
PENERAPAN ARSITEKTUR MODEL VIEW CONTROLLER PADA APLIKASI RESERVASI PENGINAPAN BERBASIS WEB (STUDI KASUS: PENGINAPAN PANTAI UMA WANAU PRIVATE BEACH)

Yehuda Pantow¹⁾, Djoni Hatidja²⁾, Christian A. J. Soewoeh³⁾, Eliasta Ketaren⁴⁾

Program Studi Sistem Informasi

Universitas Sam Ratulangi

Jl. Kampus Unsrat, Bahu-Kleak, Manado

email: yehudapantow106@student.unsrat.ac.id¹⁾, dhatidja@unsrat.ac.id²⁾, christian.suwuh@unsrat.ac.id³⁾, eliasketaren@unsrat.ac.id⁴⁾

Abstrak

Pantai Uma Wanau Private Beach merupakan destinasi wisata yang terletak di Desa Makalisung, Kecamatan Kombi, Kabupaten Minahasa. Pantai Uma Wanau menyediakan beberapa fasilitas yang dapat dinikmati salah satunya adalah kamar penginapan bagi pelanggan atau tamu yang ingin menginap. Namun, terdapat masalah yang dihadapi oleh pihak pengelola yaitu pelayanan reservasi kamar yang masih dilakukan secara konvensional. Tujuan dari penelitian ini adalah untuk merancang dan membangun sebuah aplikasi reservasi penginapan berbasis web dengan menerapkan arsitektur MVC, sehingga diharapkan dapat meningkatkan efisiensi pengelolaan reservasi penginapan bagi pihak pengelola Pantai Uma Wanau Private Beach. Sistem ini dikembangkan metode pengembangan Rapid Application Development (RAD) menggunakan framework Laravel. Hasil pengujian menunjukkan bahwa aplikasi ini dapat berjalan dengan baik dan memenuhi kebutuhan fungsional yang diharapkan, sehingga berdasarkan hasil tersebut dapat disimpulkan penerapan arsitektur MVC dengan metode pengembangan RAD dapat menghasilkan aplikasi reservasi yang dapat meningkatkan efisiensi pengelolaan data reservasi bagi pihak pengelola Pantai Uma Wanau Private Beach.

Kata Kunci: Reservasi Penginapan, MVC, Rapid Application Development (RAD), Web

1. Pendahuluan

Pantai Uma Wanau Private Beach merupakan destinasi wisata yang terletak di Desa Makalisung, Kecamatan Kombi, Kabupaten Minahasa. Pantai Uma Wanau menyediakan beberapa fasilitas yang dapat dinikmati salah satunya adalah kamar penginapan bagi customer atau tamu yang ingin menginap. Berdasarkan hasil observasi dan wawancara dengan pihak pengelola, jumlah pengunjung yang datang ke Pantai Uma Wanau Private Beach terus mengalami peningkatan setiap tahunnya. Hal ini menunjukkan Pantai Uma Wanau Private Beach sebagai salah satu kontributor dalam sektor pariwisata di Kabupaten Minahasa. Namun, terdapat masalah yang dihadapi oleh pihak pengelola yaitu pelayanan reservasi kamar yang masih dilakukan secara konvensional dengan mencatat data customer atau tamu dalam sebuah buku. Sementara itu, para tamu yang ingin menginap hanya bisa menghubungi pihak pengelola melalui whatsapp ataupun melakukan pemesanan langsung di tempat. Hal ini menyebabkan kesulitan dalam mengelola data reservasi secara efisien, kurangnya informasi yang akurat mengenai ketersediaan kamar penginapan, serta potensi terjadinya kesalahan dalam pencatatan data.

Dalam meningkatkan kualitas pelayanan di Pantai Uma Wanau Private Beach, diperlukan sebuah aplikasi reservasi penginapan online berbasis web. Untuk membangun aplikasi web ini diterapkan sebuah arsitektur Model-View-Controller (MVC) dengan framework Laravel. Model-View-Controller (MVC) adalah salah satu dari beberapa arsitektur model infrastruktur yang diusulkan yang umumnya digunakan dalam pengembangan web. Model menggabungkan semua konten khusus aplikasi dan algoritma pemrosesan. View berisi semua fungsi khusus antarmuka dan memungkinkan pengguna akhir untuk menyajikan konten dan menjalankan logika pemrosesan. Controller menangani akses ke model dan view, serta aliran data di antara keduanya [1].

Beberapa penelitian sebelumnya antara lain, penelitian mengenai perancangan sistem informasi reservasi kamar hotel di Hotel Royal Ambarrukmo Yogyakarta menggunakan Model-View-Controller (MVC) pada aplikasi web [2]. Kemudian penelitian yang membahas tentang pengembangan aplikasi reservasi hotel yang memudahkan pengelola dalam mendata tamu yang ingin melakukan reservasi secara online di Hotel Tambora Sumbawa [3]. Penelitian lainnya yang menerapkan metode Rapid Application Development (RAD) dalam pengembangan sistem reservasi dan penyewaan kamar Hotel Queen [4]. Meskipun ada beberapa penelitian tentang aplikasi reservasi penginapan berbasis web, tetapi belum ada penelitian yang menerapkan arsitektur Model View Controller dan metode pengembangan Rapid Application Development.

Penelitian ini bertujuan untuk merancang dan membangun sebuah aplikasi reservasi penginapan berbasis web dengan menerapkan arsitektur MVC. Dengan adanya aplikasi ini, diharapkan dapat memberikan pengalaman bagi pengunjung dalam melakukan reservasi kamar penginapan, menyediakan informasi yang akurat mengenai ketersediaan kamar penginapan, serta meningkatkan efisiensi pengelolaan data reservasi bagi pihak pengelola Pantai Uma Wanau Private Beach.

2. Landasan Teori

Reservasi

Reservasi dalam konteks penginapan adalah pemesanan kamar oleh tamu atau pelanggan. Jenis kamar yang dipesan disediakan dan dinyatakan dengan benar untuk beberapa orang selama jangka waktu tertentu. Pada saat reservasi, pengelola penginapan diharapkan berjanji untuk menerima reservasi dan memastikan bahwa kamar yang dibeli tersedia saat tamu tiba. Reservasi adalah permintaan tamu untuk mendapatkan kamar yang diinginkan yang dilakukan beberapa waktu sebelumnya melalui berbagai sumber dan metode pemesanan dengan tujuan untuk memastikan bahwa tamu tersebut memiliki kamar ketika mereka tiba atau check-in [5].

Model-View-Controller

Arsitektur Model-View-Controller (MVC) adalah salah satu dari beberapa arsitektur model infrastruktur yang diusulkan yang umumnya digunakan dalam pengembangan web. Model menggabungkan semua konten khusus aplikasi dan algoritma pemrosesan. View berisi semua fungsi khusus antarmuka dan memungkinkan pengguna akhir untuk menyajikan konten dan menjalankan logika pemrosesan. Controller menangani akses ke model dan view, serta aliran data di antara keduanya [6].

Laravel

Laravel adalah framework PHP berbasis web gratis yang dikembangkan oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi web yang menggunakan pola arsitektur model-view-controller (MVC). Laravel memiliki banyak fitur, termasuk pengembangan sistem modul-modul yang dapat diatur, berbagai cara untuk mengakses database relasional, utilitas yang memudahkan penyebaran aplikasi, dan kemudahan pemeliharaan [5].

Database

Database adalah kumpulan data yang berhubungan secara logis dan deskripsinya yang dimaksudkan untuk memenuhi kebutuhan informasi organisasi. Sistem Manajemen Basis Data (DBMS) adalah sistem perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membangun, mengelola, dan mengontrol akses basis data. Perangkat lunak aplikasi (database) adalah program komputer yang berinteraksi dengan database dengan mengirimkan permintaan yang sesuai (biasanya berupa pernyataan SQL) ke DBMS [7].

PHP

PHP adalah bahasa pemrograman serba guna yang populer yang cocok untuk pengembangan web dan dapat diintegrasikan ke dalam HTML. PHP dirancang untuk melakukan tindakan hanya ketika suatu peristiwa terjadi, seperti pengguna mengirimkan formulir atau mengunjungi URL [8].

MySQL

MySQL adalah sistem manajemen basis data yang bersifat open source. MySQL adalah sistem manajemen basis data relasional. Artinya, data dalam database akan dibagi menjadi beberapa tabel independen, sehingga memungkinkan penanganan data yang jauh lebih cepat. MySQL memungkinkan pengguna untuk merepresentasikan hubungan yang kompleks antara item data dan menghitung korelasi ini dengan kecepatan yang dibutuhkan untuk pengambilan keputusan di perusahaan modern. Metodologi pengembangan MySQL difokuskan untuk menyediakan implementasi yang sangat efisien dari fungsi-fungsi yang dibutuhkan oleh sebagian besar orang [9].

Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah "bahasa standar untuk membuat cetak biru perangkat lunak." UML dapat digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem perangkat lunak yang intensif. Dengan kata lain, seperti halnya arsitek bangunan yang membuat cetak biru untuk perusahaan konstruksi, arsitek perangkat lunak merancang diagram UML untuk membantu insinyur perangkat lunak dalam mengembangkan perangkat lunak. Memahami kosakata diagram UML, termasuk bagian bergambar dan artinya, memungkinkan spesifikasi dan penjelasan sistem yang lebih mudah kepada orang lain [6].

Pengujian White Box

Pengujian White Box adalah teknik pengujian perangkat lunak. Pengujian white box berfokus pada struktur aplikasi. Metode ini melihat ke dalam logika internal, struktur kode, dan aliran kontrol aplikasi. Metode ini disebut juga sebagai pengujian clear box, pengujian open box, pengujian glass box, pengujian transparent box, pengujian code-based, atau pengujian structural [10].

Metode Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah strategi pengembangan sistem sekuensial linier yang menekankan pada siklus pengembangan sistem yang singkat untuk menghemat waktu dan mempercepat proses pengembangan sistem. Proses pengembangan sistem menjadi lebih cepat. Penerapan pendekatan Rapid Application Development (RAD) dalam perancangan perangkat lunak dapat meningkatkan efisiensi pengembangan dan pemeliharaan sistem [11].

3. Metode Penelitian

Waktu dan Tempat Penelitian

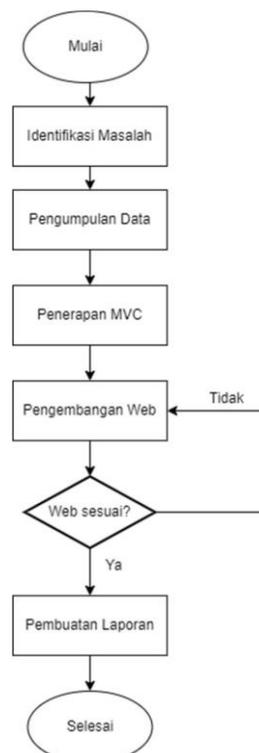
Penelitian ini dimulai pada bulan Maret 2024 hingga September 2024 dan akan dilaksanakan di Pantai Uma Wanau Private Beach yang beralamat di Jalan Trans Kema-Rumbia, Desa Makalisung, Kecamatan Kombi, Kabupaten Minahasa.

Teknik Pengumpulan Data

- a. Studi Literatur
Studi literatur dilakukan dengan mempelajari beberapa jurnal dan buku-buku yang berkaitan dengan reservasi penginapan, arsitektur Model-View-Controller dan penggunaan framework laravel
- b. Observasi
Observasi dilakukan dengan pengamatan langsung di obyek penelitian dan mengambil dokumentasi berupa foto di penginapan Pantai Uma Wanau Private Beach.
- c. Wawancara
- d. Studi pustaka dilakukan di lokasi penelitian dengan pihak pengelola pantai Uma Wanau Private Beach, dengan melakukan komunikasi berupa tanya-jawab berdasarkan pertanyaan yang telah disiapkan. Wawancara ini bertujuan untuk mengumpulkan informasi mengenai proses bisnis yang terjadi di pantai Uma Wanau Private Beach.

Tahapan Penelitian

Pada tahapan penelitian ini menjelaskan serangkaian langkah yang dilakukan oleh penulis dengan menggambarannya melalui flowchart. Tahapan-tahapan penelitian dapat dilihat pada gambar 1.



Gambar 1. Tahapan Penelitian

Metode Pengembangan

Metode pengembangan perangkat lunak yang digunakan adalah metode Rapid Application Development (RAD). Tahapan metode RAD tersebut dijelaskan sebagai berikut:

- a. **Perencanaan Kebutuhan** Pada tahap ini, dilakukan pengumpulan informasi tentang proses bisnis untuk mengidentifikasi kebutuhan dari aplikasi reservasi penginapan online. Informasi diperoleh melalui wawancara dengan pihak pengelola maupun observasi langsung di Pantai Uma Wanau Private Beach.
- b. **Desain Sistem**
Pada tahap ini dilakukan perancangan sistem berdasarkan hasil perencanaan kebutuhan pada tahap sebelumnya. Tahap ini melibatkan proses desain antarmuka dengan pengguna yaitu pengelola Pantai Uma Wanau Private Beach. Selain desain antarmuka, desain sistem ini juga akan menggunakan UML (Unified Modelling Language) yaitu Use Case Diagram, Activity Diagram, Class Diagram dan Entity Relationship Diagram (ERD) untuk desain basis datanya.
- c. **Pengembangan**
Setelah tahap desain sistem, aplikasi reservasi penginapan dibangun sesuai dengan rancangan desain sistem yang telah disempurnakan. Pengembangan aplikasi dilakukan dengan menggunakan bahasa pemrograman dengan framework Laravel dan database MySQL. Arsitektur MVC diterapkan untuk memisahkan komponen Model (data dan logika bisnis), View (antarmuka pengguna), dan Controller (penghubung antara Model dan View).
- d. **Implementasi**
Melakukan pengujian untuk memastikan aplikasi berjalan dengan baik dan sudah memenuhi kebutuhan sesuai dengan yang diharapkan, kemudian mengidentifikasi dan memperbaiki jika terdapat kesalahan dalam pengujian aplikasi.

4. Hasil Penelitian

Perencanaan Kebutuhan

Penelitian dilakukan di lokasi Pantai Uma Wanau, Kecamatan Kombi, Kabupaten Minahasa. Data penelitian diperoleh dengan melakukan wawancara dengan bapak Riflex Manderos selaku pihak pengelola Pantai Uma Wanau. Wawancara ini bertujuan untuk mengetahui proses bisnis yang meliputi reservasi, informasi terkait penginapan dan fasilitas, dan kebutuhan dari pengelola usaha.

Kebutuhan sistem dalam pembuatan aplikasi reservasi penginapan berbasis web antara lain sebagai berikut:

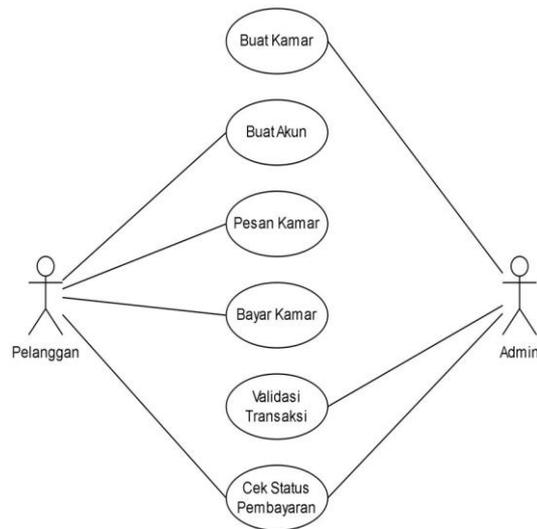
1. **Kebutuhan Fungsional**
 - a. Registrasi dan login pengguna dan admin.
 - b. Pemesanan kamar oleh pengguna.
 - c. Pencarian dan filter kamar.
 - d. Konfirmasi pemesanan oleh admin.
 - e. Laporan data reservasi oleh admin.
 - f. Dashboard admin untuk mengelola data pengguna, data kamar, dan reservasi.
2. **Kebutuhan Non-Fungsional:**
 - a. Aplikasi web yang responsif.
 - b. Desain antarmuka pengguna yang mudah digunakan.

Desain Sistem

Desain atau perancangan sistem ini menggunakan Unified Modelling Language (UML) yang meliputi use case diagram, activity diagram, entity relationship diagram (ERD) dan class diagram.

a. Use Case Diagram

Pada use case diagram ini dimana terdapat admin dan pelanggan sebagai aktor, admin yang memiliki tiga use case yaitu membuat kamar, memvalidasi transaksi, dan cek status pembayaran. Sedangkan aktor pelanggan dapat membuat akun, memesan kamar, bayar kamar, cek status pembayaran dan logout dari sistem. Use case diagram pengguna dapat dilihat pada Gambar 2.

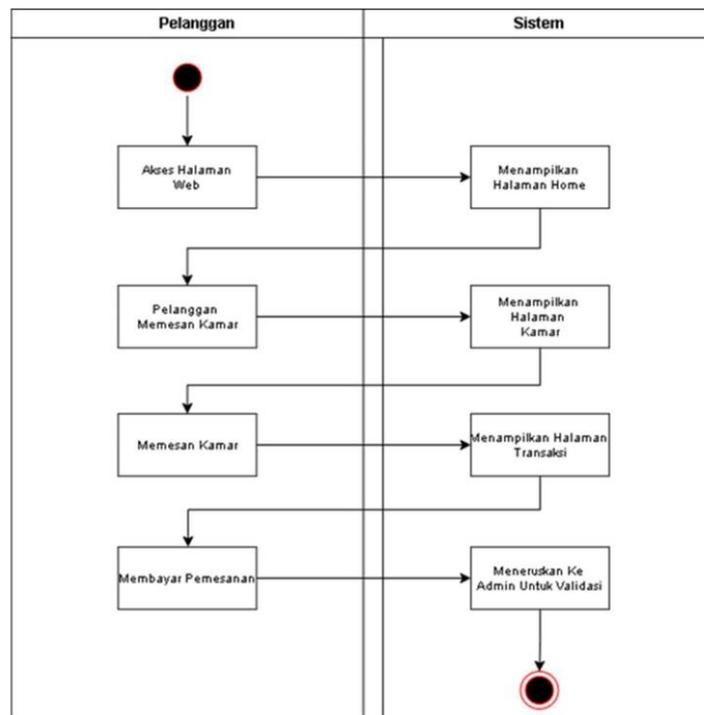


Gambar 2. Use Case Diagram Sistem Reservasi Penginapan

b. Activity Diagram Pada system ini terdapat dua activity diagram yaitu pelanggan dan admin. Berikut adalah activity diagram dari sistem yang dibangun dapat dilihat pada Gambar 3 dan Gambar 4.

1. Activity Diagram Pelanggan

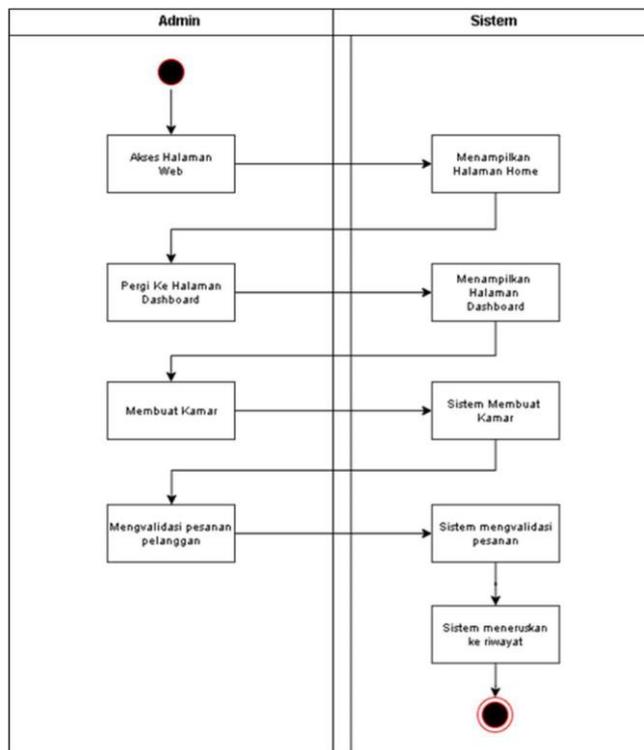
Pada activity diagram pelanggan berfokus pada pemesanan kamar dan melakukan pembayaran, dengan interaksi utama antara pelanggan dan halaman web sistem.



Gambar 3. Activity Diagram Pelanggan

2. Activity Diagram Admin

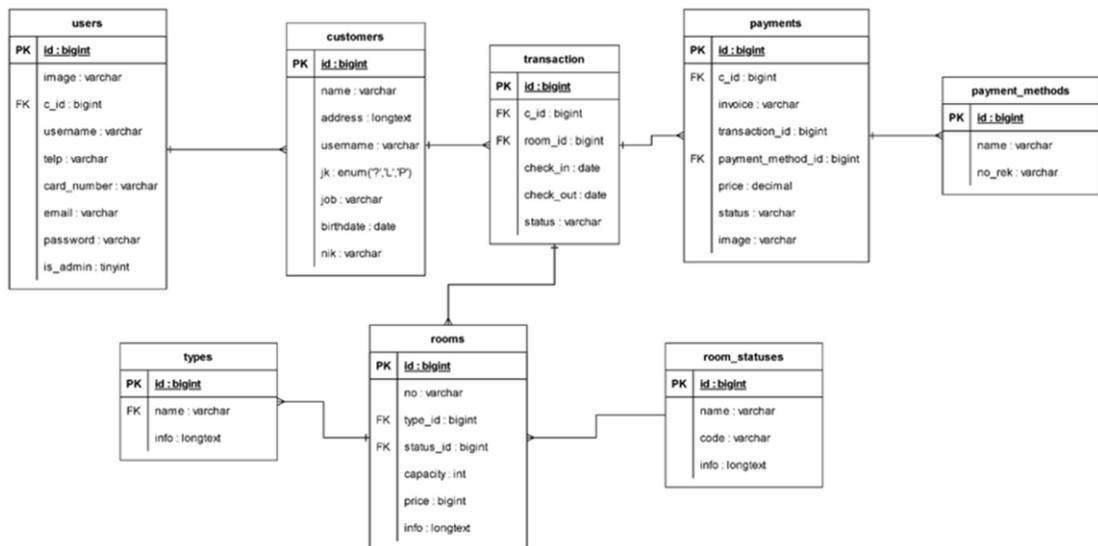
Activity diagram admin berfokus pada pengelolaan kamar dan validasi pesanan pelanggan oleh admin. Admin dapat mengakses dashboard, mengubah data kamar, dan melakukan validasi pembayaran dapat dilihat pada Gambar 5.



Gambar 4. Activity Diagram Admin

3. Entity Relationship Diagram

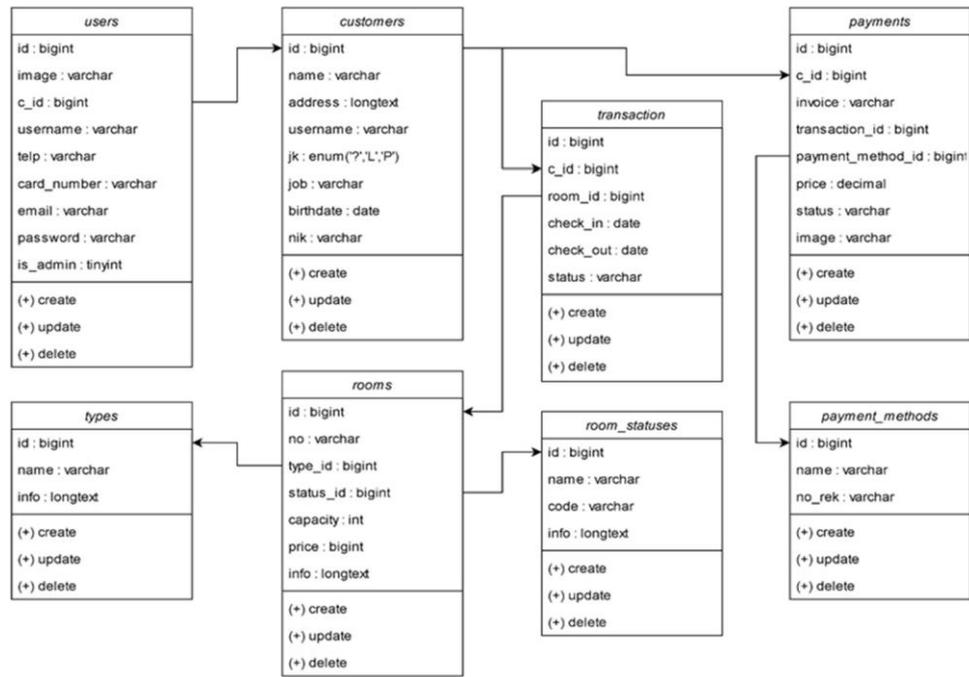
Entity Relationship Diagram (ERD) menggambarkan rancangan basis data yang terdapat entitas-entitas dengan atributnya serta relasi antar entitas tersebut. ERD pada sistem yang akan dibangun terdapat entitas yaitu user, customer, transaction, payment, payment method, types, rooms, room status. ERD dari Sistem Reservasi Penginapan dilihat pada Gambar 5.



Gambar 5. Entity Relationship Diagram Sistem Reservasi

4. Class Diagram

Class diagram menggambarkan tentang kelas-kelas atau objek, dimana pada class diagram tersebut terdapat atribut serta metode atau operasi dari sistem yang dirancang. Class tersebut saling terhubung satu sama lain. Pada class diagram ini terdapat delapan class yaitu users, customers, transactions, payments, types, rooms, room status dan payment method. Class diagram pengembangan sistem dapat dilihat pada Gambar 6.

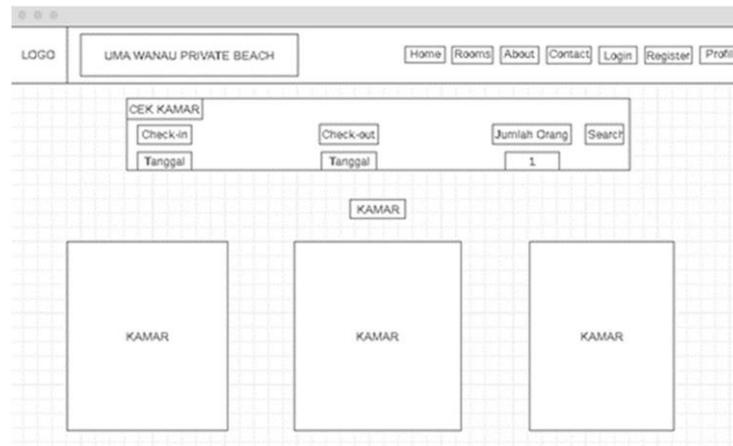


Gambar 6. Class Diagram Sistem Reservasi

Prototype Low-Fidelity Web Reservasi

a. Halaman Utama

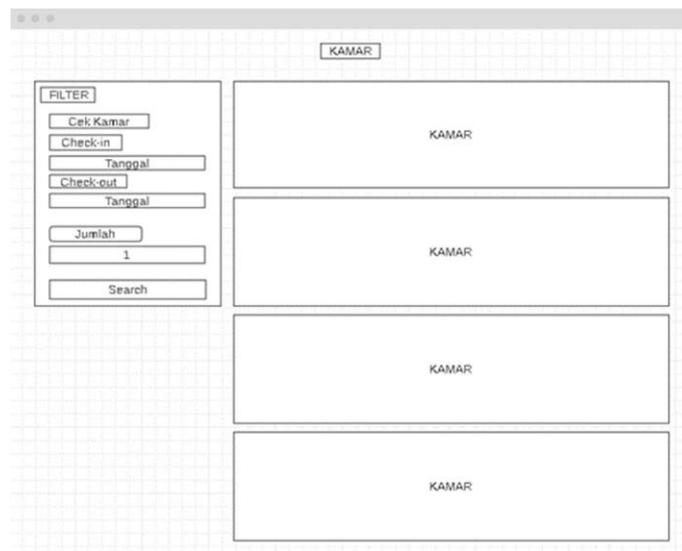
Halaman Utama berfungsi sebagai titik awal bagi pengguna untuk memulai pencarian kamar dengan cepat. Formulir pencarian berada di pusat perhatian, sementara beberapa kamar ditampilkan di bawahnya. Wireframe Halaman Utama dapat dilihat pada Gambar 7.



Gambar 7. Prototype Low-Fidelity Halaman Utama

b. Halaman Kamar

Halaman Kamar berfokus pada hasil pencarian yang lebih rinci, memungkinkan pengguna untuk mempersempit hasil dengan filter dan melihat lebih banyak kamar dalam tata letak vertikal yang memudahkan penjelajahan. Wireframe Halaman Kamar dapat dilihat pada Gambar 8.

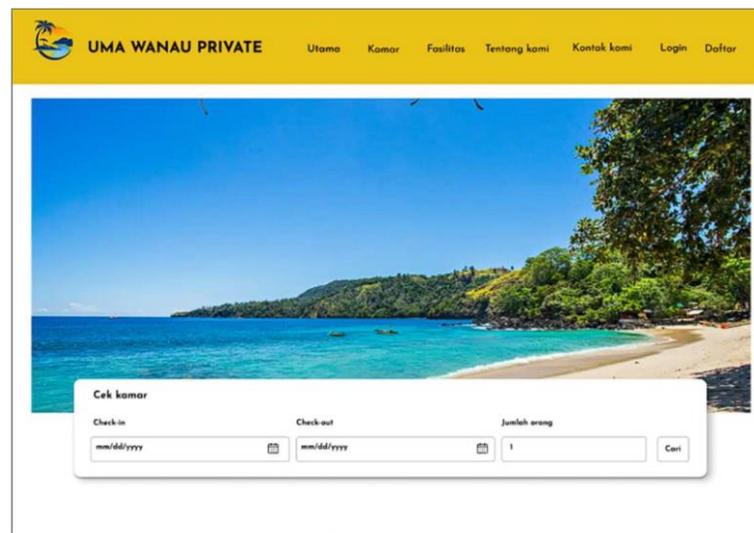


Gambar 8. Prototype Low-Fidelity Halaman Kamar

Prototype High-Fidelity Web Reservasi

a. Halaman Utama

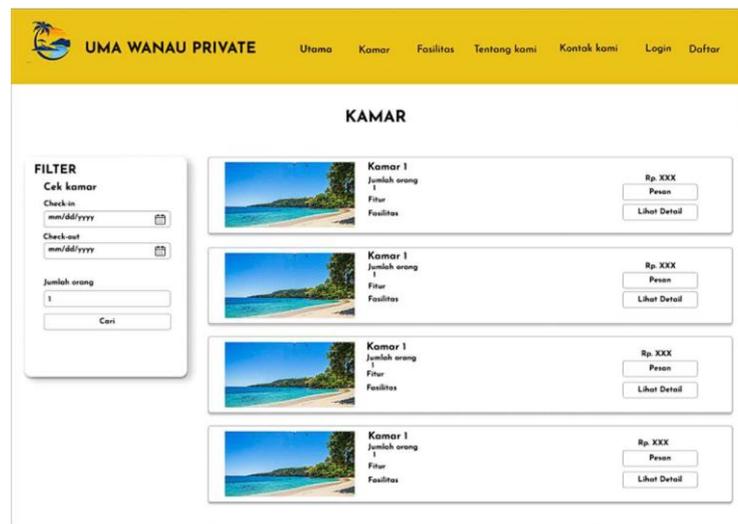
Halaman utama menampilkan gambar pantai dengan form pencarian kamar di bagian bawah. Pengguna dapat memasukkan tanggal check-in, check-out, dan jumlah tamu untuk mencari kamar. Navigasi di bagian atas memudahkan akses ke berbagai fitur seperti kamar, fasilitas, dan kontak. Desain sederhana dan fokus pada kemudahan reservasi. Tampilan High-Fidelity Halaman Utama dapat dilihat pada Gambar 9.



Gambar 9. Prototype High-Fidelity Halaman Utama

b. Halaman Kamar

Halaman kamar menampilkan daftar kamar yang tersedia dengan filter di sebelah kiri untuk memilih tanggal check-in, check-out, dan jumlah tamu. Setiap kamar ditampilkan dalam kartu yang berisi gambar, jumlah orang yang bisa menginap, fitur, fasilitas, dan harga. Ada dua tombol di setiap kartu: Pesan untuk langsung memesan dan Lihat Detail untuk informasi lebih lanjut. Layout yang bersih, memudahkan pengguna untuk melihat dan memesan kamar. Tampilan High-Fidelity Halaman Kamar dapat dilihat pada Gambar 10.



Gambar 10. Prototype High-Fidelity Halaman Kamar

Pengembangan

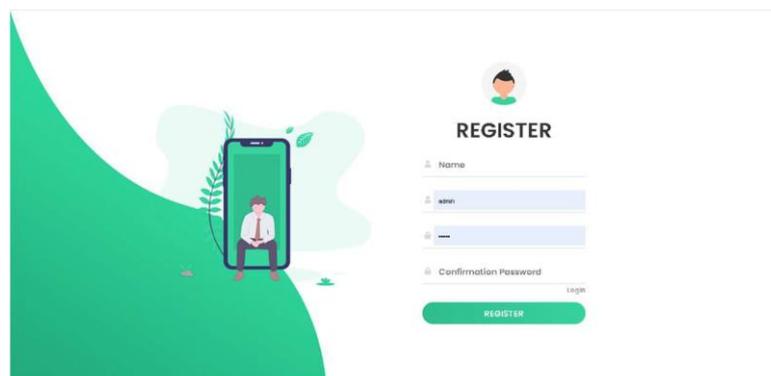
Pengembangan dilakukan dengan menggunakan framework Laravel, yang mendukung struktur MVC secara terintegrasi. Pembangunan sistem diawali dengan pembuatan migrasi untuk tabel-tabel yang dibutuhkan, seperti tabel users, customers, transaction, payments, types, rooms, room_status dan payment_methods menggunakan fitur migrasi Laravel. Selanjutnya, komponen model dikembangkan untuk menghubungkan aplikasi dengan database. Pengembangan view dilakukan menggunakan Blade, yang memungkinkan pembuatan template dinamis dengan sintaks yang sederhana. Penggunaan komponen Bootstrap membantu dalam membuat tampilan yang responsif untuk berbagai perangkat. Komponen controller dikembangkan untuk mengelola logika aplikasi seperti pengecekan ketersediaan kamar, validasi data input pengguna, dan pengelolaan transaksi.

Implementasi MVC

Implementasi kode program dibuat menggunakan bahasa pemrograman PHP dengan framework Laravel. Berikut adalah alur Model-View-Controller untuk proses pemesanan kamar:

a. Halaman Register

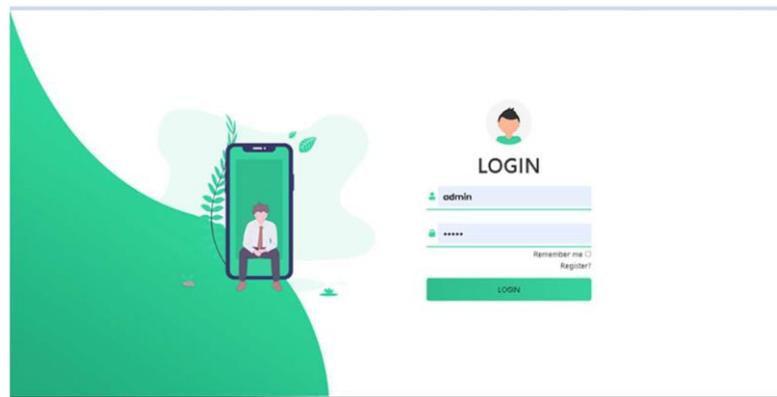
Halaman register memungkinkan pengguna baru untuk membuat akun. Pengguna yang belum memiliki akun yang teregistrasi harus melakukan registrasi dengan nama, username dan password yang baru agar dapat melakukan reservasi. Halaman register dapat dilihat pada Gambar 11.



Gambar 11. Halaman Register

b. Halaman Login

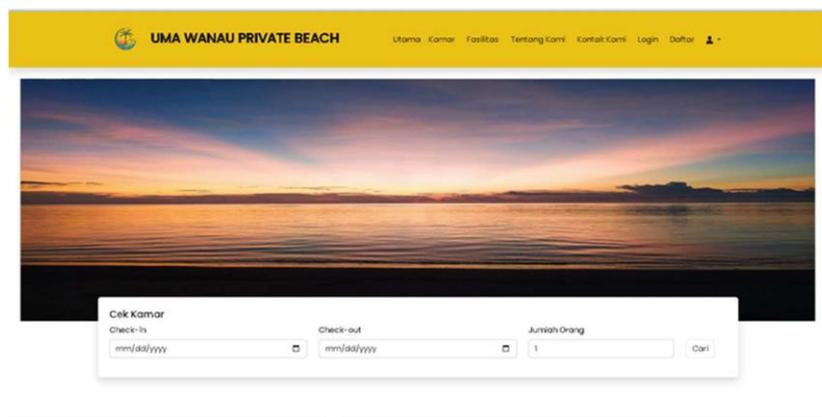
Halaman Halaman Login memungkinkan pengguna yang sudah terdaftar untuk mengakses akun mereka. Pengguna diminta memasukkan username dan kata sandi yang telah terdaftar. Halaman Login dapat dilihat pada Gambar 12.



Gambar 12. Halaman Login

c. Halaman Utama

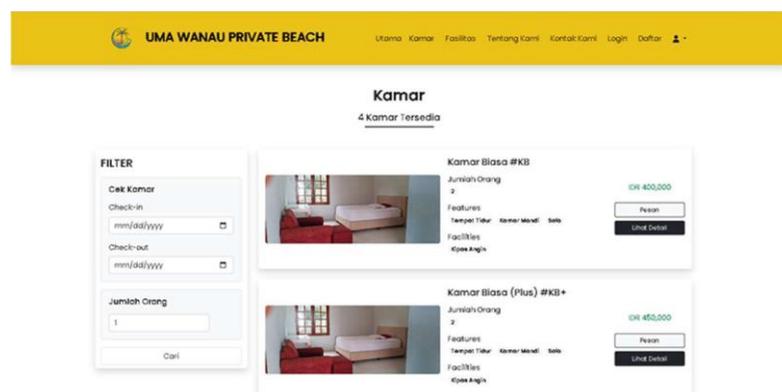
Halaman utama merupakan tampilan awal dari web reservasi. Di bagian atas terdapat menu navigasi, tombol Utama, Kamar, Fasilitas, Tentang Kami, Kontak Kami, Login, Daftar dan tombol Profil, dan fitur Cek Kamar untuk mencari kamar yang tersedia pada tanggal yang dipilih. Halaman utama dapat dilihat pada Gambar 13.



Gambar 13. Halaman Utama

d. Halaman Kamar

Halaman Kamar menampilkan daftar kamar yang tersedia beserta deskripsinya, seperti jenis kamar, fasilitas, harga per malam, serta kapasitas kamar. Pengguna dapat melihat detail lebih lanjut tentang setiap kamar dan opsi untuk memesan. Halaman Kamar dapat dilihat pada Gambar 14.



Gambar 14. Halaman Kamar

e. Halaman Detail Pesanan Kamar

Halaman Detail Pesanan Kamar menampilkan detail pesanan dan mengharuskan pemesan untuk segera mengirim bukti pembayaran berupa screenshot. Halaman Detail Pesanan Kamar dapat dilihat pada Gambar 15.

Detail Pesanan #05INV3bxxp

Room: KB+

Type: Kamar Biosa (Plus)

Capacity: 2

Price / Day: IDR 400,000

Check in: 3 November 2024

Check Out: 4 November 2024

Total Day: 1 Day

Total Price: IDR 400,000

Payment Method: BCA

Kirim Bukti Pembayaran

Transfer ke: 5221-8420-7788-2024

Upload Bukti (Maks 2MB)

Choose File No file chosen

KIRIM

Gambar 15. Halaman Detail Pesanan Kamar

f. Halaman Invoice

Halaman Invoice menampilkan rincian pembayaran setelah pengguna berhasil melakukan pemesanan. Halaman ini mencakup informasi penting seperti nomor invoice, tanggal pemesanan, rincian kamar yang dipesan, durasi menginap dan total biaya. Halaman Invoice dapat dilihat pada Gambar 16.

INVOICE Down Payment
#05INV2xkbV

INVOICE #05INV2xkbV

Booking ROOM KB
CLIENT 8612
ADDRESS
EMAIL
DATE 17 Sep 2024
DUAR DATE 18 Sep 2024

Caltra Gurdan
Srik Bina Mandiri Multimedik,
Cileungsi, Jawa Barat
+62 85773674231
Anggergurdan@gmail.com

ROOM	CHECK IN	CHECK OUT	DAY	PRICE	TOTAL
Kamar Biosa # KB	17 Sep 2024	18 Sep 2024	1 Days	400,000	400,000
				SUBTOTAL	Rp. 400,000
				GRAND TOTAL	Rp. 400,000

Gambar 16. Halaman Invoice Pengguna

g. Halaman Dashboard Admin

Halaman Dashboard Admin memberikan pandangan menyeluruh mengenai ketersediaan kamar, status pemesanan, dan laporan pendapatan, serta memungkinkan pengelolaan kamar dan penyesuaian tarif. Halaman Dashboard Admin dapat dilihat pada Gambar 17.

UMA WANAU PRIVATE BEACH

Dashboard

BOOKING OPTIMIZED MONTHLY
IDR 800,000

BOOKING ALL TIME
IDR 800,000

TOTAL TRANSACTIONS
4

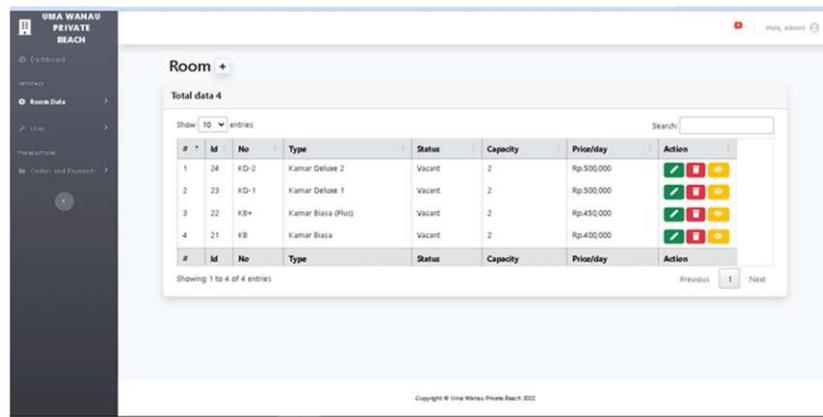
Generate Report

Copyright © Uma Wanau Private Beach 2024

Gambar 17. Halaman Dashboard Admin

h. Halaman Data Kamar

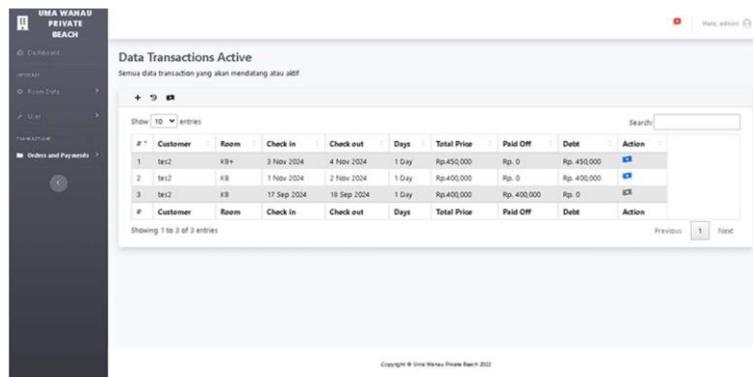
Halaman Data Kamar menyajikan informasi rinci mengenai setiap kamar, termasuk tipe, fasilitas, tarif, dan status ketersediaan, serta memungkinkan admin untuk memperbarui data kamar, menambah kamar ataupun menghapus kamar. Halaman Data Kamar dapat dilihat pada Gambar 18.



Gambar 18. Halaman Data Kamar

i. Halaman Transaksi

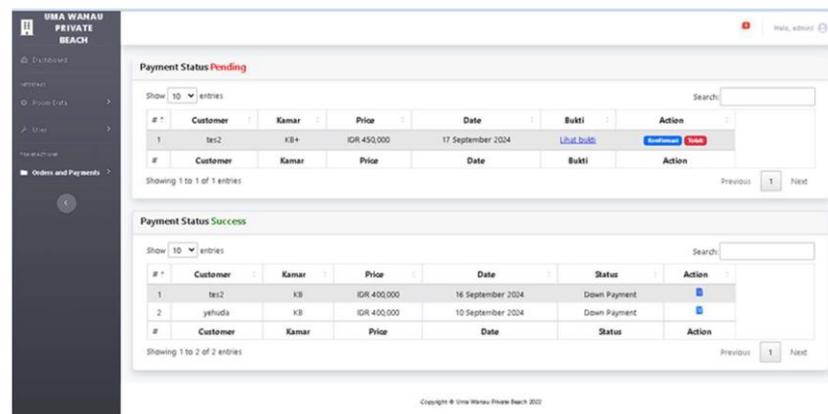
Halaman Transaksi menampilkan ringkasan dari semua transaksi pemesanan yang berhasil. Halaman Transaksi dapat dilihat pada Gambar 19.



Gambar 19. Halaman Transaksi

j. Halaman Status Pembayaran

Halaman Status Pembayaran memberikan gambaran menyeluruh tentang status setiap pembayaran, termasuk konfirmasi, pending, serta memungkinkan admin untuk melacak dan mengelola transaksi pembayaran. Halaman Status Pembayaran dapat dilihat pada Gambar 20.



Gambar 20. Halaman Transaksi

Pengujian Black Box

Pengujian ini berfokus pada fungsionalitas web tanpa memeriksa struktur kode. Terdapat beberapa skenario pengujian untuk memastikan web bekerja seperti yang diharapkan. Berikut ini adalah table hasil pengujian beberapa kasus dapat dilihat pada tabel pada tabel 1.

Tabel 1. Hasil Pengujian Black Box

No.	Skenario Pengujian	Proses yang diuji	Hasil yang diharapkan	Hasil Pengujian
1.	Semua <i>field</i> kosong	Menguji apakah registrasi gagal ketika semua <i>field</i> kosong.	Sistem menampilkan error bahwa <i>field</i> kosong	Berhasil
2.	<i>Username</i> terlalu pendek	Menguji apakah registrasi gagal ketika <i>username</i> kurang dari 3 karakter.	Sistem menampilkan error bahwa <i>username</i> terlalu pendek.	Berhasil
3.	<i>Username</i> sudah ada	Pengecekan keunikan <i>username</i> di <i>database</i> .	Sistem menampilkan error bahwa <i>username</i> sudah digunakan.	Berhasil
4.	<i>Password</i> terlalu pendek	Menguji apakah registrasi gagal ketika <i>username</i> yang sama.	Sistem menampilkan error bahwa <i>password</i> terlalu pendek (minimal 3 karakter).	Berhasil
5.	Konfirmasi <i>password</i> tidak sesuai	Menguji apakah registrasi gagal ketika <i>password</i> dan konfirmasi <i>password</i> tidak cocok.	Sistem menampilkan error bahwa konfirmasi <i>password</i> tidak sesuai.	Berhasil
6.	Semua input valid	Menguji apakah pengguna dapat melakukan registrasi ketika semua input valid.	Data disimpan di tabel <i>customer</i> dan pengguna kemudian diarahkan ke halaman <i>login</i> .	Berhasil
7.	Pemesanan kamar berhasil.	Pengguna melakukan pemesanan kamar dengan data yang valid.	Pemesanan berhasil diproses	Berhasil
8.	Pemesanan gagal (kamar sudah dipesan)	Pengguna mencoba memesan kamar yang sudah dipesan oleh pengguna lain pada tanggal yang sama.	Sistem menolak pemesanan	Berhasil
9.	Pemesanan kamar berhasil	Pengguna berhasil memesan kamar.	Pemesanan tersebut berhasil tersimpan di riwayat	Berhasil

			pemesanan.	
10.	Notifikasi admin berhasil.	Pengguna berhasil memesan kamar dengan data valid.	Admin menerima notifikasi pemesanan dan tersimpan di tabel pesanan.	Berhasil
11.	Notifikasi admin pembayaran kamar	Pengguna berhasil memesan kamar tapi belum mengunggah bukti pembayaran.	Pesanan tersebut tersimpan di tabel pesanan dan hanya muncul notifikasi pending pembayaran.	Berhasil
12.	Unggah bukti pembayaran berhasil	Pengguna mengunggah bukti pembayaran.	Admin dapat melihat bukti pembayaran tersebut.	Berhasil

Pengujian White Box

Pengujian ini untuk memastikan bahwa setiap komponen dalam aplikasi berjalan dengan baik. Hasil pengujian white box untuk dapat dilihat pada tabel 2.

Tabel 2. Hasil Pengujian White Box

No.	Skenario Pengujian	Komponen MVC dan Kasus Uji	Hasil yang diharapkan	Hasil Pengujian
1.	Menguji proses registrasi dengan data pengguna yang valid.	<p><i>Model:</i> Memastikan data pengguna baru tersimpan dengan benar di tabel <i>users</i> dan <i>customers</i> ketika input valid.</p> <p><i>View:</i> Memastikan pengguna diarahkan <i>redirect</i> ke halaman <i>login</i> setelah registrasi sukses.</p> <p><i>Controller:</i> Memastikan alur registrasi berjalan tanpa error dan menyimpan data jika input valid.</p>	<p><i>Model:</i> - Data tersimpan di tabel <i>users</i> dengan <i>username</i> "testuser". - Data tersimpan di tabel <i>customers</i> dengan nama "testuser".</p> <p><i>View:</i> Aplikasi melakukan <i>redirect</i> ke <i>/login</i> setelah data pengguna tersimpan dengan sukses.</p> <p><i>Controller:</i> - Data pengguna baru disimpan tanpa masalah jika semua input valid. - Pengguna diarahkan ke halaman <i>login</i> setelah registrasi.</p>	Berhasil
2.	Menguji apakah sistem gagal validasi ketika seluruh <i>form</i> tidak	<p><i>View:</i> Memastikan pengguna diarahkan ke halaman <i>register</i></p>	<p><i>View:</i> Sistem melakukan <i>redirect</i> ke halaman <i>/register</i> karena semua <i>field</i> kosong, yang</p>	Berhasil

	diisi	jika semua <i>field</i> registrasi kosong.	melanggar aturan validasi.	
		<i>Controller</i> : Memastikan alur registrasi berhenti dan tidak menyimpan data jika semua <i>field</i> registrasi kosong.	<i>Controller</i> : - Registrasi gagal jika semua <i>field</i> kosong. - Pengguna diarahkan ke <i>/register</i> untuk mengisi ulang <i>form</i> .	
3.	Menguji apakah sistem gagal validasi ketika <i>username</i> kurang dari 3 karakter.	<i>View</i> : Memastikan pengguna diarahkan kembali ke halaman <i>register</i> jika <i>username</i> kurang dari 3 karakter. <i>Controller</i> : Memastikan alur registrasi berhenti jika panjang <i>username</i> kurang dari 3 karakter, dan memberi pesan error.	<i>View</i> : - Sistem melakukan redirect ke <i>/register</i> karena <i>username</i> kurang dari 3 karakter. - Terdapat pesan error terkait panjang <i>username</i> . <i>Controller</i> : - Registrasi gagal jika panjang <i>username</i> tidak memenuhi syarat. - Pengguna diarahkan ke <i>/register</i> dengan pesan error.	Berhasil
4.	Menguji apakah sistem gagal validasi ketika <i>username</i> sudah ada.	<i>Model</i> : Memastikan tidak ada data pengguna baru yang disimpan jika <i>username</i> sudah ada di database. <i>View</i> : Memastikan pengguna diarahkan kembali ke halaman register jika <i>username</i> sudah ada di database. <i>Controller</i> : Memastikan alur registrasi berhenti jika <i>username</i> sudah ada di database, dan memberi notifikasi error.	<i>Model</i> : - Data tidak disimpan karena <i>username</i> duplikat. - Hanya ada satu pengguna dengan <i>username</i> "testuser" di tabel users. <i>View</i> : - Pengguna diarahkan kembali ke halaman <i>/register</i> . - Terdapat pesan error pada sesi terkait <i>username</i> yang sudah ada. <i>Controller</i> : - Registrasi gagal jika <i>username</i> sudah ada. - Pengguna diarahkan ke <i>/register</i> dan diberi pesan error <i>username</i> sudah ada.	Berhasil
5.	Menguji apakah sistem	<i>View</i> : Memastikan	<i>View</i> : - Sistem melakukan redirect ke	Berhasil

	gagal validasi ketika password kurang dari 3 karakter.	pengguna diarahkan kembali ke halaman register jika password kurang dari 3 karakter. <i>Controller:</i> Memastikan alur registrasi berhenti jika panjang password kurang dari 3 karakter, dan memberi pesan error.	/register karena password kurang dari 3 karakter. - Terdapat pesan error terkait panjang password. <i>Controller:</i> - Registrasi gagal jika panjang password tidak memenuhi syarat. - Pengguna diarahkan ke /register dengan pesan error.	
6.	Menguji apakah sistem gagal validasi ketika konfirmasi password tidak sama dengan password	<i>View:</i> Memastikan pengguna di redirect ke halaman register jika password dan konfirmasi password tidak cocok. <i>Controller:</i> Memastikan alur registrasi berhenti dan tidak menyimpan data jika password dan konfirmasi password tidak cocok.	<i>Model:</i> - Sistem melakukan redirect ke halaman /register jika password dan confirmation_password tidak cocok. <i>Controller:</i> - Registrasi gagal dan tidak menyimpan data jika validasi password tidak cocok. - Pengguna diarahkan ke /register untuk memperbaiki input.	Berhasil

5. Kesimpulan

Penelitian ini telah berhasil merancang dan membangun sebuah aplikasi reservasi penginapan berbasis web untuk penginapan pantai Uma Wanau Private Beach dengan menerapkan arsitektur Model-View-Controller (MVC). Penerapan arsitektur MVC ini memungkinkan pemisahan antara logika bisnis, antarmuka pengguna, dan manipulasi data. Dengan demikian, pengembangan aplikasi menjadi lebih terstruktur, mudah dipelihara, serta memungkinkan pengembangan lebih lanjut di masa depan. Aplikasi ini menyediakan fitur-fitur penting yang mendukung proses reservasi, seperti pencarian kamar, pemesanan, dan manajemen data pengguna. Aplikasi yang dikembangkan dapat membantu pengelolaan data reservasi bagi pihak pengelola Pantai Uma Wanau Private Beach. Aplikasi ini menyediakan sistem manajemen data reservasi yang terintegrasi, sehingga pengelola dapat dengan mudah memantau status pemesanan kamar, ketersediaan kamar, serta riwayat transaksi.

6. Daftar Pustaka

- [1] Pressman R, Maxim B. (2020). Software Engineering. McGraw-Hill Education, 2 Penn Plaza, New York
- [2] Awaludin, F., Liana Sari, W., Alfaridzi Hidayat, G., Rizky Darmawan, M., & Saifudin, A. (2023). Analisa Dan Perancangan Sistem Informasi Reservasi Kamar Hotel Menggunakan Model View Controller (MVC) Pada Aplikasi Web. Jurnal Manajemen, Ekonomi, Hukum, Kewirausahaan, Kesehatan, Pendidikan dan Informatika (MANEKIN), 1(4), 207–218. <https://journal.mediapublikasi.id/index.php/manekin>
- [3] Nawassyarif, Santika, Y., & Dery Sofya, N. (2022). Rancang Bangun Reservation Hotel Berbasis Web (Studi Kasus Hotel Tambora Sumbawa). JINTEKS (Jurnal Informatika Teknologi Dan Sains), 4(2), 87–93. <https://doi.org/10.51401> [4] Lahama, C., Montolalu, C., Tenda, E. (2024). Sistem Informasi Apotek Berbasis Web Menggunakan Metode Personal Extreme Programming (XP Programming) Dengan Arsitektur MVC

- [4] Fadli, S. (2018). Model Rapid Application Development dalam Pengembangan Sistem Reservasi dan Penyewaan Kamar Hotel. *JIRE (Jurnal Informatika & Rekayasa Elektronika)*, 1(1), 57–64
- [5] Baker S, Bradley P, Huyton J. (2003). *Principles of hotel front office operations*. Thomson Learning, London
- [6] Pressman R, Maxim B. (2020). *Software Engineering*. McGraw-Hill Education, 2 Penn Plaza, New York
- [7] Connolly T, Begg C. (2006). *A Constructivist-Based Approach to Teaching Database Analysis and Design*. Paisley, PA1 2BE, Scotland.
- [8] Ullman L. (2011). *PHP for the Web*. Peachpit Press. Berkeley [10] Krasner G, Pope S. (1988). *A Cookbook for Using the Model-View-Controller User Interface Paradigm in Smalltalk-80*. ParcPlace Systems
- [9] Tahaghogi S, Williams H. (2007). *Learning MySQL: Get a Handle on Your Data*. O'Reilly Media, Inc. Sebastopol
- [10] Verma, A., Khatana, A., Chaudhary, S. (2017). *A Comparative Study of Black Box Testing and White Box Testing*
- [11] McConnell S. (1996). *Rapid Development: Taming Wild Software Schedules*. Microsoft Press. Redmond, Washington