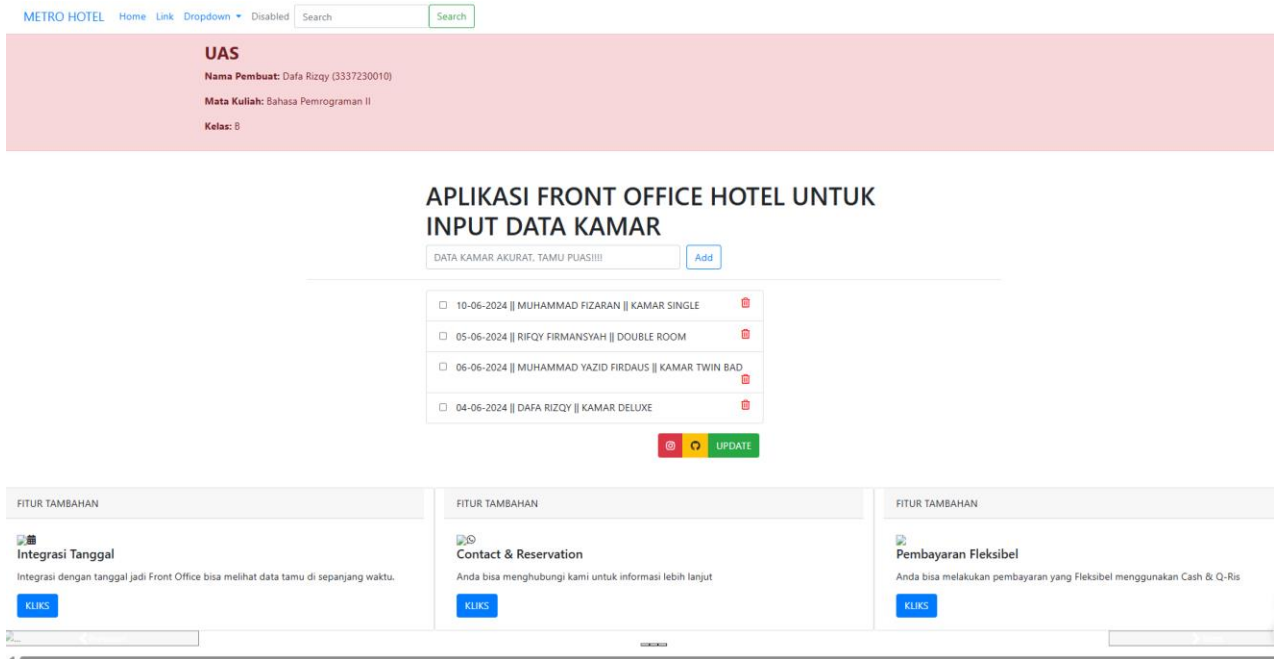


Gambar 3.2 Tampilan Login Database Menggunakan SQ lite3



Gambar 3.3 Tampilan Database Menggunakan SQ lite3

### 3.5 Perancangan Desain Interface



Gambar 4.1 Menu Base

## BAB IV

### PEMBAHASAN DAN HASIL PROYEK PRAKTIKUM

#### 4.1 Coding Program

```
{% extends 'apphotel/base.html' %}

{% block title %}
<title>APLIKASI FRONT OFFICE HOTEL UNTUK INPUT DATA KAMAR</title>
{% endblock %}

{% load static %}

{% block content %}

<!-- NAVBAR -->
<nav class="navbar navbar-expand-lg bg-body-tertiary">
```



```

<div class="container-fluid">

  <a class="navbar-brand" href="#">METRO HOTEL</a>

  <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">

    <span class="navbar-toggler-icon"></span>

  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">

    <ul class="navbar-nav me-auto mb-2 mb-lg-0">

      <li class="nav-item">

        <a class="nav-link active" aria-current="page" href="#">Home</a>

      </li>

      <li class="nav-item">

        <a class="nav-link" href="#">Link</a>

      </li>

      <li class="nav-item dropdown">

        <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-
expanded="false">

          Dropdown

        </a>

        <ul class="dropdown-menu">

          <li><a class="dropdown-item" href="#">Action</a></li>

          <li><a class="dropdown-item" href="#">Another action</a></li>

          <li><hr class="dropdown-divider"></li>

          <li><a class="dropdown-item" href="#">Something else here</a></li>

        </ul>

      </li>

      <li class="nav-item">

        <a class="nav-link disabled" aria-disabled="true">Disabled</a>

      </li>

    </ul>

    <form class="d-flex" role="search">

      <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">

      <button class="btn btn-outline-success" type="submit">Search</button>

    </form>

```

```

</div>

</div>

</nav>

<!-- ALERT -->

<div class="alert alert-danger d-flex align-items-center" role="alert">

  <svg class="bi flex-shrink-0 me-2" role="img" aria-label="Danger:"><use xlink:href="#exclamation-
triangle-fill"/></svg>

  <div>

    <h3 style="font-weight: bold;">UAS</h3>

    <p><strong>Nama Pembuat:</strong> Dafa Rizqy (3337230010)</p>

    <p><strong>Mata Kuliah:</strong> Bahasa Pemrograman II</p>

    <p><strong>Kelas:</strong> B</p>

  </div>

</div>

<div class="container">

  <!-- title row -->

  <div class="row">

    <div class="offset-md-2 col-lg-9">

      <div class="page-header">

        <h1>

          APLIKASI FRONT OFFICE HOTEL UNTUK INPUT DATA KAMAR

        </h1>

      </div>

    </div>

  </div>

</div>

<!-- Add a apphotel row -->

```

```

<div class="row">

  <div class="offset-md-2 col-lg-9">

    <form method="post" action="{% url 'apphotel:add' %}">

      {% csrf_token %}

      <div class="form-row">

        <div class="col-md-6">

          <input type="text" class="form-control" name="title" placeholder="DATA KAMAR AKURAT,
TAMU PUAS!!!!" required>

        </div>

        <div class="col-md-6">

          <button type="submit" name="submit" class="btn btn-outline-primary">

            Add

          </button>

        </div>

      </div>

    </form>

  </div>

</div>

<hr />

<!-- apphotel list row -->

<div class="row">

  <div class="offset-md-2 col-lg-6">

    <div class="list-group">

      {% for apphotel in apphotel _list %}

        <div class="list-group-item {% if apphotel.isCompleted %} apphotel -complete {% endif %}">

          <form style="display: inline;" method="post" action="{% url 'apphotel:update' apphotel.id %}">

            {% csrf_token %}

            <input type="checkbox" name="isCompleted" onchange="this.form.submit()" {% if
apphotel.isCompleted %} checked

              {% endif %} class=" apphotel -status-checkbox"

              title="{% if not apphotel.isCompleted %} mark as done {% else %} mark undone {% endif %}">

          </form>

          {{ apphotel.title }}

        </div>

      {% endfor %}

    </div>

  </div>

</div>

```

```

<a href="{% url 'apphotel:delete' apphotel.id %}" title="Delete">

    <i class="far fa-trash-alt"></i>

</a>

</div>

{% endfor %}

</div>

</div>

</div>

</div>

<!-- Button group row -->

<div class="row justify-content-center mt-3">

    <div class="col-lg-5 offset-lg-1 d-flex justify-content-center">

        <div class="btn-group" role="group" aria-label="Basic mixed styles example">

            <a href="https://www.instagram.com/daffansyah_rizqy/" class="btn btn-danger button-spacing"><i
            class="fa fa-instagram" aria-hidden="true"></i></a>

            <a href="https://github.com/DaffansyahRizqy" class="btn btn-warning button-spacing"><i class="fa fa-
            github" aria-hidden="true"></i></a>

            <a href="url_ke_halaman_kanan" class="btn btn-success button-spacing">UPDATE</a>

        </div>

    </div>

</div>

<!-- Edit Modal -->

<div class="modal fade" id="editModal" tabindex="-1" role="dialog" aria-labelledby="editModalLabel"
aria-hidden="true">

    <div class="modal-dialog" role="document">

        <div class="modal-content">

            <form method="post" action="">

                {% csrf_token %}

                <div class="modal-header">

                    <h5 class="modal-title" id="editModalLabel">Edit apphotel </h5>

                    <button type="button" class="close" data-dismiss="modal" aria-label="Close">

                        <span aria-hidden="true">&times;</span>

                    </button>

                </div>

            </form>

        </div>

    </div>

</div>

```

```

</div>

<div class="modal-body">

  <input type="hidden" name="id" id=" apphotel -id" value="">

  <div class="form-group">

    <label for="apphotel-title">Title</label>

    <input type="text" class="form-control" name="title" id=" apphotel -title" value="" required>

  </div>

</div>

<div class="modal-footer">

  <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>

  <button type="submit" class="btn btn-primary">Save changes</button>

</div>

</form>

</div>

</div>

</div>

<!-- header -->

<div class="mt-5">

  <div class="row">

    <div class="card col">

      <div class="card-header">

        FITUR TAMBAHAN

      </div>

      <div class="card-body">

        <img src="">

        <h5 class="card-title">Integrasi Tanggal</h5>

        <p class="card-text">Integrasi dengan tanggal jadi Front Office bisa melihat data tamu di sepanjang waktu.</p>

        <a href="#" class="btn btn-primary">KLIK</a>

      </div>

    </div>

    <div class="card col">

      <div class="card-header">

```

#### FITUR TAMBAHAN

</div>

<div class="card-body">



<h5 class="card-title">Contact & Reservation </h5>

<p class="card-text">Anda bisa menghubungi kami untuk informasi lebih lanjut </p>

<a href="https://wa.me/6283107270104" class="btn btn-primary">KLIKS</a>

</div>

</div>

<div class="card col">

<div class="card-header">

#### FITUR TAMBAHAN

</div>

<div class="card-body">

</i>

<h5 class="card-title">Pembayaran Fleksibel</h5>

<p class="card-text">Anda bisa melakukan pembayaran yang Fleksibel menggunakan Cash & Q-Ris</p>

<a href="#" class="btn btn-primary">KLIKS</a>

</div>

</div>

</div>

</div>

<!-- NAV NEXT DLLNYA-->

<div id="carouselExampleDark" class="carousel carousel-dark slide">

<div class="carousel-indicators">

<button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="0" class="active" aria-current="true" aria-label="Slide 1"></button>

<button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="1" aria-label="Slide 2"></button>

<button type="button" data-bs-target="#carouselExampleDark" data-bs-slide-to="2" aria-label="Slide 3"></button>

</div>

```

<div class="carousel-inner">

  <div class="carousel-item active" data-bs-interval="10000">

    <div class="carousel-caption d-none d-md-block">

      <h5>First slide label</h5>

      <p>Some representative placeholder content for the first slide.</p>

    </div>

  </div>

  <div class="carousel-item" data-bs-interval="2000">

    <div class="carousel-caption d-none d-md-block">

      <h5>Second slide label</h5>

      <p>Some representative placeholder content for the second slide.</p>

    </div>

  </div>

  <div class="carousel-item">

    <div class="carousel-caption d-none d-md-block">

      <h5>Third slide label</h5>

      <p>Some representative placeholder content for the third slide.</p>

    </div>

  </div>

  <div>

    <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleDark" data-bs-
slide="prev">

      <span class="carousel-control-prev-icon" aria-hidden="true"></span>

      <span class="visually-hidden">Previous</span>

    </button>

    <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleDark" data-bs-
slide="next">

      <span class="carousel-control-next-icon" aria-hidden="true"></span>

```

```
<span class="visually-hidden">Next</span>

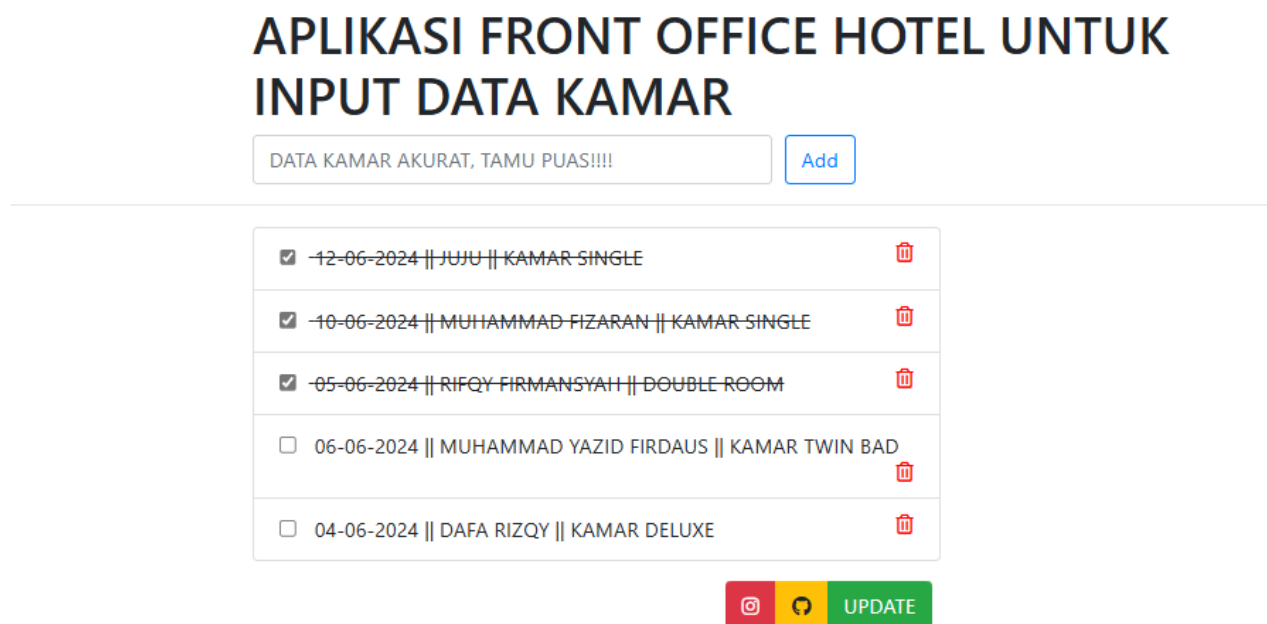
</button>

</div>

{% endblock %}

3.
```

## 4.1 Hasil Coding



Gambar 4.1 Hal Menu Menambahkan, Menghapus, Menandai, Mengupdate To do list

CRUD merupakan singkatan dari Create, Read, Update, dan Delete. Ini adalah operasi dasar yang digunakan dalam pengembangan perangkat lunak untuk mengelola data pada suatu sistem. Berikut adalah penjelasan singkat untuk setiap operasi CRUD:

- **Create:** Membuat data baru dalam sistem dengan menambahkan entri baru ke dalam database atau membuat objek baru dalam aplikasi.
- **Read:** Membaca atau mengambil data yang sudah ada dari sistem, biasanya ditampilkan melalui antarmuka pengguna atau API.
- **Update:** Memperbarui atau mengubah data yang sudah ada dalam sistem dengan mengedit nilai-nilai tertentu dari entri data atau objek yang sudah ada.
- **Delete:** Menghapus data yang sudah ada dari sistem dengan menghapus entri data atau objek yang tidak diperlukan lagi dari database atau aplikasi.



Dengan demikian, operasi CRUD adalah dasar dari banyak aplikasi perangkat lunak yang memungkinkan pengguna untuk berinteraksi dengan data dan mengelola informasi secara fleksibel dan efisien.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Aplikasi Front Office Hotel yang dikembangkan menggunakan Django dan Bootstrap memberikan kemudahan dalam manajemen data kamar hotel. Fitur-fitur yang diimplementasikan memungkinkan pengguna untuk:

- A. **Input Data Kamar:** Pengguna dapat dengan mudah menambahkan data kamar hotel melalui antarmuka aplikasi yang intuitif.
- B. **List Data Kamar:** Aplikasi menyediakan tampilan list kamar hotel dengan informasi yang jelas dan navigasi yang nyaman.
- C. **Edit dan Update Data Kamar:** Pengguna dapat melakukan perubahan data kamar hotel dengan mudah dan langsung melalui fitur edit yang tersedia.
- D. **Kombinasi Django dan Bootstrap:** Penggunaan Django sebagai backend memberikan kestabilan dan kemudahan dalam pengolahan data, sementara menggunakan Bootstrap sebagai frontend membuat tampilan aplikasi menjadi responsif dan menarik.

Dengan demikian, aplikasi Front Office Hotel ini diharapkan dapat meningkatkan efisiensi dan pengalaman pengguna dalam manajemen data kamar hotel.

#### **5.2 Saran**

- A. **Penyempurnaan Antarmuka Pengguna (UI/UX):**
  - Mengoptimalkan tata letak elemen, warna, dan desain antarmuka pengguna agar lebih intuitif dan menarik.
  - Memperhatikan responsivitas antarmuka agar dapat diakses dengan baik di berbagai perangkat.
- B. **Peningkatan Fungsionalitas Aplikasi:**
  - Menambah fitur pencarian kamar berdasarkan kriteria tertentu untuk mempermudah pengguna dalam menemukan informasi yang dibutuhkan.
  - Implementasi fitur notifikasi atau reminder untuk manajemen kamar yang membutuhkan perhatian khusus.
  - Integrasi dengan sistem pembayaran online untuk memfasilitasi reservasi kamar secara langsung.
- C. **Pemeliharaan Aplikasi dan Keamanan Data:**
  - Melakukan pemeliharaan rutin terhadap aplikasi, termasuk pembaruan teknologi yang digunakan seperti Django dan Bootstrap.
  - Memastikan keamanan data kamar hotel dengan menerapkan lapisan keamanan yang kuat, termasuk enkripsi data sensitif.

**Pelatihan Pengguna:**

- Mengadakan sesi pelatihan untuk pengguna aplikasi, terutama bagi staf front office hotel, agar dapat mengoptimalkan penggunaan aplikasi dan memahami fitur-fitur yang disediakan.

**Feedback dan Evaluasi:**

- Mendorong pengguna untuk memberikan feedback mengenai pengalaman penggunaan aplikasi, serta melakukan evaluasi secara berkala untuk mengetahui kekurangan dan kelebihan yang dapat diperbaiki.

Dengan menerapkan saran-saran di atas, diharapkan aplikasi Front Office Hotel dapat terus berkembang dan memberikan nilai tambah bagi penggunanya.

# DAFTAR PUSTAKA

## 1. Dokumentasi Resmi:

- Dokumentasi Django: <https://docs.djangoproject.com/en/5.0/intro/tutorial01/>
- Dokumentasi Bootstrap: <https://getbootstrap.com/docs/5.0/getting-started/introduction/>

## 2. Tutorial dan Panduan:

- Tutorial Django untuk Pemula: <https://realpython.com/tutorials/django/>
- Membuat Aplikasi Web dengan Bootstrap: <https://www.w3schools.com/bootstrap/>
- Membangun Aplikasi Front Office Hotel dengan Django dan Bootstrap: <https://www.youtube.com/watch?v=yenjz1Wv9Yo>

## 3. Buku dan Artikel:

- Django 3.x Essential Recipes: <https://www.packtpub.com/product/web-development-with-django/9781839212505>
- Building Web Applications with Django and Bootstrap: <https://www.apress.com/gp/web-development>
- Designing and Developing Responsive Web Applications with Bootstrap: <https://www.packtpub.com/product/learning-bootstrap/9781782161844>

## 4. Komunitas dan Forum Online:

- Komunitas Django: <https://stackoverflow.com/questions/629551/how-to-query-as-group-by-in-django>
- Forum Bootstrap: <https://stackoverflow.com/questions/tagged/bootstrap-4>
- Subreddit Django: <https://www.reddit.com/r/django/>