

LAPORAN PRATIKUM
SISTEM MANAJAMEN RESERVASI KAMAR BERBASIS
QR CODE

Dibuat untuk memenuhi tugas kelompok mata kuliah Komputasi Numerik dan
Statistika

Dosen Pengampu: Iqbal Lafifa Zulfa, S.Si., M.Si



Disusun oleh:

- | | |
|---------------------------------|--------------|
| 1. Mia Amalia Damayanti | (2301010014) |
| 2. Muhammad Faisyal Nur Pratama | (2301010058) |
| 3. Rizza Widi Nugraha | (2301010015) |

PROGRAM STUDI TEKNOLOGI REKAYASA PERANGKAT
LUNAK

POLITEKNIK MARDIRA INDONESIA

2024/2025

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era digitalisasi, teknologi semakin berkembang untuk mempermudah berbagai aktivitas manusia, termasuk dalam sistem reservasi kamar. Sistem manual yang masih menggunakan pencatatan konvensional sering kali menyebabkan kesalahan dalam pencatatan dan keterlambatan dalam proses pemesanan. Oleh karena itu, dibutuhkan sistem manajemen reservasi kamar berbasis QR Code yang dapat meningkatkan efisiensi dan akurasi dalam proses pemesanan kamar.

1.2 Rumusan Masalah

1. Bagaimana cara membangun sistem reservasi kamar yang lebih efisien dan akurat?
2. Bagaimana implementasi QR Code dapat membantu dalam proses pemesanan kamar?
3. Bagaimana sistem dapat memastikan QR Code yang sudah digunakan tidak bisa dipakai kembali?

1.3 Tujuan

1. Mengembangkan sistem reservasi kamar berbasis QR Code untuk mempermudah proses pemesanan.
2. Memastikan keakuratan data pemesanan dengan memanfaatkan teknologi QR Code.
3. Memberikan fitur yang memudahkan admin dalam mengelola data reservasi kamar.

BAB II

RENCANA & PELAKSANAAN

2.1 Waktu dan Tempat

No	Tanggal	Tempat
1.	28 Januari 2025	Basecamp 1
2.	30 Januari 2025	Kampus Politeknik Mardira Indonesia
3.	1 Febuari 2025	Basecamp 2
4.	2 Febuari 2025	Basecamp 2

2.2 Alat dan Bahan

No.	Perangkat Keras	Perangkat Lunak
1.	2 Laptop	Boostrap 5 (Front-end)
2.	Handphone	Chart JS (Javascript)
3.	Jaringan Wifi	Python Flask (Backend)
4.		Library QR Code Generator (Python)
5.		Library Pillow (Python)
6.		Library Werkzeug (Python)
7.		Visual Studio Code
8.		Chrome Browser
9.		Github

2.3 Algoritma Program

1. Admin membuat dan mengelola QR Code untuk setiap kamar.
2. User memindai QR Code untuk melakukan reservasi.
3. Sistem mengarahkan user ke form pengisian data dan melakukan pembayaran yang telah tersedia.
4. Jika QR Code telah digunakan, user akan diminta untuk memilih QR Code lain.
5. Admin dapat melakukan input data manual untuk reservasi offline.
6. Admin dapat mengosongkan kamar dan otomatis merubah status kamar jadi kosong/tersedia untuk di scan.
7. Admin dapat membatalkan reservasi jika diperlukan.

8. Admin dapat melihat laporan statistik penggunaan kamar.

2.4 Proses Produksi

1. Pembuatan database untuk menyimpan informasi kamar dan reservasi.
2. Pengembangan backend menggunakan Flask untuk menangani request dan response.
3. Implementasi QR Code untuk menghubungkan user dengan form reservasi.
4. Pengembangan frontend menggunakan HTML dan Bootstrap.
5. Pengujian sistem untuk memastikan fungsionalitas berjalan dengan baik.

BAB III

PEMBAHASAN

3.1 Hasil dan Analisis

1. Sistem berhasil dibangun dengan fitur utama pemindaian QR Code untuk reservasi.
2. QR Code yang sudah digunakan tidak dapat dipakai kembali.
3. Admin dapat mengelola pemesanan, baik online maupun offline.
4. Dashboard admin dilengkapi dengan chart penggunaan kamar.
5. User dapat langsung melakukan pembayaran dengan metode pembayaran yang sudah di sediakan.

BAB IV

PENUTUP

4.1 Kesimpulan

Sistem manajemen reservasi kamar berbasis QR Code dapat meningkatkan efisiensi dan akurasi dalam pemesanan kamar. Teknologi QR Code memberikan kemudahan dalam reservasi dan mengurangi kemungkinan kesalahan pencatatan.

4.2 Saran

1. Pengembangan fitur notifikasi otomatis untuk konfirmasi reservasi.
2. Penambahan fitur manajemen pelanggan untuk meningkatkan layanan.
3. Penambahan fitur login dan registrasi mencegah penyalahgunaan data.
4. Terintegrasi dengan database seperti MySQL agar data tidak hilang.

BAB V

DAFTAR PUSTAKA

Bootstrap 5 framework front-end.

<https://getbootstrap.com/>

Flask adalah framework web ringan berbasis Python.

<https://flask.palletsprojects.com/>

Pillow pustaka Python untuk manipulasi gambar, termasuk membaca, mengedit, dan menyimpan berbagai format gambar.

<https://pillow.readthedocs.io/>

Werkzeug pustaka WSGI yang digunakan untuk menangani permintaan HTTP dalam aplikasi web berbasis Python.

<https://werkzeug.palletsprojects.com/>

QRCode pustaka Python yang digunakan untuk membuat kode QR secara dinamis.

<https://pypi.org/project/qrcode/>

Chart.js pustaka JavaScript open-source yang digunakan untuk membuat grafik interaktif dan dinamis di aplikasi web.

<https://www.chartjs.org/>

BAB VI

LAMPIRAN

```
1 from flask import Flask, render_template, request, jsonify, send_from_directory
2 import qrcode
3 from io import BytesIO
4 import base64
5 import os
6 from werkzeug.utils import secure_filename
7
```

```
1 # Konfigurasi folder upload dan ekstensi file yang diizinkan
2 UPLOAD_FOLDER = os.path.join(os.getcwd(), 'uploads')
3 ALLOWED_EXTENSIONS = ('png', 'jpg', 'jpeg', 'pdf')
4
5 app = Flask(__name__)
6 app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
7
8 if not os.path.exists(UPLOAD_FOLDER):
9     os.makedirs(UPLOAD_FOLDER)
10
11 # Fungsi untuk memeriksa apakah file yang diunggah memiliki ekstensi yang diperbolehkan
12 def allowed_file(filename):
13     return '.' in filename and filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS
14
15 # Data awal status kamar
16 tables = [
17     {'id': i, 'status': 'kosong', 'count': 0} for i in range(201, 209)
18 ]
```

```
1 @app.route('/')
2 def index():
3     """
4     Halaman utama yang menampilkan daftar kamar beserta QR Code untuk pemindaian.
5     """
6     table_data = []
7     for table in tables:
8         qr = qrcode.QRCode()
9         qr_data = f"http://192.168.0.11:5000/update_status/{table['id']}"
10         qr.add_data(qr_data)
11         qr.make(fit=True)
12         img = BytesIO()
13         qr_img = qr.make_image(fill_color="black", back_color="white")
14         qr_img.save(img, format="PNG")
15         img_base64 = base64.b64encode(img.getvalue()).decode('utf-8')
16         table_data.append({'table': table, 'qr': img_base64})
17     return render_template('index.html', table_data=table_data)
18
```

```
1 @app.route('/get_status', methods=['GET'])
2 def get_status():
3     """
4     API untuk mendapatkan status semua kamar.
5     """
6     for table in tables:
7         table['show_button'] = table['status'] == 'terisi'
8     return jsonify({'tables': tables})
9
```

```

1 @app.route('/update_status/<int:table_id>', methods=['GET', 'POST'])
2 def update_status(table_id):
3     """
4     Endpoint untuk memperbarui status kamar.
5     - User biasa hanya bisa scan kamar yang kosong.
6     - Admin dapat melihat detail dan mengosongkan kamar yang terisi.
7     """
8     table = next((t for t in tables if t['id'] == table_id), None)
9     if not table:
10        return jsonify({"success": False, "message": "Kamar tidak ditemukan"}), 404
11
12    is_admin = request.args.get('is_admin', 'false').lower() == 'true'
13
14    if table['status'] == 'terisi' and not is_admin:
15        return """
16        <script>
17            alert("Anda hanya bisa mengosongkan kamar yang kosong!");
18            window.location.href = "/";
19        </script>
20        """
21
22    if request.method == 'POST':
23        if is_admin and request.form.get('reset_kamar'):
24            table['status'] = 'kosong'
25            table.pop('pelanggan', None)
26
27            return """
28            <script>
29                alert("Kamar {table_id} berhasil dikosongkan!");
30                window.location.href = "/update_status/{table_id}?is_admin=true";
31            </script>
32            """
33
34        nama_pelanggan = request.form.get('nama_pelanggan')
35        nomor_telepon = request.form.get('nomor_telepon')
36        check_in = request.form.get('check_in')
37        check_out = request.form.get('check_out')
38        metode_pembayaran = request.form.get('metode_pembayaran')
39        file = request.files.get('bukti_pembayaran')
40
41        if not all([nama_pelanggan, nomor_telepon, check_in, check_out, metode_pembayaran]):
42            return jsonify({"success": False, "message": "Data tidak lengkap"}), 400
43
44        if not file or not allowed_file(file.filename):
45            return jsonify({"success": False, "message": "File tidak valid"}), 400
46
47        filename = secure_filename(file.filename)
48        file_path = os.path.join(app.config['UPLOAD_FOLDER'], filename)
49        file.save(file_path)
50
51        table['status'] = 'terisi'
52        table['pelanggan'] = {
53            "nama": nama_pelanggan,
54            "telepon": nomor_telepon,
55            "check_in": check_in,
56            "check_out": check_out,
57            "metode_pembayaran": metode_pembayaran,
58            "bukti_pembayaran": filename
59        }
60
61        if request.method == 'POST':
62            table['status'] = 'terisi'
63            table['count'] += 1 # Peningkatkan jumlah penggunaan kamar
64
65        return render_template('detail_pelanggan.html', table=table)
66
67    pelanggan = table.get('pelanggan', {})
68    return render_template('form_input.html', table=table, pelanggan=pelanggan, is_admin=is_admin)
69

```

```

1 @app.route('/cancel_menginap/<int:table_id>', methods=['GET'])
2 def cancel_menginap(table_id):
3     """
4     Endpoint untuk membatalkan menginap dan mengosongkan kamar.
5     """
6     table = next((t for t in tables if t['id'] == table_id), None)
7     if not table:
8         return jsonify({"success": False, "message": "Kamar tidak ditemukan"}), 404
9
10    if table['status'] == 'kosong':
11        return jsonify({"success": False, "message": "Kamar sudah kosong"}), 400
12
13    table['status'] = 'kosong'
14    table.pop('pelanggan', None)
15    table['count'] = 0
16
17    return """
18    <script>
19        alert("Kamar {table_id}, Order Berhasil Dibatalakan");
20        window.location.href = "/update_status/{table_id}?is_admin=true";
21    </script>
22    """
23

```

```

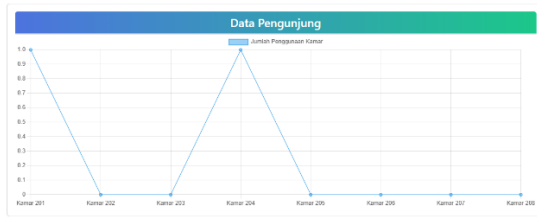
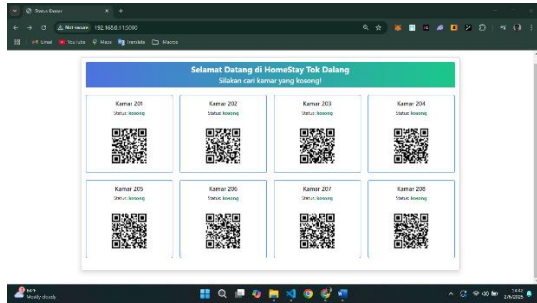
1 @app.route('/get_table_details/<int:table_id>', methods=['GET'])
2 def get_table_details(table_id):
3     """
4     API untuk mendapatkan detail kamar berdasarkan ID.
5     """
6     table = next((t for t in tables if t['id'] == table_id), None)
7     if not table or table['status'] == 'kosong':
8         return jsonify({"success": False, "message": "Data tidak ditemukan"})
9
10    return jsonify({"success": True, "table": table})
11
12 @app.route('/admin')
13 def admin():
14    return render_template('admin.html', tables=tables)
15
16 @app.route('/room_usage_data', methods=['GET'])
17 def room_usage_data():
18    """Mengirim data penggunaan kamar dalam format JSON"""
19    data = [{"id": table['id'], "count": table['count']} for table in tables]
20    return jsonify(data)
21

```

```

1 @app.route('/uploads/<path:filename>')
2 def uploaded_file(filename):
3     """Mengirim file gambar atau dokumen dari folder uploads."""
4     try:
5         return send_from_directory(app.config['UPLOAD_FOLDER'], filename)
6     except FileNotFoundError:
7         abort(404)
8
9 @app.route('/uploads/')
10 def list_uploaded_files():
11     """Menampilkan daftar file dalam folder uploads."""
12     files = os.listdir(app.config['UPLOAD_FOLDER'])
13     file_urls = [f'<a href="/uploads/{file}">{file}</a>' for file in files]
14     return "Daftar File Upload<br>" + "<br>".join(file_urls)
15
16 if __name__ == '__main__':
17     app.run(host='192.168.0.11', port=5000, debug=True)

```



Input Data Pelanggan untuk Kamar 201

Data Pelanggan Kamar: 201

Nama Pelanggan: (1212)


Telepon: 0853337353

Check-in: 2025-02-01T14:42

Check-out: 2025-02-07T14:42

Metode Pembayaran: (DANA)

Bukti Pembayaran:



[Kembali ke Kamar](#) [Tambah Data](#) [Simpan](#)

Input Data Pelanggan untuk Kamar 202

Nama:

No. Trip:

Check-in (Tanggal dan Waktu):

Check-out (Tanggal dan Waktu):

Metode Pembayaran:

Unggah Bukti Pembayaran:

[Simpan Data](#)

Kelola Kamar

ID Kamar	Status	Aksi
201	terisi	Kosong
202	kosong	Kosong
203	kosong	Kosong
204	terisi	Kosong
205	kosong	Kosong
206	kosong	Kosong
207	kosong	Kosong
208	kosong	Kosong



