

**RANCANG BANGUN SISTEM INFORMASI KOS-
KOSAN BERBASIS *WEBSITE* DI KECAMATAN
TAMPAN (*ANALYSIS*)**

SKRIPSI



Disusun Oleh:

Muhammad Sukri (1755201038)

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI
BANGKINANG
2021**

**RANCANG BANGUN SISTEM INFORMASI KOS-
KOSAN BERBASIS *WEBSITE* DI KECAMATAN
TAMPAN (*ANALYSIS*)**

SKRIPSI



Disusun Oleh:

Muhammad Sukri (1755201038)

**Diajukan Sebagai Persyaratan untuk Memperoleh Gelar Sarjana
S1 Teknik Informatika**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI
BANGKINANG
2021**

BAB I

PENDAHULUAN

A. Latar Belakang

Rumah kos merupakan salah satu tempat penyedia jasa penginapan atau tempat tinggal sementara yang terdiri dari beberapa kamar setiap kamar memiliki beberapa fasilitas yang ditawarkan atau disediakan dan juga memiliki harga yang telah ditentukan oleh pemilik kos. Pada tahun 2019, sebanyak 118.292 mahasiswa melanjutkan pendidikan di Kecamatan Tampan (Sumber, riau.bps.go.id, 2019). Baik yang berasal dari dalam daerah Kecamatan Tampan sendiri maupun luar daerah Kecamatan Tampan. Tentunya membutuhkan tempat tinggal, baik itu menumpang di rumah saudara maupun tinggal di kos.

Para mahasiswa yang akan mencari rumah kos sebagai tempat tinggalnya tentunya membutuhkan informasi yang benar, akurat, serta lengkap mengenai rumah kos ini, mahasiswa biasanya menggunakan cara manual, antara lain: bertanya kepada masyarakat setempat yang dekat dengan kampus, membaca iklan yang ditempel di tiang, mencari informasi seputar kos melalui media sosial seperti *Facebook* dan lain-lain.

Para mahasiswa membutuhkan waktu yang lama dan biaya lebih untuk mendapatkan informasi mengenai rumah kos yang sesuai dengan keinginan. Mengingat mereka harus mendatangi satu persatu rumah kos

yang mereka temui, oleh sebab itu akan memakan waktu yang lama serta biaya yang banyak.

Adapun permasalahan yang dihadapi oleh pemilik jasa rumah kos adalah kesulitan dalam memasarkan jasa rumah kos serta memberikan informasi kepada calon penghuni rumah kos, apakah ada kamar yang kosong atau tidak. Pemilik rumah kos juga kesulitan mengelola rumah kos apabila rumah kosnya sudah banyak. Kemudian pemilik rumah kos tidak mengingat berapa lama waktu penyewa menyewa kamar kosnya.

Sistem informasi mempunyai dampak yang cukup besar terhadap perkembangan suatu usaha yang dikelola oleh individu itu sendiri. Informasi yang berkualitas atau bernilai tinggi hanya akan dapat dihasilkan dari sebuah sistem informasi yang juga berkualitas. Banyak sekali manfaat dari penggunaan suatu sistem informasi bagi suatu usaha yaitu akan lebih mudah, dan cepat dalam penyelesaian suatu urusan ataupun pekerjaan.

Tidak hanya di negara-negara maju saja, di Indonesia pun sistem informasi telah banyak diterapkan di mana-mana. Di kantor, pasar, dan bahkan di rumah ketika pemakai bercengkerama dengan dunia internet melalui telepon seluler (ponsel). Sistem informasi telah membantu manusia, salah satunya adalah teknologi informasi berbasis *website*, *Website* adalah suatu halaman yang saling berhubungan yang umumnya berada pada *server* yang sama dan berisikan suatu informasi yang disediakan secara perorangan, kelompok, ataupun organisasi.

Berdasarkan semua permasalahan yang telah dipaparkan serta demi memajukan Kecamatan Tampan, maka judul untuk skripsi ini adalah **Rancang Bangun Sistem Informasi Kos-kosan Berbasis Website di Kecamatan Tampan.**

Keunggulan sistem ini dari sistem yang terdahulu adalah pemilik kos dapat dengan mudah mengelola kos, menampilkan riwayat pengunjung kos, mengetahui berapa lama periode penyewa menyewa kos, dan pada sistem ini juga disediakan fitur *Booking* kamar.

B. Rumusan Masalah

Berdasarkan identifikasi masalah di atas, maka penelitian ini mendapatkan perumusan masalah sebagai berikut :

1. Bagaimana implementasi sistem informasi kos-kosan Kecamatan Tampan?
2. Bagaimana perancangan sistem informasi kos-kosan Kecamatan Tampan?
3. Bagaimana membuat fitur pada *website* sistem informasi kos-kosan yang memudahkan pengunjung?
4. Bagaimana membuat tampilan *interface* yang menarik para pengunjung?

C. Tujuan Penelitian Menjawab Rumusan Masalah

Tujuan dari penelitian adalah sebagai berikut:

1. Membangun sistem informasi pencarian Kos-kosan di Kecamatan Tampan.
2. Memberikan informasi tentang jasa Kos-kosan dalam bentuk *website* yang dapat diakses oleh pengunjung sehingga pengunjung dapat berinteraksi dengan pemilik kos.

D. Manfaat Penelitian

Manfaat dari perancangan dan penelitian ini adalah :

1. Bagi pembaca, penelitian ini bermanfaat memberikan kemudahan mahasiswa untuk mencari kos, di sini menyediakan informasi yang lengkap mengenai tempat kos.
2. Bagi pemilik kos, sistem ini sangat berguna dalam mempromosikan kos tanpa membuat brosur.
3. Bagi penulis, ilmu yang diperoleh selama masa perkuliahan dapat dipraktekkan. Hasil penelitian akan menambah ilmu dan wawasan mengenai *website*.

E. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut:

1. Sistem informasi ini tidak dapat melakukan pembayaran di dalam sistem.
2. Pengujian sistem informasi ini hanya mempunyai ruang lingkup di Kecamatan Tampan.

3. Sistem informasi ini hanya dapat diakses dengan menggunakan jaringan internet.

BAB II

TINJAUAN PUSTAKA

A. Rumah Kos

Menurut Murdayanti, dkk. (2017), rumah kos adalah suatu bidang jasa kecil dan menengah yang merupakan usaha penyewaan kamar kos. Banyak hal yang menjadi pertimbangan memilih usaha kos sebagai alternatif berwirausaha. Kos atau kos-kosan merupakan suatu ladang bisnis yang mulai banyak diminati. Mulai dari kategori bisnis sampingan sampai dengan kategori bisnis serius. Dikatakan kategori bisnis sampingan, karena banyak menggunakan ruangan sisa yang tidak terpakai dalam rumah atau rumah-rumah kosong yang tidak digunakan tapi memiliki nilai tinggi apabila dilakukan sebagai nilai usaha. Untuk kategori bisnis yang serius, para pengusaha ini memang menyediakan ruang khusus atau memiliki lahan untuk dijadikan rumah kos-kosan dan pengelolaannya juga dilakukan dengan serius serta professional.

B. Sistem Informasi

Menurut Kertahadi (dalam Sutiyono, dkk. 2020), sistem informasi adalah “alat untuk menyajikan informasi sedemikian rupa sehingga bermanfaat bagi penerimanya. Tujuannya adalah untuk memberikan informasi dalam perencanaan, memulai, pengorganisasian, operasional sebuah perusahaan yang melayani sinergi organisasi dalam proses mengendalikan pengambilan keputusan” (Dini, 2015).

1. Website

Dimuat dalam jurnal Guntur Wibisono, Susanto (2015), menyatakan bahwa menurut pendapat Arief (2011), “web adalah salah satu aplikasi yang berisikan dokumen–dokumen *multimedia* (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (*hypertext transfer protocol*) dan untuk mengakses menggunakan perangkat lunak yang disebut *browser*”.

2. Internet

a. Pengertian Internet

Menurut Hidayatullah dan Kawistara (2015), “internet adalah jaringan global yang menghubungkan komputer–komputer seluruh dunia, dengan internet sebuah komputer bisa mengakses data yang terdapat pada komputer lain di benua yang berbeda”.

b. *World Wide Web* (WWW)

Menurut Hidayatullah dan Kawistara (2015), *World Wide Web* (WWW) adalah suatu program yang ditemukan oleh Tim Berners-Lee pada tahun 1991. Awalnya Berners-Lee hanya ingin menemukan cara untuk menyusun arsip-arsip risetnya. WWW bekerja berdasarkan pada tiga mekanisme berikut:

- a. Informasi disimpan di dalam dokumen yang disebut halaman *web*.

- b. Halaman web adalah *file-file* yang disimpan dalam komputer yang disebut *web-server*.
- c. Komputer yang mengakses isi dari halaman web disebut *web-clients*.

3. XAMPP

Afifah dalam Riyanto (2018), menyatakan bahwa XAMPP merupakan “paket *web-server* berbasis *open source* yang dapat dipasang pada beberapa sistem operasi yang ada (Windows, Linux, dan Mac OS)” (Afifah, 2018). XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan meng-*install* XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web-server* Apache, PHP dan MySQL secara manual (Amin, 2016).

Menurut Madcoms (2016), “XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla, dan lain.” XAMPP berfungsi untuk memudahkan instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan web memerlukan PHP, Apache, MySQL dan PhpMyadmin”.

4. Basis Data

Fathansyah dalam Abdul Rozaq, dkk (2015), “Basis Data (*Database*) terdiri atas 2 kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat

bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya”.

5. MySQL

Menurut MADCOMS (2016), “MySQL adalah sistem manajemen *Database SQL* yang bersifat *Open Source* dan paling populer saat ini. Sistem *Database MySQL* mendukung beberapa fitur seperti *multithreaded*, *multiuser* dan *SQL Database managemen system (DBMS)*”.

Menurut Arizona (2017), “MySQL adalah bahasa yang digunakan untuk mengelola data pada RDBMS”, sedangkan menurut Risdiansyah (2017) “MySQL merupakan *Database server* yang bersifat *multiuser dan multi-threaded*. SQL adalah bahasa *Database* standar yang memudahkan penyimpanan, pengubahan dan akses informasi. Pada MySQL dikenal istilah *Database* dan tabel. Tabel adalah sebuah struktur data dua dimensi yang terdiri dari baris-baris *record* dan kolom”.

6. Visual Studio Code

Visual Studio Code adalah kode *editor* yang dikembangkan oleh Microsoft untuk *Windows*, *Linux* dan *macOS*. Pada visual studio code juga sudah tersedia *fitur debugging control version* seperti *git* dan

refactoring kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, dan menginstal ekstensi yang menambah fungsionalitas tambahan dengan mudah.

7. **Hypertext Markup Language (HTML)**

Menurut Hidayatullah dan Kawistara (2015), "*Hypertext Markup Language (HTML)* adalah bahasa standar yang digunakan untuk menampilkan halaman web". Yang bisa dilakukan dengan HTML menurut Hidayatullah dan Kawistara (2015) yaitu:

1. Mengatur tampilan dari halaman web dan isinya.
2. Membuat tabel dalam halaman web.
3. Mempublikasikan halam web secara *online*.
4. Membuat *form* yang bisa digunakan untuk menangani registrasi dan transaksi *via* web.

8. **Cascading Style Sheet (CSS)**

Menurut Rohi Abdulloh (2015), *Cascading Style Sheets (CSS)*, yaitu "skrip yang digunakan untuk mengatur desain *website*". Walaupun HTML mempunyai kemampuan untuk mengatur tampilan *website*, namun kemampuannya sangat terbatas. Fungsi CSS adalah Memberikan pengaturan yang lebih lengkap agar struktur *website* yang dibuat dengan HTML terlihat lebih rapi dan indah.

9. **Framework Bootstrap**

Menurut Faried Effendy & Barry Nuqoba (2016), "*bootstrap* adalah *front-end Framework* yang bagus dan luar biasa yang

mengedapkan tampilan untuk *mobile device* (*handphone*, *smartphone* dll.) guna mempercepat dan mempermudah pengembangan *website*” .

10. PHP

Menurut Ripai (2017), mengatakan bahwa “PHP adalah bahasa pemrograman yang bisa disisipkan kedalam *script* HTML dan berjalan di sisi *server*, biasanya dipakai untuk membuat *website* agar lebih dinamis”.

11. Framework Laravel

Menurut I Gede Handika dan Ayi Purbasari (2018), mengatakan bahwa “ *Framework* laravel adalah sebuah kerangka kerja *open source* yang diciptakan oleh Taylor Otwell. Laravel merupakan *Framework bundle*, migrasi dan artisan CLI (*Command Line Interface*) yang menawarkan seperangkat alat dan arsitektur aplikasi yang menggabungkan banyak fitur terbaik dari kerangka kerja seperti Codeigniter, Yii, ASP.NET MVC, Ruby on Rails, Sinatra dan lain-lain. Laravel memiliki seperangkat sangat kaya fitur yang akan meningkatkan kecepatan pengembangan web”.

12. JavaScript

Menurut Ipan Ripai (2017), mengatakan bahwa “JavaScript adalah bahasa *script* yang ditempelkan pada kode HTML dan proses pada sisi klien, sehingga kemampuan dokumen HTML menjadi lebih luas.”

C. Teknik Pengumpulan Data

1. Wawancara

Menurut Singh (dalam Lukman Nul Hakim, 2013), wawancara adalah “situasi berhadapan-hadapan antara pewawancara dan responden yang dimaksudkan untuk menggali informasi yang diharapkan, dan bertujuan mendapatkan data tentang responden dengan minimum bias dan maksimum efisiensi”.

2. Observasi

Menurut Denzin & Lincoln (dalam Hasyim, 2016), mengatakan bahwa “Observasi merupakan langkah awal menuju fokus perhatian lebih luas yaitu observasi partisipan, hingga observasi hasil praktis sebagai sebuah metode dalam kapasitasnya sendiri-sendiri. Observasi ini dapat dilacak pada kemapanan akar teoretis metode interaksionis simbolik, karena dalam mengumpulkan data, peneliti sekaligus dapat berinteraksi dengan subjek penelitiannya”.

Menurut Adler & Adler (dalam Hasyim 2016), menyebutkan bahwa “observasi merupakan salah satu dasar fundamental dari semua metode pengumpulan data dalam penelitian kualitatif, khususnya menyangkut ilmu-ilmu sosial dan perilaku manusia”.

3. Dokumentasi

Menurut Sugiyono dalam Nuning Indah Pratiwi (2017), menyatakan bahwa “Dokumen merupakan catatan peristiwa yang sudah berlalu. Dokumen bisa berbentuk tulisan, gambar, atau karya-karya

monumental dari seseorang.” Dokumen yang digunakan merupakan data pendukung terhadap hasil pengamatan dan wawancara berkaitan dengan bentuk pesan verbal dan non verbal dan juga hambatan-hambatan yang ditemui oleh peneliti. Contohnya seperti dokumentasi saat informan tengah melakukan komunikasi di *video call*.

4. Tinjauan Pustaka

Menurut Maria Magdalena Zagoto, dkk (2019), Metode studi pustaka atau studi literature adalah mengumpulkan informasi dari buku-buku dan referensi lainnya yang terkait dengan masalah dan tujuan riset. Buku-buku dan literatur lain adalah sebagai sumber data yang akan diolah dan dianalisis oleh peneliti. Penelitian dilakukan dengan mengumpulkan sumber kepustakaan untuk mendapatkan informasi/keterangan yang bersifat teoritis.

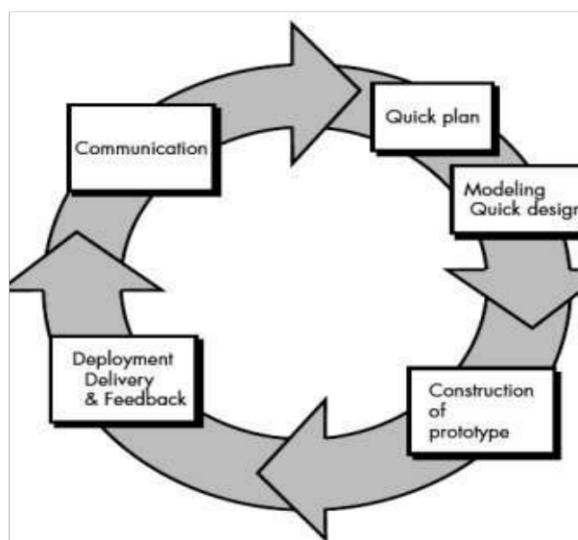
D. Prototyping Model

Prototype adalah model atau simulasi dari alur pengembangan suatu *product*, model ini harus bersifat representatif dari produk akhirnya. Pada pengembangan sistem seringkali terjadi keadaan dimana pengguna sistem sebenarnya telah mendefinisikan secara umum atau tujuan perangkat lunaknya meskipun belum mendefinisikan secara rinci masukan, proses dan keluaran. (Li & Teori, 2019)

Sebuah *Prototype* merupakan versi awal dari sistem perangkat lunak yang digunakan untuk menampilkan konsep-konsep, percobaan rancangan,

dan menemukan lebih banyak masalah dan solusi yang memungkinkan (Pradipta et al, 2015; Sommerville, 2011).

Sistem *Prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik. Penggunaan metode *Prototyping* di dalam penelitian ini bertujuan agar peneliti mendapatkan gambaran aplikasi yang akan dibangun melalui tahap pembangunan aplikasi *Prototype* terlebih dahulu yang akan dievaluasi oleh *user*. Aplikasi *Prototype* yang telah dievaluasi oleh *user* selanjutnya akan dijadikan acuan untuk membuat aplikasi yang dijadikan produk akhir sebagai keluaran dari penelitian ini.



Gambar 2.1 *Prototype Model*
Sumber : Pressman (2010)

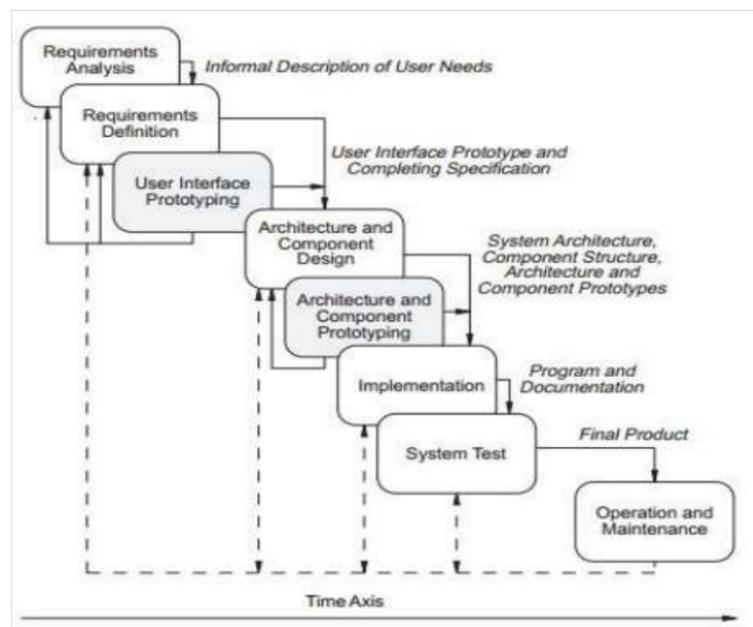
Gambar 2.1 menjelaskan bahwa metode *Prototyping* dimulai dengan mendengarkan kebutuhan dan masukan dari pengguna. Pengembang dan pengguna bertemu dan bersama-sama menentukan tujuan keseluruhan untuk perangkat lunak dan mengidentifikasi apapun persyaratan yang diperlukan.

Lalu pengembang membuat sebuah gambaran tentang aplikasi yang selanjutnya dapat dipresentasikan kepada pelanggan.

Berdasarkan model *Prototype* yang telah digambarkan diatas, maka dapat diuraikan pembahasan masing-masing tahap dalam model tersebut adalah sebagai berikut:

1. *Communication* / Komunikasi tim perancang perangkat lunak melakukan pertemuan dengan para *stakeholder* untuk menentukan kebutuhan perangkat lunak yang saat itu diketahui dan untuk menggambarkan area-area dimana definisi lebih jauh untuk iterasi selanjutnya.
2. *Quick Plan* / perencanaan secara cepat dalam perencanaan ini iterasi pembuatan *Prototype* dilakukan secara cepat. Setelah itu dilakukan pemodelan dalam bentuk “rancangan cepat”.
3. *Modeling Quick Design* / model rancangan cepat pada tahap ini dilakukan pemodelan perencanaan ditahap sebelumnya dengan menggunakan pemodelan terstruktur dalam bentuk DFD (*Data Flow Diagram*), ERD (*Entity Relationship Diagram*) dan *Flowchart* untuk menggambarkan analisis dan desain *system*.
4. *Construction of Prototype* / pembuatan *prototype* dalam pembuatan rancangan cepat berdasarkan pada representasi aspek-aspek perangkat lunak yang akan terlihat oleh para *end user* (misalnya rancangan antarmuka pengguna atau format tampilan). Rancangan cepat merupakan dasar untuk memulai konstruksi pembuatan *Prototype*.

5. *Deployment Delivery & Feedback* / penyerahan dan memberikan umpan balik terhadap pengembangan *prototype* kemudian diserahkan kepada para *stakeholder* untuk mengevaluasi *Prototype* yang telah dibuat sebelumnya dan memberikan umpan-balik yang akan digunakan untuk memperbaiki spesifikasi kebutuhan. Iterasi terjadi saat pengembang melakukan perbaikan terhadap *Prototype* tersebut.



Gambar 2.2 Alur proses metode *Prototyping*

Gambar 2.2 menjelaskan mengenai alur pengembangan sistem dengan menggunakan metode *Prototyping Oriented Software*. Pada tahap pertama, dilakukan analisis kebutuhan dan pendefinisian kebutuhan. Kebutuhan yang dimaksudkan disini adalah kebutuhan pelanggan/pengguna. Selanjutnya pada tahap kedua dilakukan pembuatan *Prototype* dari aplikasi yang akan dibangun, mulai dari *User Interface Prototyping* dan dilanjutkan hingga penyusunan arsitektur dan komponen-komponen yang berkaitan dengan aplikasi yang akan dibangun. Selanjutnya dilakukan pengembangan sistem,

di mana aplikasi akan dibangun sesuai dengan *Prototype* yang telah dibuat sebelumnya, dan setelah aplikasi berhasil dibuat sesuai dengan kebutuhan maka dilakukan proses pengujian aplikasi sebelum aplikasi tersebut diimplementasikan.

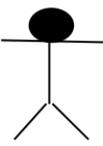
1. Unifed Modeling language (UML)

Menurut Sulianta (2017), dalam buku Teknik Perancangan Arsitektur Sistem Informasi “*Unifed Modeling language (UML)* merupakan kumpulan diagram-diagram yang sudah memiliki standar untuk membangun perangkat lunak berbasis objek”. UML memiliki banyak diagram di antaranya :

2. Use Case Diagram

Use Case Diagram merupakan diagram yang harus dibuat pertama kali saat pemodelan perangkat lunak berorientasi objek dilakukan. Tabel 2.1 menunjukkan simbol yang digunakan untuk membuat *Use Case Diagram* ini antara lain:

Tabel 2.1 *Use Case Diagram*

Simbol	Nama	Keterangan
 Actor 1	Aktor	Merupakan penggunaan dari sistem, penamaan aktor menggunakan kata benda.
 UseCase 1	<i>Use case</i>	Merupakan pekerjaan yang dilakukan oleh aktor penamaan use case oleh kata kerja.

 <p>-End1 –End2</p>	Asosiasi	Hubungan antara aktor dengan <i>use case</i> .
 <p><<use>></p>	<i>Include</i>	Hubungan <i>use case</i> dengan <i>use case</i> , <i>include</i> menyatakan bahwa sebelum pekerjaan dilakukan harus mengerjakan pekerjaan lain terlebih dahulu.
 <p><<extends>></p>	<i>Extends</i>	Hubungan <i>use case</i> dengan <i>use case</i> , <i>extends</i> menyatakan bahwa jika pekerjaan yang dilakukan tidak sesuai atau terdapat kondisi khusus, maka lakukan pekerjaan itu.

3. Class Diagram

Class Diagram dibuat setelah diagram *Use Case* dibuat terlebih dahulu. Pada diagram ini harus menjelaskan hubungan apa saja yang terjadi antara suatu objek dengan objek lainnya sehingga terbentuklah suatu sistem aplikasi.

Simbol yang digunakan untuk membuat *Class Diagram* terlihat pada tabel 2.2.

Tabel 2.2 Simbol-simbol *Class Diagram*

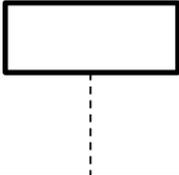
Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antar muka/<i>Interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.

Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .
Asosiasi berarah/ <i>directed association</i> . 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antar kelas dengan makna generalisasi spesialis (umum khusus).
Kebergantungan/ <i>dependency</i> 	Relasi antar kelas dengan makna ketergantungan antar kelas
Agresiasi/agresigation 	Relasi antar kelas dengan makna semua bagian

4. *Sequence Diagram*

Sequence Diagram adalah diagram yang dibuat untuk mengetahui alur dari interaksi antar objek. Isi dari *Sequence Diagram* harus sama dengan *Use Case* dan diagram kelas, seperti yang terlihat pada tabel 2.3.

Tabel 2.2 Simbol-simbol *Sequence Diagram*

Simbol	Nama	Keterangan
	Objek/aktor	Sebuah objek yang berasal dari kelas. Atau dapat dinamai dengan kelasnya saja. Aktor termasuk objek. Garis putus-putus menunjukkan garis hidup suatu objek.
	Aktivasi	Menunjukkan masa hidup objek

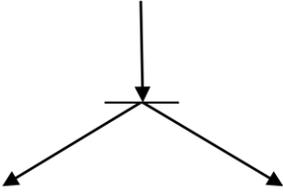
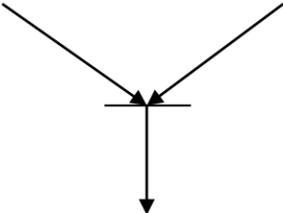
<i>Message 1</i> 	Pesan	Interaksi antar satu objek dengan objek lainnya. Objek dapat mengirimkan pesan ke objek lain. Interaksi antar objek ditunjukkan pada bagian operasi pada diagram kelas.
<i>Message 2</i> 	Return	Pesan kembalian dari komunikasi antar objek

Sumber : Feri Sulianta, Arsitektur Sistem Informasi, 2017

5. Activity Diagram

Menggambarkan aktifitas-aktifitas, *objek*, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas.

Tabel 2.3 Simbol-simbol Activity Diagram

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas
	<i>End point</i> , akhir aktivitas
	<i>Activitas</i> , menggambarkan suatu proses/ kegiatan bisnis
	<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menghubungkan dua kegiatan paralel menjadi satu.
	<i>Joint</i> (penggabungan) atau <i>rake</i> digunakan untuk menunjukkan adanya dekomposisi.

Sumber : Ade Hendini, 2016.

E. Populasi dan *Sample*

1. Populasi

Menurut Sugiyono dalam Risma Dwi Komala,dkk (2016), dalam bukunya mengemukakan mengenai Populasi adalah: “Wilayah generalisasi yang terdiri atas obyek/subyek yang mempunyai kualitas dan karakteristik tertentu yang ditetapkan oleh penelitian untuk dipelajari dan kemudian ditarik kesimpulan”.

2. *Sample*

Menurut Sugiyono dalam Risma Dwi Komala,dkk (2014), dalam bukunya “mengemukakan mengenai Sampel adalah bagian dari jumlah dan karakteristik yang dimiliki oleh populasi”.

F. *Black box Testing*

Menurut Lila Setiyani (2018), “*Black box Testing* merupakan strategi pengujian yang memperhatikan atau memfokuskan kepada faktor fungsionalitas dan spesifikasi perangkat lunak”.

BAB III

METODE PENELITIAN

A. Lokasi dan Subjek Penelitian

Dalam penulisan proposal skripsi ini penulis melakukan penelitian di daerah Kecamatan Tampan, Kota Pekanbaru, Provinsi Riau. Lokasi tersebut dipilih karena saat proses penelitian, penelitian menyadari bahwasanya sangat sulit mencari Kos-kosan yang kita inginkan jika kita bertanya oleh sebab itu, penulis memilih Kecamatan Tampan sebagai lokasi penelitian.

Subjek dalam penelitian ini adalah pemilik kos. Penulis menjadikan seluruh pemilik kos sebagai subjek penelitian. Selain itu untuk menunjang kelengkapan data penelitian, mahasiswa juga menjadi subjek penelitian.

B. Populasi dan *Sample*

1. Populasi

Populasi yang digunakan dalam penelitian ini sebanyak 67 Kos-kosan yang tersebar di Kecamatan Tampan. Sumber data (Trovit.com).

2. *Sample*

Pemilik kos yang akan dijadikan sasaran *sample* penelitian menggunakan taraf kesalahan 5%. Untuk menghitung penentuan jumlah sampel dari populasi tertentu yang dikembangkan, maka pengambilan sampel menggunakan rumus Slovin.

C. Metode Pengumpulan Data

1. Observasi

Observasi dilaksanakan dengan pengamatan atau peninjauan langsung di Kecamatan Tampan, Kota Pekanbaru. Dalam pelaksanaan observasi ini, penulis melakukan studi di:

Nama tempat : Kecamatan Tampan, Kota Pekanbaru.

Waktu Pelaksanaan: 25 April 2021 sampai 29 April 2021.

Observasi ini digunakan untuk mendapatkan data yang dibutuhkan dalam penelitian sistem informasi Kos-kosan di Kecamatan Tampan.

2. Wawancara

Wawancara atau *interview* adalah sebuah proses memperoleh keterangan untuk tujuan peneliti dengan cara tanya jawab sambil bertatap muka antara pewawancara dengan orang yang diwawancarai, dengan atau tanpa menggunakan pedoman wawancara. Peneliti melakukan wawancara dengan pemilik kos.

Tabel 3. 1 Pedoman wawancara

No	Aspek yang ditanyakan	Indikator	Item pertanyaan
1	Riwayat pendidikan pemilik kos	a. Riwayat pendidikan pemilik kos b. Pendidikan terakhir pemilik kos	<ul style="list-style-type: none"> • Bagaimana riwayat pendidikan anda? • Apa pendidikan terakhir anda?
2	Fitur-fitur kos	a. Fitur-fitur yang diinginkan	<ul style="list-style-type: none"> • Seperti apa fitur yang ibu/bapak inginkan di dalam <i>website</i> Kos-kosan? • Apakah sebelumnya ibu/bapak pernah berlangganan pada <i>website</i> Kos-kosan? • Jika pernah apa fitur yang ibu/bapak inginkan tetapi tidak ada pada <i>website</i> penyedia Kos-kosan tersebut? • Jika tidak pernah. Apakah kesulitan ibu/bapak dalam mengelola Kos-kosan?

3. Dokumentasi

Suatu pengumpulan data dengan cara melihat langsung sumber-sumber dokumen yang terkait. Dengan arti lain bahwa dokumentasi sebagai pengambilan data melalui dokumen tertulis maupun elektronik. Digunakan sebagai pendukung kelengkapan data yang lain.

4. Studi Pustaka

Peneliti melakukan studi pustaka dengan cara membaca dan mempelajari jurnal-jurnal dan artikel terkait yang berhubungan dengan analisa dan perancangan sistem, pemrograman *web* serta jurnal-jurnal yang mendukung metode pengembangan sistem *Prototype* yang dibahas pada penyusunan proposal ini. Terdapat beberapa jurnal dan artikel-artikel yang dapat dijadikan referensi dalam penyusunan proposal ini dan untuk daftar jurnal dan artikel-artikel dapat dilihat pada bagian daftar pustaka.

D. Metode Analisis

Metode Analisis yang digunakan yaitu:

1. Metode Kualitatif

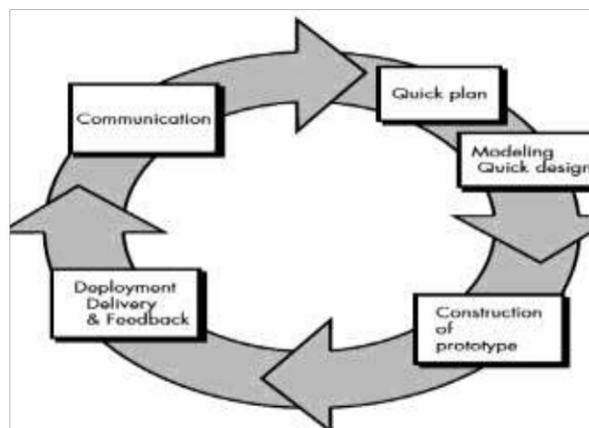
Untuk melakukan proses meneliti Kos-kosan peneliti menggunakan metode kualitatif. Dengan metode ini penelitian melakukan wawancara dan observasi secara langsung. Tujuannya agar mendapatkan informasi dan data yang akurat mengenai Kos-kosan yang ada di Kecamatan Tampan.

2. Metode Pengamatan

Sistem pengamatan dilakukan secara rinci terhadap fungsi-fungsi sistem yang terdapat pada sistem. Sehingga dapat diketahui kekurangan dari sistem tersebut.

E. Metode Perancangan

1. *Prototype*



Gambar 3. 1 Metode *Prototype*

Sumber: (Pressman: 2010)

Metode yang digunakan dalam pengembangan sistem ini adalah *Prototype*. Pengertian dari *Prototype* yaitu suatu teknik pengumpulan data yang berguna mengumpulkan informasi tertentu mengenai pengguna, *Prototype* memiliki ciri-ciri pengembang dan pelanggan dapat secara langsung melihat proses terjadinya pengembangan sistem, dalam pengembangan *prototype* ini pelanggan dapat melakukan evaluasi kembali terhadap rancangan yang telah dibuat, dengan menggunakan model pengembangan *prototype* akan sangat membantu bagi *developer* untuk menjaga efisiensi dari algoritma yang digunakan, proses pada model *prototype* bisa dijelaskan dengan beberapa tahap sebagai berikut:

a. *Communication*

Pada tahap ini dilakukan analisis terhadap kebutuhan pengguna terhadap sistem, pada tahap ini sangat dibutuhkan komunikasi untuk mencapai sebuah tujuan yang ingin dicapai, tahap *communication* atau pengumpulan data dapat dilakukan dengan cara melakukan wawancara langsung kepada pelanggan.

b. *Quick plan*

Pada tahap *quick plan* akan dibuat rancangan *prototype*, dan memperkirakan sumber-sumber daya yang diperlukan, produk yang akan dihasilkan, dan melakukan *tracking* dalam proses pengerjaan sistem.

c. *Modelling Quick Design*

Pada tahap ini dilakukan pemodelan perencanaan ditahap sebelumnya, dengan menggunakan pemodelan berorientasi objek dalam bentuk *Class Diagram*, untuk menggambarkan dan menganalisis *design* sistem.

d. *Construction of Prototype*

Pada tahap ini akan dilakukan pembuatan *prototype* dengan memperlihatkan tampilan *interface* kepada para *end user*.

e. *Deployment Delivery & Feedback*

Prototype yang telah dibuat pada tahap sebelumnya kemudian diserahkan kepada para *stakeholder* untuk dilakukan evaluasi dan menerima *feedback* yang akan digunakan untuk memperbaiki spesifikasi kebutuhan.

F. Teknik Pengujian

Metode pengujian dilakukan pada penelitian ini adalah metode pengujian langsung yaitu dengan pengujian *black box*. Pengujian *black box* digunakan untuk menguji fungsi khusus dari perangkat lunak yang dirancang. Perangkat lunak diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai sehingga dapat diketahui kesalahannya.

BAB IV

HASIL DAN PEMBAHASAN

A. Analisa Sistem Gejala

Analisa sistem berjalan merupakan langkah awal yang harus dilaksanakan sebelum Memberikan sistem usulan, berdasarkan analisa ini nantinya akan dijadikan alasan kenapa sistem usulan dibutuhkan. Analisa sistem yang sudah ada dapat berupa identifikasi permasalahan, dari hasil identifikasi tersebut telah ditemukan beberapa masalah yang dapat diselesaikan. Seperti pencarian kos-kosan yang mengharuskan calon penyewa melakukan survei lokasi kos-kosan tersebut, dan pemilik kos yang kesulitan menyampaikan informasi detail kos-kosan, yang mana sebelumnya pemilik kos mempromosikan jasa kos-kosan melalui brosur dan spanduk tentu itu kurang efisien.

B. Identifikasi Permasalahan

Berdasarkan observasi yang peneliti lakukan pada kos-kosan, ditemukan suatu permasalahan dalam proses penyampaian informasi yang berkaitan dengan detail kos, fasilitas serta layanan kos-kosan. Saat ini alur proses calon penyewa rumah kos-kosan untuk mendapatkan informasi mengenai detail, fasilitas serta layanan kos-kosan masih belum efektif. Informasi didapatkan oleh mahasiswa atau calon penyewa dengan cara mendatangi langsung ke lokasi kos-kosan dan menanyakan secara langsung kepada pihak pemilik kos-kosan, dengan kondisi seperti ini tentu

penyampaian informasi tidak akurat dan tidak efisien. Sistem pencarian Kos-kosan sekitar Kecamatan Tampan yang ada saat ini belum dilengkapi dengan Informasi mengenai Detail Kos-kosan yang masih terbatas, seperti Kos-kosan di surat kabar atau koran, tulisan terima kos didepan rumah kos, dan juga informasi kontak pemilik Kos-kosan yang terdapat di selebaran kertas yang ditempel pada pohon sekitar jalan rumah kos.

C. Analisa Sistem Usulan

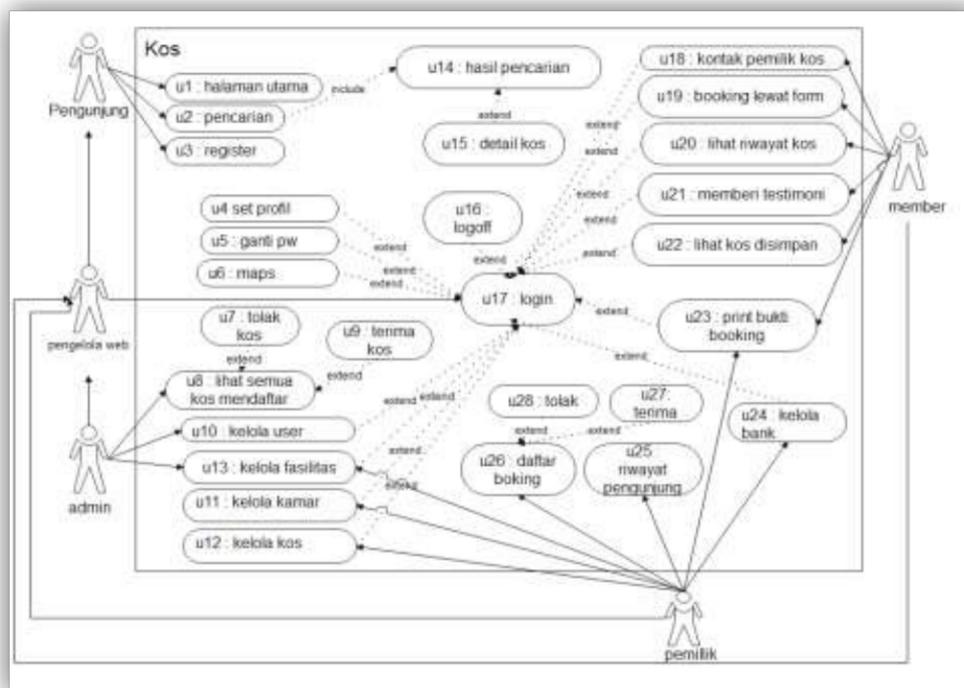
Pada tahap ini, perancangan sistem usulan yang akan dibangun berguna untuk pemilik kos dalam mempromosikan jasa kos-kosan dengan detail serta pemilik kos dapat mengelola kos-kosan lebih mudah, seperti mengetahui kapan calon penyewa melakukan *booking* serta mengetahui kapan penyewa sudah habis masa kosnya, sehingga penyampaian informasi kos-kosan lebih efektif dan efisien, sistem yang akan dibangun menggunakan bahasa pemrograman PHP dengan *framework* laravel.

D. Perancangan Sistem dengan *Unified Modelling Language* (UML)

Pada tahap perancangan sistem akan digunakan *unified modelling language* (UML), digunakan untuk memvisualisasikan dan menggambarkan sebuah sistem informasi, diagram UML yang akan digunakan ialah *Use Case Diagram* sebagai *tools* untuk memperlihatkan hubungan pengguna dengan sistem, dan *Class Diagram* sebagai *tools* untuk menggambarkan struktur sebuah sistem.

1. Use Case Diagram

Digunakan untuk menggambarkan kegiatan-kegiatan yang dilakukan oleh pengguna pada sistem. Berikut gambaran aliran *Use Case Diagram* dapat dilihat pada gambar 4.1.



Gambar 4.1 Use Case Diagram

a. Bussines Actor

Bussines actor menggambarkan interaksi pengguna dengan sistem seorang *actor* dapat memberikan informasi masukan kepada sistem, menerima informasi dari sistem dan dapat melakukan keduanya, menerima dan memberi informasi pada sistem, terdapat 5 *actor* pada sistem ini yaitu pengunjung, pengelola web, admin, pemilik kos dan *member*.

b. *Business Use Case*

Setiap *Use Case* adalah cuplikan bisnis, pada tahap ini *Use Case* dapat melibatkan komunikasi dua arah antara sejumlah *actor*. Berikut adalah *list business Use Case*:

- 1) B1 : Calon penyewa kos mendatangi Kos-kosan atau melihat informasi kos yang tersedia.
- 2) B2 : Calon penyewa menyewa kos dengan *via* telepon atau mendatangi kos tersebut.
- 3) B3 : Penyewa yang melakukan *Booking* harus membayar uang muka.
- 4) B4 : Pemilik kos Memberikan kunci kos kepada penyewa kos.
- 5) B5 : Penyewa yang sudah jatuh tempo harus membayar jika ingin memperpanjang masa kosnya.
- 6) B6 : Jika penyewa kos tidak memperpanjang masa kos maka penyewa harus mengembalikan kunci kos kepada pemilik kos.

c. *System Use Case*

Use Case system menyediakan alur kerja khusus *system* berikutnya. Adapun *system Use Case* sebagai berikut:

- 1) U1 Halaman utama: menampilkan halaman utama untuk seluruh pengunjung.

- 2) U2 *Filter* pencarian: *filter* pencarian untuk mencari kos-kosan sesuai apa yang dicari.
- 3) U3 *Register*: pengunjung dapat melakukan *Register* pada sistem.
- 4) U4 *Setting profile*: pengguna dapat melakukan *Setting profile*.
- 5) U5 *Ganti Password*: pengguna dapat melakukan perubahan *Password*.
- 6) U6 *Maps*: pengguna dapat menggunakan *maps* untuk mencari lokasi kos.
- 7) U7 *Lihat semua kos mendaftar*: admin dapat melihat semua kos yang mendaftar.
- 8) U8 *Terima kos*: Admin dapat menerima kos-kosan yang mendaftar.
- 9) U9 *Tolak kos*: Admin dapat menolak kos-kosan yang mendaftar.
- 10) U10 *Kelola user*: Admin dapat mengelola semua *user* yang terdaftar pada web.
- 11) U11 *Kelola kamar*: pemilik kos dapat mengelola kamar kos.
- 12) U12 *Kelola kos*: pemilik kos dapat mengelola kos.
- 13) U13 *Kelola fasilitas*: Pemilik kos dan admin dapat mengelola fasilitas kos.

- 14) U14 Hasil pencarian: pengunjung dapat melihat hasil pencarian kos-kosan.
- 15) U15 Detail kos: pengunjung dapat melihat detail kos yang telah dicari.
- 16) U16 *Logoff*: pengunjung dapat melakukan *logoff*.
- 17) U17 *Login*: pengunjung dapat melakukan *login* kedalam sistem.
- 18) U18 Kontak pemilik kos: pengunjung dapat menghubungi pemilik kos.
- 19) U19 *Booking* kos *via form*: pengunjung dapat melakukan *Booking* melalui *form Booking*.
- 20) U20 Lihat riwayat kos: pengunjung kos dapat melihat riwayat kosnya.
- 21) U21 Memberikan testimonial: pengunjung memberikan testimoni pada kos yang telah dikunjungi.
- 22) U22 Lihat kos disimpan: pengunjung dapat melihat kos favoritnya.
- 23) U23 *Print* bukti *Booking* kos: pengunjung dapat *print* bukti *Booking* kosnya.
- 24) U24 Kelola rekening *bank*: pemilik kos dapat mengelola daftar rekening *bank*.
- 25) U25 Lihat riwayat pengunjung: pemilik kos dapat melihat riwayat kunjungan kos dari pengunjung kos.

- 26) U26 Lihat daftar *Booking*: pemilik kos dapat melihat daftar *Booking* kos.
- 27) U27 Terima *Booking* kos: pemilik kos dapat menerima *Booking* kos.
- 28) U28 Tolak *Booking* kos: pemilik kos dapat menolak *Booking* kos.

d. Deskripsi Use Case Diagram

Pada tahap ini menjelaskan deskripsi *Use Case* antara *actor* dan sistem.

- 1) Berikut adalah penjelasan deskripsi *Use Case* pengunjung sistem informasi kos-kosan di Kecamatan Tampan seperti tabel 4.1.

Tabel 4.1 Deskripsi Use Case Pengunjung

No	Use Case	Deskripsi
1	U1 Halaman utama	Menampilkan halaman utama untuk seluruh pengunjung
2	U2 Pencarian	Dapat melakukan pencarian kos
3	U3 <i>Register</i>	pengunjung dapat melakukan <i>Register</i> pada sistem.
4	U14 Hasil pencarian	pengunjung dapat melihat hasil pencarian Kos-kosan.
5	U15 Detail kos	pengunjung dapat melihat Detail kos yang telah dicari.

- 2) Berikut adalah penjelasan deskripsi *Use Case* pengelola web sistem informasi kos-kosan di Kecamatan Tampan seperti tabel 4.2.

Tabel 4.2 Deskripsi *Use Case* Pengelola Web

No	<i>Use Case</i>	Deskripsi
1	U17 <i>Login</i>	pengunjung dapat melakukan <i>Login</i> kedalam sistem.
2	U4 <i>Setting profie</i>	pengguna dapat melakukan <i>Setting profile</i> .
3	U5 <i>Ganti Password</i>	pengguna dapat melakukan perubahan <i>Password</i> .
4	U6 <i>Maps</i>	pengguna dapat menggunakan <i>maps</i> untuk mencari lokasi kos.
5	U16 <i>Logoff</i>	pengunjung dapat melakukan <i>logoff</i> .

- 3) Berikut adalah penjelasan deskripsi *Use Case* admin sistem informasi kos-kosan di Kecamatan Tampan seperti tabel 4.3.

Tabel 4.3 Deskripsi *Use Case* Admin

No	<i>Use Case</i>	Deskripsi
1	U10 <i>Kelola user</i>	Admin dapat mengelola semua <i>user</i> yang terdaftar pada web.
2	U13 <i>Kelola fasilitas</i>	Pemilik kos dan admin dapat mengelola fasilitas kos.
3	U7 <i>Lihat semua kos mendaftar</i>	Admin dapat melihat semua kos yang mendaftar.
4	U8 <i>Terima kos</i>	Admin dapat menerima Kos-kosan yang mendaftar.
5	U9 <i>Tolak kos</i>	Admin dapat menolak Kos-kosan yang mendaftar.

- 4) Berikut adalah penjelasan deskripsi *Use Case* pemilik kos sistem informasi kos-kosan di Kecamatan Tampan seperti tabel 4.4.

Tabel 4.4 Deskripsi *Use Case* Pemilik Kos

No	<i>Use Case</i>	Deskripsi
1	U26 <i>Lihat daftar Booking</i>	Pemilik kos dapat melihat daftar <i>Booking</i> kos.
2	U27 <i>Terima Booking kos</i>	Pemilik kos dapat menerima <i>Booking</i> kos.
3	U28 <i>Tolak Booking</i>	Pemilik kos dapat menolak <i>Booking</i> kos.
4	U24 <i>Kelola rekening bank</i>	Pemilik kos dapat mengelola daftar rekening <i>bank</i> .
5	U25 <i>Riwayat pengunjung</i>	Pemilik kos dapat melihat riwayat kunjungan kos dari pengunjung kos.

6	U23 Print bukti <i>Booking</i> kos	Pengunjung dapat print bukti <i>Booking</i> kos.
7	U12 Kelola kos	Pemilik kos dapat mengelola kos.
8	U11 Kelola kamar	Pemilik kos dapat mengelola kamar kos.
9	U13 Kelola fasilitas	Pemilik kos dan admin dapat mengelola fasilitas kos.

- 5) Berikut adalah penjelasan deskripsi *Use Case Member* sistem informasi kos-kosan di Kecamatan Tampan seperti tabel 4.5.

Tabel 4.5 Deskripsi Use Case Member

No	<i>Use Case</i>	Deskripsi
1	U18 Kontak pemilik kos	Pengunjung dapat menghubungi pemilik kos.
2	U19 <i>Booking</i> kos <i>via form</i>	Pengunjung dapat melakukan <i>Booking</i> melalui <i>form Booking</i> .
3	U20 Lihat riwayat kos	Pengunjung kos dapat melihat riwayat kos.
4	U21 Memberikan testimonial	Pengunjung Memberikan testimoni pada kos yang telah dikunjungi.
5	U22 Lihat kos disimpan	Pengunjung dapat melihat kos favorit.
6	U23 <i>Print</i> bukti <i>Booking</i> kos	Pengunjung dapat <i>print</i> bukti <i>Booking</i> kos.

2. *Class Diagram*

Class Diagram merupakan penjelasan lengkap dari beberapa *class* yang ditangani oleh sistem. Dimana tiap-tiap *class* dipasangkan dengan beberapa *attribute* dan *method* yang diperlukan. Berikut adalah skema dari model *Class Diagram* sistem informasi kos-kosan di Kecamatan Tampan yang dapat dilihat pada gambar 4.2.

Tabel 4.6 Kelas, Atribut, dan Method

No	Nama kelas	Atribut	Method
1	<i>Users</i>	id : int nama : string email : string <i>Password</i> : string alamat : text no_hp : int Jenis_kelamin : text Status_kerja : string Avatar : string Created_at : timestamp Update_at : timestamp	+ <i>Login</i> () + <i>Register</i> () + Create () + Update () + Delete () + read () + <i>getBooking</i> () + <i>changePassword</i> () + <i>makeBlog</i> ()
2	Kamar	id : int kos_id : int luas_kamar : string ukuran_kamar : string jumlah_kasur : int biaya_bulanan : int status : string Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
3	Gambar_kamar	id : int kamar_id : int gambar : string keterangan : text Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
4	Fasilitas_kamar	ID : int kamar_id : int fasilitas_id : int Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
5	Kos	ID : int Nama_kos : string user_id : int type_kos : string latitude : string longitude : string alamat : text slug : string deskripsi_kos : text status : string <i>Is_Booking</i> : boolean <i>Biaya_Booking</i> : string Created_at : timestamp	+ Create () + Read () + Update () + Delete () + <i>maps_direction</i> () + <i>maps_location</i> ()

No	Nama kelas	Atribut	Method
		update_at : timestamp	
6	Kos_fasilitas	id : int kos_id : int fasilitas_id : int Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
7	Gambar_kos	ID : int kos_id : string lokasi : string ket : text gambar : string Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
8	Testimonial	ID : int user_id : int kos_id : int kesimpulan : text Created_at : timestamp update_at : timestamp	+ Create () + Read ()
9	Kos_tersimpan	id : int user_id : int kos_id : int Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
10	<i>Booking</i>	id : int user_id : int bank_id : int masa_inap : string nominal : int bukti_bayar : string Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
11	<i>Booking_Detail</i>	id : int <i>Booking_id</i> : int kost_id : int kamar_id : int biaya_sewa : int Created_at : timestamp update_at : timestamp	+ Read () + Print ()
12	Blog	id : int title : string	+ Create () + Read ()

No	Nama kelas	Atribut	Method
		slug : string kategori : string isi : string gambar : string user_id : int Created_at : timestamp update_at : timestamp	+ Update () + Delete ()
13	Bank	id : int user_id : string Nama_bank : string no_rek : int Nama_nasabah : string Status : string Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()

Berdasarkan tabel 4.6 terdapat 13 *class* yang dibangun pada sistem informasi kos-kosan di Kecamatan Tampan antara lain:

- a. *Users* merupakan *class* yang berisi data-data pengguna dengan berbagai *role*.
- b. Kamar merupakan *class* yang berisi data-data kamar.
- c. Gambar kamar merupakan *class* yang menampung data-data gambar kamar berdasarkan kamar tertentu.
- d. Fasilitas kamar merupakan *class* yang menampung data-data fasilitas kamar.
- e. Fasilitas kos merupakan *class* yang menampung data-data fasilitas kos.
- f. Kos merupakan *class* yang menampung seluruh data-data kos.

- g. Gambar kos merupakan *class* yang menampung data-data gambar kos berdasarkan kos tertentu.
- h. Testimonial merupakan *class* yang menampung data-data testimonial Kos-kosan yang berikan pengguna.
- i. Kos tersimpan merupakan *class* yang menampung data kos-kos yang disimpan oleh pengguna pada halaman *Dashboard*.
- j. *Booking* merupakan *class* yang menampung data-data *Booking* pengguna.
- k. *Booking Detail* merupakan *class* yang menampung Detail *Booking* pengguna.
- l. *Bank* merupakan *class* yang menampung data-data nomor rekening pemilik kos.
- m. Blog merupakan *class* yang menampung data-data Blog *ger* atau berita.

E. Perancangan *Database*

Dalam sistem yang dibangun terdapat beberapa bisnis proses yang ingin diselesaikan adalah rancang bangun sistem informasi Kos-kosan berbasis *website* di Kecamatan Tampan. Pada perancangan sistem ini terdapat 13 tabel.

1. Merancang Relasi Antar Tabel

Pada tahapan ini tampungan data dan relasi tabel pada sistem informasi Kos-kosan berbasis *website* di Kecamatan Tampan yang dibuat dengan menyesuaikan antara bisnis proses dan sistem yang

a. Tabel Users

Tabel *Users* digunakan untuk menampung semua data-data pengguna sistem yang dapat dilihat pada tabel 4.7.

Tabel 4.7 Users

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Name</i>	<i>Varchar(60)</i>	Nama pengguna
3	<i>Email</i>	<i>Varchar(60)</i>	<i>Email</i> pengguna
4	<i>Password</i>	<i>Varchar(60)</i>	<i>Password</i> pengguna
5	<i>Avatar</i>	<i>Varchar(60)</i>	Foto pengguna
6	<i>Roles</i>	<i>Varchar(60)</i>	<i>Role</i> pengguna
7	No_hp	<i>Integer</i>	Nomor telepon pengguna
8	Alamat	<i>Longtext</i>	Alamat pengguna
9	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
10	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

b. Tabel Kos

Tabel kos digunakan untuk menampung semua data-data kos-kosan yang dapat dilihat pada tabel 4.8.

Tabel 4.8 Kos

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>User_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	Nama_kos	<i>Varchar(60)</i>	Nama Kos-kosan yang didaftarkan

No	Nama Kolom	Tipe Data	Keterangan
4	<i>Slug</i>	<i>Varchar(60)</i>	Digunakan untuk mesin pencarian Kos-kosan
5	Alamat	<i>Longtext</i>	Alamat dari Kos-kosan
6	<i>Type_kos</i>	<i>Varchar(60)</i>	Kos dikhususkan untuk laki-laki, perempuan atau keduanya
7	Aturan_kos	<i>Longtext</i>	Aturan yang digunakan pada Kos-kosan
8	Deskripsi_kos	<i>Longtext</i>	Deskripsi atau penjelasan singkat tentang Kos-kosan
9	Luas_kos	<i>Integer</i>	Luas kos pada Kos-kosan
10	<i>Is_Booking</i>	<i>Boolean</i>	Pemilik kos menentukan memakai atau tidaknya fitur <i>Booking</i>
11	<i>Biaya_Booking</i>	<i>Varchar(60)</i>	Jika pemilik kos menggunakan fitur <i>Booking</i> maka pemilik kos menentukan biaya awal untuk <i>Booking</i>
12	<i>Latitude</i>	<i>Varchar(60)</i>	Titik koordinat lokasi kos pada peta
13	<i>Longitude</i>	<i>Varchar(60)</i>	Titik koordinat lokasi kos pada peta
14	Status	<i>Varchar(60)</i>	Status kos pada <i>website</i>
15	<i>Created_at</i>	<i>Timestamp</i>	Tanggal Kos-kosan didaftarkan pada sistem
16	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal Kos-kosan melakukan <i>update</i>

c. Tabel Kamar

Tabel kamar digunakan untuk menampung semua data-data kamar yang dapat dilihat pada tabel 4.9.

Tabel 4.9 Kamar

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Kos_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	Luas_kamar	<i>Char(60)</i>	Luas kamar kos
4	Ukuran_kamar	<i>Char(60)</i>	Ukuran kamar kos
5	Jumlah_kasur	<i>Integer</i>	Jumlah kasur pada kamar

No	Nama Kolom	Tipe Data	Keterangan
6	Biaya_perbulan	<i>Integer</i>	Harga kamar tiap periode bulanan
7	Status	<i>Varchar(60)</i>	Status ketersediaan kamar
8	<i>Created_at</i>	<i>Timestamp</i>	Tanggal kamar dibuat pada sistem

d. Tabel Gambar Kamar

Tabel gambar kamar digunakan untuk menampung semua data-data gambar kamar yang dapat dilihat pada tabel 4.10.

Tabel 4.10 Gambar Kamar

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Kamar_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	Gambar	<i>Varchar(60)</i>	Kumpulan gambar-gambar kamar
4	<i>Created_at</i>	<i>Timestamp</i>	Tanggal dibuatnya gambar-gambar kamar
5	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal gambar kamar diperbarui

e. Tabel Gambar Kos

Tabel gambar kos digunakan untuk menampung semua data-data gambar kos yang dapat dilihat pada tabel 4.11.

Tabel 4.11 Gambar Kos

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Kos_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	Gambar	<i>Varchar(60)</i>	Kumpulan gambar-gambar kos
4	<i>Created_at</i>	<i>Timestamp</i>	Tanggal dibuatnya gambar-gambar kos

No	Nama Kolom	Tipe Data	Keterangan
5	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal gambar kos diperbarui

f. Tabel Fasilitas

Tabel fasilitas digunakan untuk menampung semua data-data gambar kos yang dapat dilihat pada tabel 4.12.

Tabel 4.12 Fasilitas

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Nama_fasilitas</i>	<i>Varchar(60)</i>	Nama fasilitas
3	<i>Icon</i>	<i>Varchar(60)</i>	<i>Icon</i> fasilitas
4	<i>Keterangan</i>	<i>Varchar(60)</i>	Keterangan fasilitas
5	<i>Created_at</i>	<i>Timestamp</i>	Tanggal dibuatnya fasilitas
6	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal gambar kamar diperbarui

g. Tabel Fasilitas Kos

Tabel fasilitas kos digunakan untuk menampung semua data-data fasilitas kos yang dapat dilihat pada tabel 4.13.

Tabel 4.13 Fasilitas Kos

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Kos_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	<i>Fasilitas_id</i>	<i>Integer</i>	<i>Foreign key</i>
4	<i>Created_at</i>	<i>Timestamp</i>	Tanggal dibuatnya fasilitas kos
5	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal fasilitas kos diperbarui

h. Tabel Fasilitas Kamar

Tabel fasilitas kamar digunakan untuk menampung semua data-data fasilitas kamar yang dapat dilihat pada tabel 4.14.

Tabel 4.14 Fasilitas Kamar

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Kamar_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	<i>Fasilitas_id</i>	<i>Integer</i>	<i>Foreign key</i>
4	<i>Created_at</i>	<i>Timestamp</i>	Tanggal dibuatnya fasilitas kamar
5	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal gambar kamar diperbarui

i. Tabel *Booking*

Tabel *Booking* digunakan untuk menampung semua data-data *Booking* yang dapat dilihat pada tabel 4.15.

Tabel 4.15 *Booking*

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>User_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	<i>Bank_id</i>	<i>Integer</i>	Foreign key
4	Bukti_bayar	<i>Varchar(60)</i>	Bukti bayar / bukti <i>Booking</i>
5	Mulai_sewa	<i>Date</i>	Tanggal mulai melakukan <i>Booking</i>
6	Habis_sewa	<i>Date</i>	Tanggal habis masa sewa

j. *Booking Detail*

Tabel *Booking Detail* digunakan untuk menampung semua data-data *Booking Detail* yang dapat dilihat pada tabel 4.16.

Tabel 4.16 *Booking Detail*

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Booking_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	<i>Kos_id</i>	<i>Integer</i>	<i>Foreign key</i>
4	<i>Kamar_id</i>	<i>Integer</i>	<i>Foreign key</i>
5	Total_bayar	<i>Integer</i>	Total bayar untuk <i>Booking</i>
6	Lama_sewa	<i>Integer</i>	Lama sewa kamar dalam hitungan bulan
7	Status	<i>Varchar(60)</i>	Status <i>Booking_Detail</i>
8	<i>Created_at</i>	<i>Timestamp</i>	Tanggal dibuatnya <i>Booking_Detail</i>
9	<i>Updated_at</i>	<i>Timestamp</i>	-

k. **Tabel Bank**

Tabel bank digunakan untuk menampung semua data-data bank yang dapat dilihat pada tabel 4.17.

Tabel 4.17 **Bank**

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>User_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	<i>No_rek</i>	<i>Varchar(60)</i>	Nomor rekening pemilik kos
4	Nama_bank	<i>Varchar(60)</i>	Nama <i>bank</i> pemilik kos

No	Nama Kolom	Tipe Data	Keterangan
5	Nama_nasabah	<i>Varchar(60)</i>	Nama pemilik nomor rekening
6	Status	<i>Varchar(60)</i>	Status <i>bank</i>
7	<i>Created_at</i>	<i>Timestamp</i>	Tanggal informasi <i>bank</i> ditambahkan
8	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal informasi <i>bank</i> diperbarui

l. Tabel Testimonial

Tabel testimonial digunakan untuk menampung semua data-data testimonial yang dapat dilihat pada tabel 4.18.

Tabel 4.18 Testimonial

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>User_id</i>	<i>Integer</i>	<i>Foreign key</i>
3	<i>Kos_id</i>	<i>Integer</i>	<i>Foreign key</i>
4	Testimonial	<i>Varchar(100)</i>	Konten atau isi dari testimonial
5	<i>Created_at</i>	<i>Timestamp</i>	Tanggal testimonial ditambahkan
6	<i>Updated_at</i>	<i>Timestamp</i>	-

m. Tabel Blog

Tabel Blog digunakan untuk menampung semua data-data bank yang dapat dilihat pada tabel 4.19.

Tabel 4.19 Blog

No	Nama Kolom	Tipe Data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Title</i>	<i>Varchar(60)</i>	<i>Title Blog</i>
3	<i>Slug</i>	<i>Varchar(60)</i>	<i>Slug title Blog</i> , untuk mempercantik <i>url</i>

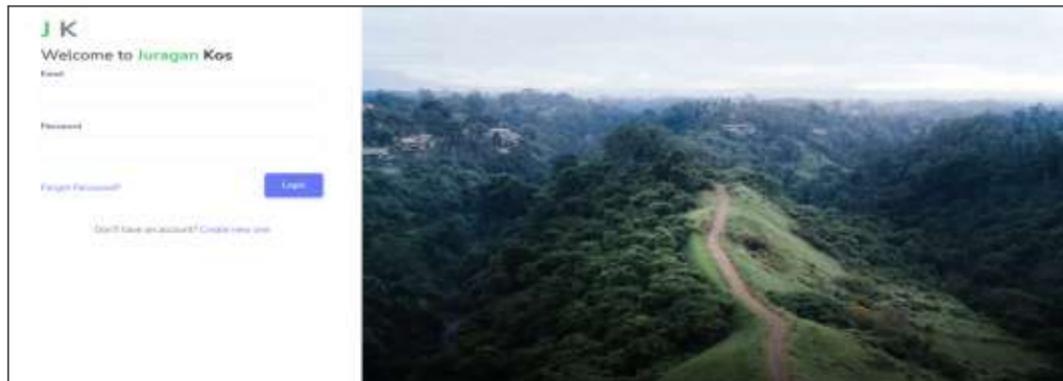
No	Nama Kolom	Tipe Data	Keterangan
4	Gambar	<i>Varchar(60)</i>	Gambar konten dari Blog
5	Kategori	<i>Varchar(60)</i>	Kategori dari Blog
6	Isi	<i>Longtext</i>	Konten utama Blog
7	<i>Author</i>	<i>Varchar(60)</i>	Penulis Blog
8	<i>Quote</i>	<i>Varchar(100)</i>	<i>Quote</i> dari Blog
9	<i>Quote_author</i>	<i>Varchar(60)</i>	Sumber atau pembuat dari <i>quote</i>
10	<i>Created_at</i>	<i>Timestamp</i>	Tanggal Blog dibuat
11	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal Blog diperbarui

F. Implementasi *User Interface*

Implementasi adalah tahap penerapan sekaligus pengujian bagi sistem berdasarkan hasil analisa dan perancangan sistem, pada tahap ini merupakan tahap implementasi hasil menjadi sebuah Sistem Informasi Pencarian kos-kosan di Kecamatan Tampan.

1. Halaman *Login*

Login merupakan tampilan untuk melanjutkan penggunaan sistem seperti mengakses *Dashboard* ataupun melakukan *Booking* kamar kos, pada saat melakukan *login username* dan *Password* yang dimasukan harus sesuai dengan *value* yang ada pada *Database*, berikut tampilan halaman *login*, seperti gambar 4.4.



Gambar 4.4 Halaman *Login*

2. Halaman *Register*

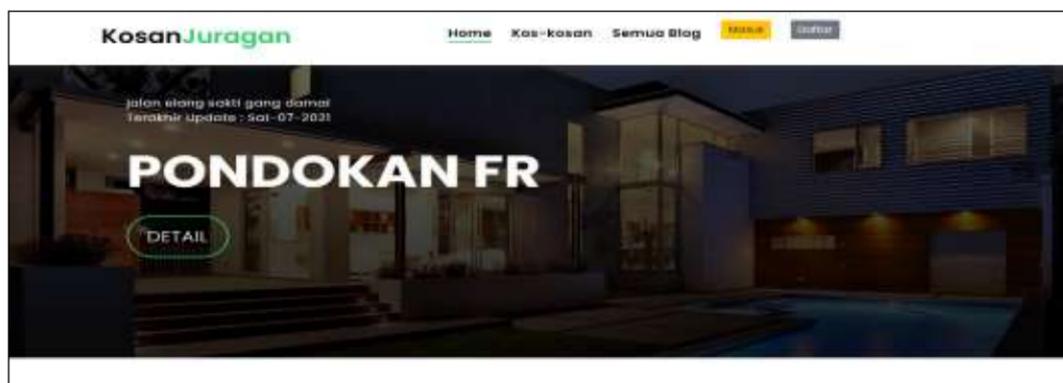
Merupakan halaman untuk melakukan pendaftaran pada *website*. pada *form registrasi* harus mengisi semua *form registrasi* yang tersedia sesuai dengan data pribadi, pada saat melakukan pendaftaran pengunjung *website* dapat memilih hak akses seperti pemilik kos ataupun pencari kos, berikut tampilan halaman *Register* seperti gambar 4.5.

The image shows a registration form titled 'JURAGAN KOS' and 'Register'. The form is set against a light blue background. It contains several input fields: 'Name', 'Email', 'Member Sebagai' (with a dropdown arrow), 'Username', 'Password', and 'Password Confirmation'. At the bottom left, there is a checkbox labeled 'I agree with the terms and conditions'. A prominent blue 'Register' button is located at the bottom center of the form.

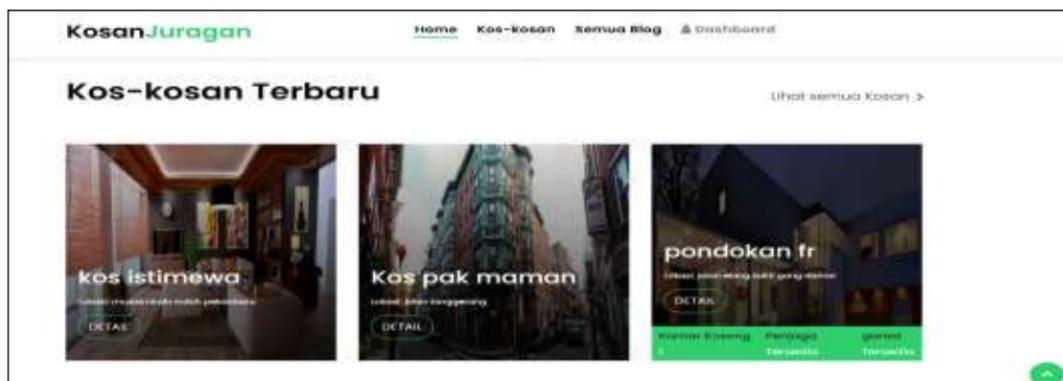
Gambar 4.5 *Form Register*

3. Halaman Utama

Halaman utama adalah tempat penyampaian informasi Kos-kosan terbaru, pada halaman utama ditampilkan 6 kos-kosan yang baru mendaftar yang sebelumnya telah di verifikasi oleh admin untuk ditampilkan pada halaman utama *website* seperti gambar 4.6, halaman Kos-kosan terbaru 4.7



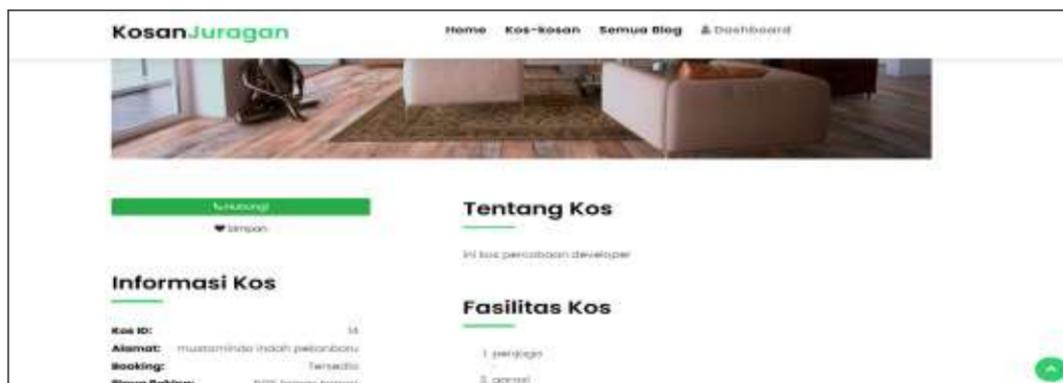
Gambar 4.6 Halaman Utama



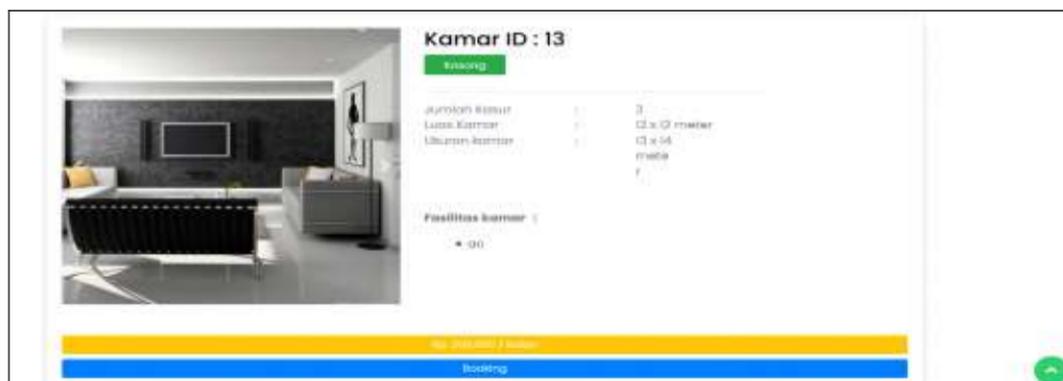
Gambar 4.7 Halaman Utama Kos-kosan Terbaru

4. Halaman Detail Kos

Pada halaman Detail kos ditampilkan detail-detail tentang kos seperti gambar 4.8, informasi kamar kos seperti gambar 4.9, dan testimonial kos seperti gambar 4.10.



Gambar 4.8 Tampilan Detail Ko



Gambar 4.9 Tampilan Detail Kamar-kamar Kos Tersedia



Gambar 4.10 Section Testimonial Kos Pada Halaman Detail

5. Halaman *Form Booking Kamar Kos*

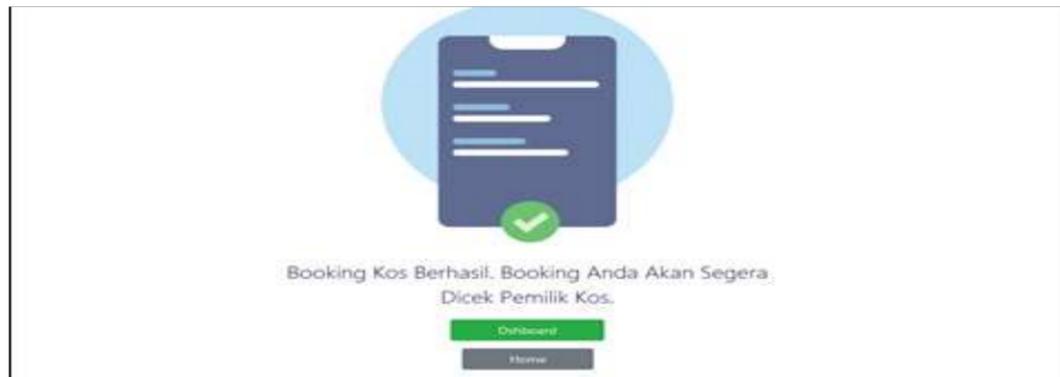
Pada halaman *Booking* kamar kos yang sebelumnya telah dipilih oleh pengguna, tersedia *form Booking* kamar yang harus diisi oleh pengunjung seperti gambar 4.11.



Gambar 4.11 *Form Booking Kamar Kos*

6. Halaman Sukses *Booking*

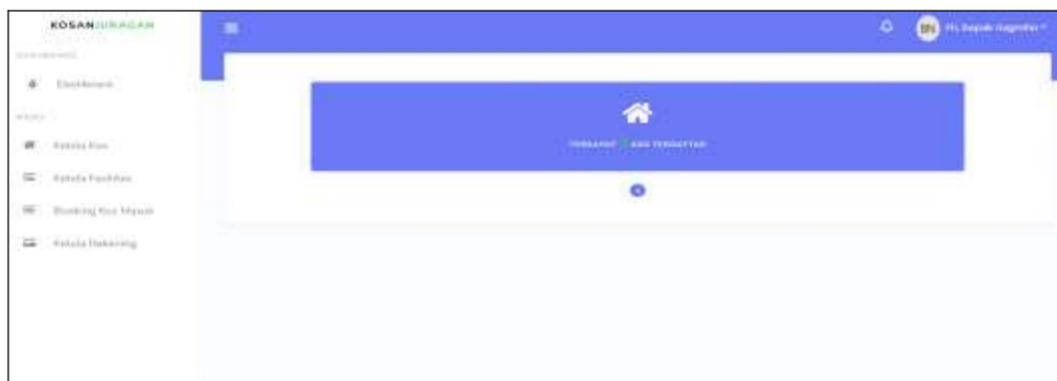
Pada halaman ini akan ditampilkan *illustration* bahwa permintaan *Booking* kamar telah dikirimkan kepada pemilik kos, serta terdapat dua tombol, tombol *home* untuk kembali ke halaman utama dan tombol *Dashboard* untuk masuk ke halaman *Dashboard* seperti gambar 4.12.



Gambar 4.12 Halaman Sukses *Booking*

7. Halaman *Dashboard* Pemilik Kos

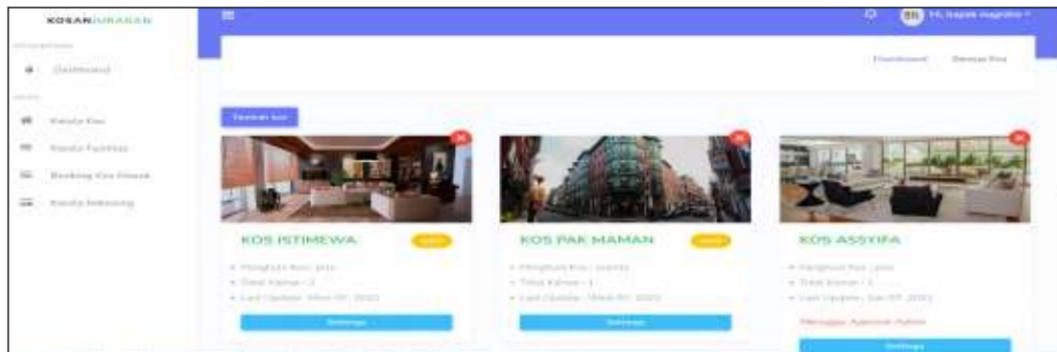
Pada halaman *Dashboard* pemilik ditampilkan *menu-menu* atau informasi yang dikhususkan untuk *role* pemilik kos seperti gambar 4.13.



Gambar 4.13 Halaman Utama *Dashboard*

a. Halaman *Dashboard* Kelola Kos

Pada halaman *Dashboard* kelola kos pemilik kos dapat melakukan tambah kos, hapus kos serta menampilkan semua kos-kosan yang dimiliki pemilik kos seperti gambar 4.14.



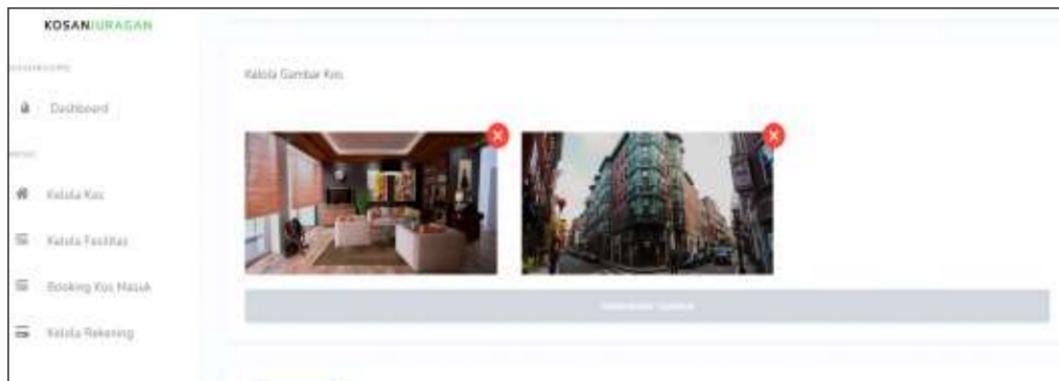
Gambar 4.14 Kelola Kos

b. Halaman Detail Kos

Pada halaman Detail kos terdapat *form* yang berfungsi

Gambar 4. 15 Tampilan *Form* Edit Kos

Untuk memperbarui kos seperti gambar 4.15, tombol mengelola gambar kos seperti 4.16 serta terdapat tabel yang menampilkan jumlah kamar kos seperti gambar 4.17.



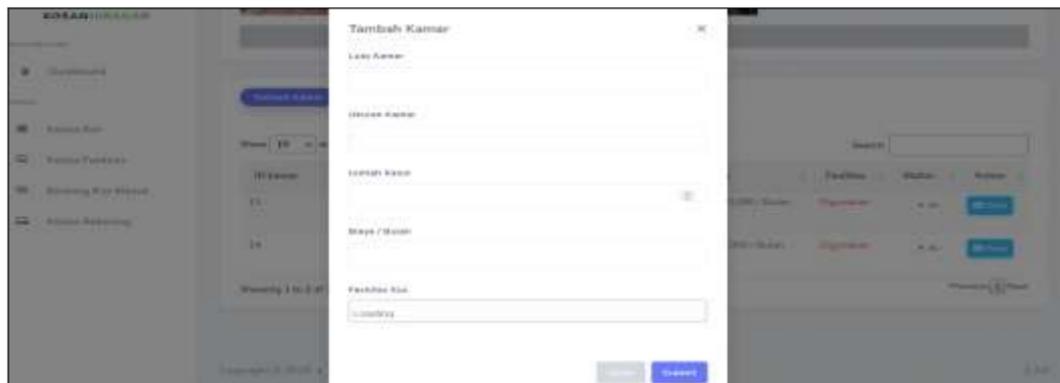
Gambar 4.16 Tampilan Kelola Gambar Kos

ID Kamar	Lantai Kamar	Ukuran Kamar	Jml Kamar	Biaya	Fasilitas	Status	Aksi
14	12 x 12 kamar standar	1.5 x 1.4 kamar standar	0	Rp. 200.000 /bulan	Diperlukan	✖	+1 Tambah Kamar
14	12 x 12 kamar standar	1.7 x 1.4 kamar standar	0	Rp. 30.000 /bulan	Diperlukan	✖	+1 Tambah Kamar

Gambar 4.17 Table Kamar Pada Kos

c. Tampilan *Form* Tambah Kamar

Pada gambar 4.17 terdapat tombol tambah kamar yang berfungsi untuk memunculkan *Pop-up modal* yang terdapat *form* tambah kamar seperti gambar 4.18, setelah mengisi *form* tambah kamar diarahkan pada halaman tambah kamar seperti gambar 4.19.



Gambar 4.18 *Pop-up Modal Form Tambah Kamar*



Gambar 4.19 *Tambah Gambar Kamar*

d. Tampilan *Booking Kos Masuk*

Pada tampilan *Booking kos masuk* menampilkan semua permintaan *Booking* kamar oleh *member*, dan terdapat empat tombol yaitu tombol terima yang berfungsi untuk menerima permintaan *Booking*. tombol tolak untuk menolak permintaan *Booking*, tombol Detail untuk melihat Detail *Booking* serta tombol *print* seperti gambar 4.20, berikut hasil cetak PDF semua *Booking* yang masuk seperti gambar 4.21.

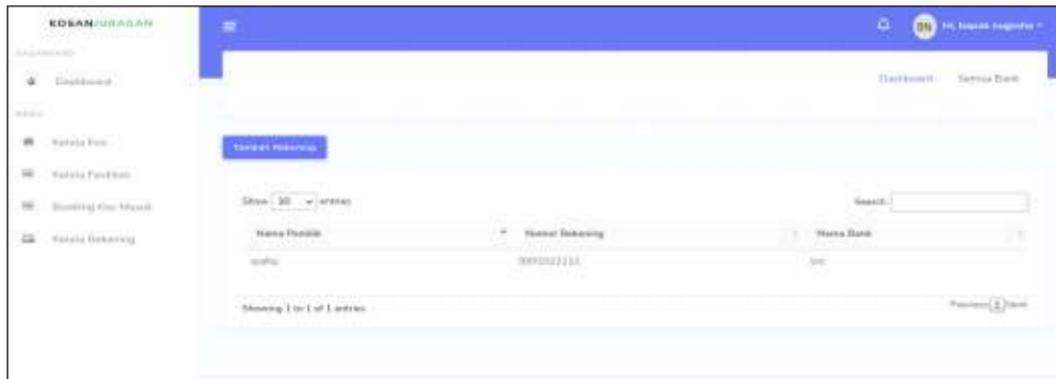
Gambar 4.20 Tampilan *Booking* Masuk

No	Penyewa	Nomor Penyewa	Email Penyewa	Alamat Penyewa	Nama Kos	Kamar ID	Lama Sewa	Tanggal
1	penyewa	0612342221	penyewa@gmail.com	jalan kulit manggis	kos istimewa	13	3 / Bulan	21 / Jul / 2021
2	penyewa	0612342221	penyewa@gmail.com	jalan kulit manggis	Kos pak mamam	15	0 / Bulan	21 / Jul / 2021
3	susanto	06123456098	susanto@gmail.com	jalan sisingerjengeraja	Kos pak mamam	15	1 / Bulan	21 / Jul / 2021

Gambar 4.21 Cetak PDF *Booking* Masuk

e. Tampilan Kelola Rekening

Pada tampilan kelola rekening menampilkan semua rekening yang dimiliki oleh pemilik kos seperti gambar 4.22, serta pemilik kos dapat menambah rekening melalui *form modal* seperti gambar 4.23. serta meg-*update* rekening seperti gambar 4.24.



Gambar 4.22 Tampilan Kelola Rekening

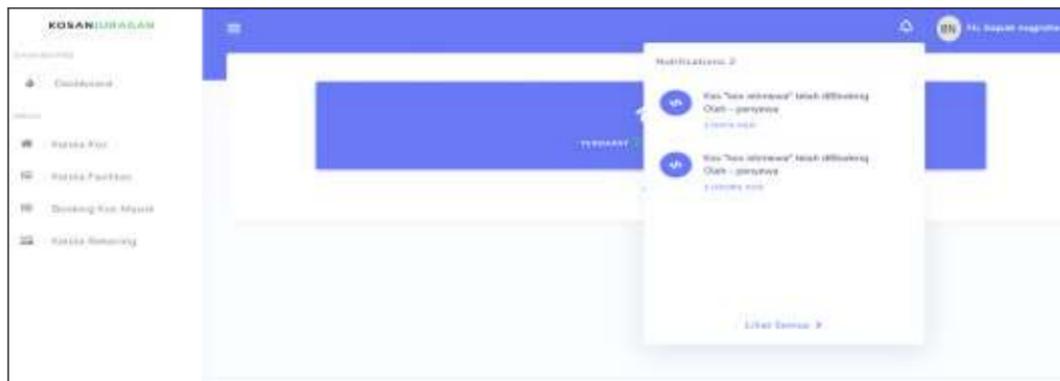


Gambar 4.23 Pop-Up Modal Form Tambah Rekening



Gambar 4.24 Form Memperbarui Rekening

f. Tampilan Notifikasi *Booking* Kos Masuk



Gambar 4.25 Notifikasi *Booking* Masuk

g. Tampilan Halaman *Profile*

Pada tampilan ini menampilkan data diri pemilik kos yang sedang *login* sesuai dengan ID pemilik kos pada tampilan ini pemilik kos dapat melakukan *Setting profile* seperti gambar 4.26 dan ganti *Password* seperti gambar 4.27.



Gambar 4.26 Tampilan Profile Pemilik Kos



Gambar 4.27 Tampilan *Form Ganti Password* Pada Halaman Profile

8. Halaman *Dashboard* Pemilik Kos

Pada halaman *Dashboard* pemilik ditampilkan *menu-menu* atau informasi yang dikhususkan untuk *role* pemilik kos seperti gambar 4.28.



Gambar 4.28 Halaman Utama *Dashboard Member*

a. Halaman *Dashboard Riwayat Booking*

Pada halaman riwayat *Booking* ditampilkan semua riwayat *Booking* yang pernah dilakukan penyewa kos, pada halaman ini juga disediakan informasi tentang periode jadwal kos periode jadwal kos, terdapat juga tombol print untuk cetak PDF bukti atau riwayat *Booking*.

Aksi	Nama Kos	ID Kamar	Periode Sewa	Lama Sewa	Status Pembayaran	Status Booking
	kos istimewa	13	01 JAN 2021 - 03 JAN 2021	3 Bulan	Melunggu Approve Pihak Kos	Booking Ditolak
	kos istimewa	13	01 JAN 2021 - 03 JAN 2021	3 Bulan	Maka Kos Anda Masih Ditunggu	Booking Diterima
	kos istimewa	13	01 JAN 2021 - 03 JAN 2021	3 Bulan	Maka Kos Anda Masih Ditunggu	Booking Diterima
	kos istimewa	13	01 JAN 2021 - 03 JAN 2021	3 Bulan	Melunggu Approve Pihak Kos	Booking Ditolak

Gambar 4.29 Halaman Riwayat *Booking*

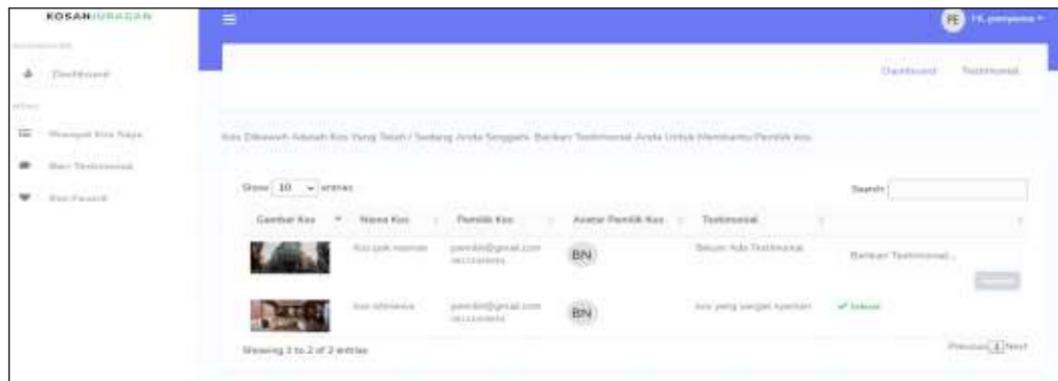
b. hasil cetak PDF riwayat *Booking*

JURAGAN KOS							
Booking kos - kos istimewa #11							
Nama	Periode Sewa	Lama / bulan	Biaya DP	Nama kos	Kamar ID	Status Booking	Tanggal
penyewa	01 / Jan / 1970 - 01 / Jan / 1970	3	300,000	kos istimewa	13	diterima	21 / Jul / 2021

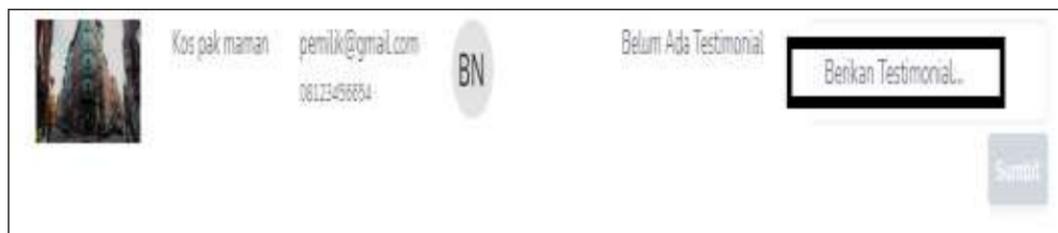
Gambar 4.30 Cetak PDF *Booking Member*

c. Halaman *Dashboard Testimonial Member*

Pada halaman testimonial ditampilkan semua kos yang pernah disinggahi oleh *Member* seperti gambar 4.31, dan juga tersedia *form* testimonial untuk Memberikan testimonial pada kos seperti gambar 4.32.



Gambar 4.31 Halaman *Dashboard* Testimonial



Gambar 4.32 *Form* Testimonial Terhadap Kos

d. Halaman *Dashboard* Kos Tersimpan

Pada halaman *Dashboard* kos tersimpan ditampilkan semua kos yang pernah disimpan oleh *Member* agar bisa dikunjungi kembali Detail kos tersebut seperti gambar 4.33.



Gambar 4.33 Halaman *Dashboard* Kos Tersimpan

e. Halaman *Dashboard* Profile Member

Pada tampilan ini menampilkan data diri *Member* yang sedang *login* sesuai dengan ID *Member*, pada tampilan ini *Member* dapat melakukan *Setting profile* seperti gambar 4.34 dan ganti *Password* seperti gambar 4.35.



Gambar 4.34 Halaman *Dashboard Profile Member*



Gambar 4.35 Halaman *Dashboard Profile Ganti Password*

9. Halaman *Dashboard Admin*

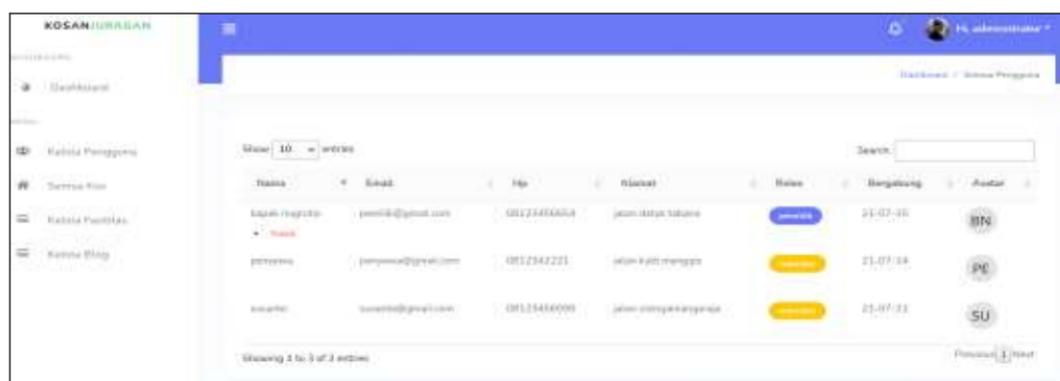
Pada halaman *Dashboard* admin ditampilkan *menu-menu* atau informasi yang dikhususkan untuk *role admin* seperti gambar 4.36.



Gambar 4.36 Halaman *Dashboard* Utama Admin

a. **Halaman *Dashboard* Kelola User**

Pada halaman *Dashboard* kelola *user* menampilkan semua pengguna *website* yang terdaftar baik pemilik kos maupun *Member*, admin dapat menghapus pengguna pada halaman ini seperti gambar 4.37.

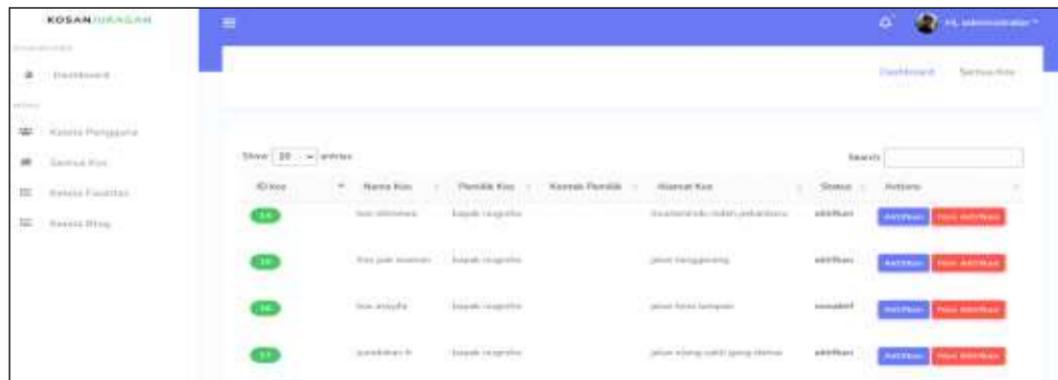


Gambar 4.37 Halaman *Dashboard* Kelola User

b. **Halaman *Dashboard* Admin Kelola Semua Kos**

Pada halaman ini menampilkan semua Kos-kosan yang terdaftar, serta terdapat dua tombol yaitu tombol aktifkan yang berfungsi untuk mengaktifkan Kos-kosan agar bisa tampil pada

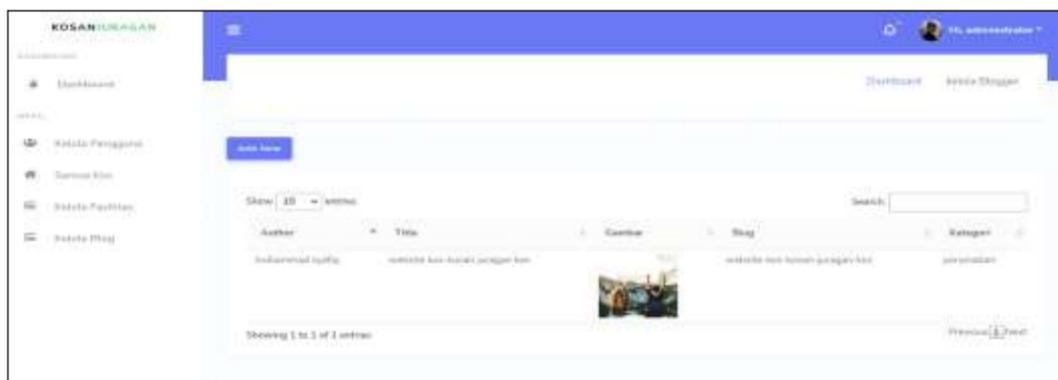
halaman *website* serta tombol nonaktifkan berfungsi untuk nonaktifkan *website* agar tidak tampil di *website* seperti gambar.4.38.



Gambar 4.38 Halaman *Dashboard* Admin Semua Kos

c. Halaman *Dashboard* Admin Kelola Blog

Pada halaman ini ditampilkan semua blog yang pernah dibuat oleh admin, *serta* admin juga dapat mengelola semua blog seperti membuat, menghapus dan memperbaiki blog seperti gambar 4.39.



Gambar 4.39 Halaman *Dashboard* Admin Kelola Blog

d. *Pop-Up Modal* Membuat Blog

Pada tampilan ini akan muncul sebuah *Pop-Up modal* yang berisi *form* tambah blog baru seperti gambar 4.40.

Gambar 4.40 *Form* Membuat Blog

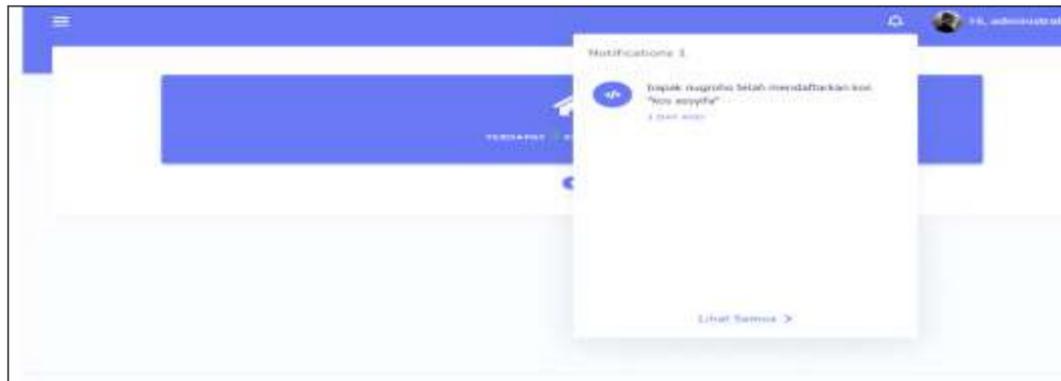
e. Halaman *Dashboard* Admin Perbarui Blog

Pada halaman ini ditampilkan *form* untuk memperbarui blog seperti gambar 4.41.

Gambar 4.41 *Form* perbarui Blog

f. Halaman *Dashboard* Admin Notifikasi Kos Mendaftar

Pada halaman ini menampilkan notifikasi jika ada Kos-kosan yang baru didaftarkan oleh pemilik kos pada *website* seperti gambar 4.42.



Gambar 4.42 Notifikasi Kos-kosan Mendaftar

g. Halaman *Dashboard Profile Admin*

Pada tampilan ini menampilkan data diri admin yang sedang *login* sesuai dengan ID admin, pada tampilan ini *Member* dapat melakukan *Setting profile* seperti gambar 4.43 dan ganti *Password* seperti gambar 4.44.



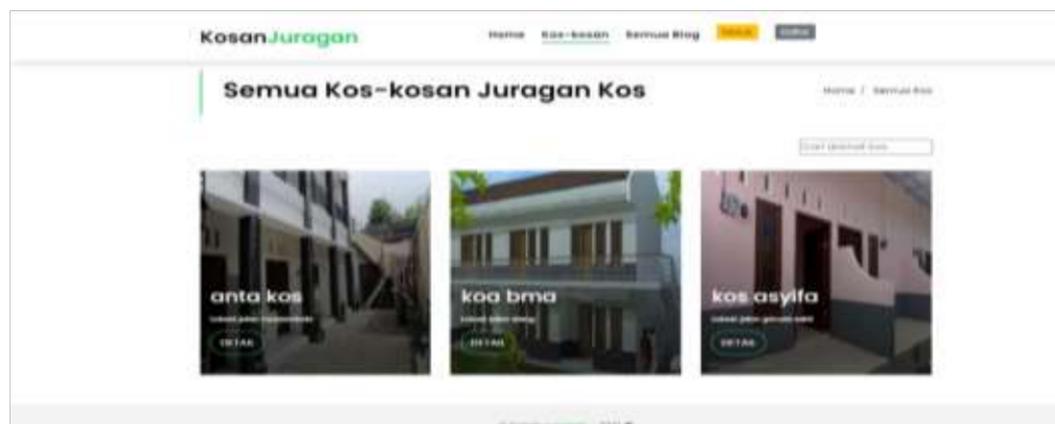
Gambar 4.43 Halaman *Dashboard Admin Setting Profile*



Gambar 4.44 Halaman *Dashboard* Admin Ganti *Password*

h. Halaman Menampilkan Semua Kos-kosan

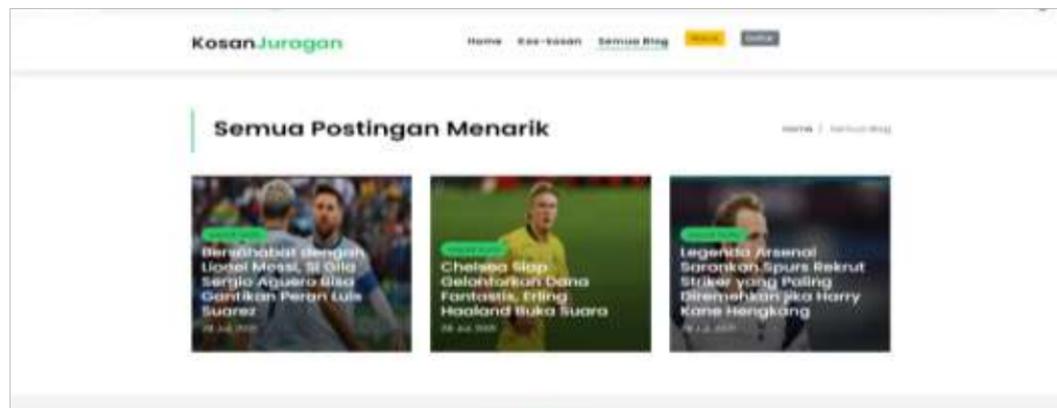
Halaman ini menampilkan semua Kos-kosan yang terdaftar pada *website* seperti gambar 4.45.



Gambar 4.45 Halaman Menampilkan Semua Kos-kosan

i. Halaman Menampilkan Semua Blog

Halaman ini menampilkan semua blog yang terdaftar pada *website* seperti gambar 4.46.



Gambar 4.46 Halaman Menampilkan Semua Blog

G. Pengujian Perangkat Lunak

Pada tahap ini akan melakukan pengujian sistem yang bertujuan untuk menemukan kesalahan-kesalahan pada perangkat lunak yang akan diuji. Pengujian bermaksud untuk mengetahui apakah perangkat lunak sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut. Pengujian perangkat lunak ini menggunakan *black box*. Pengujian *black box* berfokus pada fungsi-fungsi perangkat lunak tanpa menguji desain dari program. Berikut adalah hasil pengujian dengan *black box* yang dilakukan oleh 10 orang pengguna sebagai berikut:

Tabel 4. 20 Pengujian *Black box*

Activitas pengujian	Realisasi yang diharapkan	Hasil pengujian	Kesimpulan
Masuk halaman awal	Menampilkan 6 Kos-kosan terbaru	Halaman awal menampilkan 6 Kos-kosan terbaru	Berhasil
Halaman Login	Menampilkan 2 <i>input text field</i> dan 1 <i>button</i>	Menampilkan 2 <i>input text field</i> dan 1 <i>button</i> pada tampilan Berhasil Login	Berhasil
Halaman Register	Menampilkan 7 <i>input text field</i> dan 1 <i>button</i>	Menampilkan 7 <i>input text field</i> dan 1 <i>button</i> pada tampilan Register	Berhasil

Activitas pengujian	Realisasi yang diharapkan	Hasil pengujian	Kesimpulan
Halaman Detail kos	Menampilkan 2 tombol, menampilkan informasi kos, fasilitas kos dan informasi kamar-kamar kos	Menampilkan 2 tombol, menampilkan informasi kos, fasilitas kos dan informasi kamar-kamar kos pada halaman Detail kos	Berhasil
Halaman <i>Booking</i> kos	Menampilkan gambar kamar kos, dan menampilkan 5 <i>input text</i> dan terdapat 1 tombol untuk <i>Booking</i> kamar kos	Menampilkan gambar kamar kos, dan menampilkan 5 <i>input text</i> dan terdapat 1 tombol untuk <i>Booking</i> kamar kos pada halaman <i>Booking</i> kos	Berhasil
Halaman kelola kos	Menampilkan 1 tombol tambah kos, dan Kos-kosan yang telah dibuat	Menampilkan 1 tombol tambah kos dan menampilkan list-list kos pada halaman kelola kos	Berhasil
Halaman tambah kos	Menampilkan 11 <i>input text</i> untuk menambah kos, dan menampilkan maps untuk lokasi kos	Menampilkan 11 <i>input text</i> untuk menambah kos, dan menampilkan maps untuk lokasi kos pada halaman tambah kos	Berhasil
Halaman <i>Setting profile</i>	Menampilkan 6 <i>input text</i> serta menampilkan 2 tombol untuk memperbarui <i>profile</i> dan <i>upload</i> gambar	Menampilkan 6 <i>input text</i> serta menampilkan 2 tombol untuk memperbarui <i>profile</i> dan <i>upload</i> gambar pada halaman <i>Setting profile</i> .	Berhasil
Halaman ganti <i>Password</i>	Menampilkan 2 <i>input text</i> serta menampilkan 1 tombol untuk memperbarui <i>Password</i>	Menampilkan 2 <i>input text</i> serta menampilkan 1 tombol untuk memperbarui <i>Password</i> pada halaman ganti <i>Password</i> .	Berhasil
Halaman semua kos	Menampilkan semua Kos-kosan yang terdaftar serta terdapat 1 <i>input text</i> untuk pencarian kos	Menampilkan semua Kos-kosan yang terdaftar serta terdapat 1 <i>input text</i> untuk pencarian kos pada halaman semua kos	Berhasil

Activitas pengujian	Realisasi yang diharapkan	Hasil pengujian	Kesimpulan
Halaman <i>browse</i> kos	Menampilkan pemetaan seluruh lokasi Kos-kosan yang terdaftar	Menampilkan pemetaan seluruh lokasi Kos-kosan yang terdaftar pada halaman <i>browse</i> kos	Berhasil
Halaman semua Blog	Menampilkan semua Blog yang ditambahkan oleh admin	Menampilkan semua Blog yang ditambahkan oleh admin pada halaman semua Blog	Berhasil

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil yang didapatkan dari serangkaian proses penelitian mengenai Sistem Informasi Pencarian kos-kosan di Kecamatan Tampan Berbasis Web, maka dapat disimpulkan bahwa :

1. *Website* ini menyediakan *platform* bagi pemilik kos untuk mempromosikan dan memasarkan kosnya.
2. Dapat mempermudah pemilik Kos-kosan dalam melakukan proses promosi rumah kos miliknya tanpa harus datang ke admin lokal.
3. *Website* ini juga dapat memudahkan orang yang ingin melakukan *Booking* kamar kos dengan cara melakukan pendaftaran pada *website* melalui halaman registrasi.

B. Saran

Saran untuk penelitian selanjutnya adalah sebagai berikut :

1. Untuk mempermudah komunikasi antara pencari kos dan pemilik kos, sistem ini dapat dikembangkan kembali dengan membuat *fitur live chat*.
2. Untuk memberikan kemudahan dalam proses pencarian dan pemesanan rumah kos, sistem ini dapat dikembangkan kembali dengan membuat sistem informasi berbasis android.

3. Untuk mempermudah alur transaksi penyewaan kamar kos, sistem ini dapat dikembangkan lagi dengan membuat fitur *payment gateway*.

DAFTAR PUSTAKA

- Abdul Rozaq, dkk. (2015). “Sistem Informasi Produk Dan Data Calon Jamaah Haji Dan Umroh Pada PT. Travellindo Lusiyan Banjarmasin Berbasis Web.” *Laporan Penelitian. Politeknik Negeri Banjarmasin.*
- Agustini, Wahyu Joni Kurniawan. (2019). “Sistem E-Learning Do’a dan Iqro’ dalam Peningkatan Proses Pembelajaran pada TK Amal Ikhlas.” *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi.*
- Ariefah Rachmawati (2017). “Membangun Informasi Layanan Umum Rumah Kos Melalui Aplikasi Berbasis Web.” *Jurnal Ilmiah Fifo (Nomor 2 Tahun 2017).*
- Firman, A., Wowor, H. F., Najoran, X., Teknik, J., Fakultas, E., & Unsrat, T. (2016). Sistem Informasi Perpustakaan Online Berbasis Web. *E-Journal Teknik Elektro Dan Komputer*, 5(2), 29–36.
- Firman, A., Wowor, H. F., Najoran, X., Teknik, J., Fakultas, E., & Unsrat, T.
- Fitri Ayu dan Nia Permatasari. (2018). “Perancangan Sistem Informasi Pengolahan Data Praktek Kerja Lapangan (Pkl) Pada Devisi Humas Pt. Pegadaian.” *Jurnal Intra-Tech.*
- Haryana, K. S. (2008). Pengembangan Perangkat Lunak Dengan Menggunakan Php. *Jurnal Computech & Bisnis*, 2(1), 14–21. <http://jurnal.stmik-mi.ac.id/index.php/jcb/article/view/74>
- Hasyim Hasanah (2016), “Teknik-Teknik Observasi”. *Jurnal at-Taquaddum* (volume 8, Nomor 1, Juli 2016).
- Ii, B. A. B., & Teori, L. (2019). *Jurnal Perancangan Informasi*. 5–14.
- Ipan Ripai (2017). “Rancang Bangun Media Pembelajaran Menggunakan Android Untuk Matakuliah Pemrograman Internet Menggunakan Magazine App Maker”. *Jurnal ICT Learning* (Vol. 3 No. 1 Mei 2017).
- Irwan, A. (2013). *Framework Laravel Untuk Informasi Penunjang Perkuliahan*. 53(9), 1689–1699.
- Karsono, K. (2010). *PEMBANGUNAN APLIKASI PENYEWAAN APARTEMEN BERBASIS N-TIER DENGAN MOBILE DEVICE.*
- Kuliah, M., & Ii, P. (2007). *Mata kuliah psikodiagnostik ii (observasi).*
- Lila Setiyani (2019). *Rekayasa Perangkat Lunak. Sistem Informasi Manajemen Akreditasi Balai Pelatihan Kesehatan*

- Lukman Nul Hakim (2013). "Ulasan Metodologi Kualitatif: Wawancara Terhadap Elit". *Lukman Nul Hakim, Ulasan Metodologi Kualitatif* (Aspirasi Vol. 4No. 2, Desember 2013).
- Organisasi, P., Dunia, K., Tenggara, A., Tenggara, A., Indonesia, D., Saluran, I., & Akut, P. (2010). *Tinjauan Pustaka. Djojoningrat 2006*, 4–20.
- Pawit M. Yusup, dkk. (2019). "Pemanfaatan Internet Untuk Penghidupan Di Kalangan Pemuda Pedesaan. "*Laporan Penelitian. Universitas Padjadjaran.*
- Permadina Kanah Arieska dan Novera Herdian (2018), "Pemilihan Teknik Sampling Berdasarkan Perhitungan Efisiensi Relatif". *Statistika* (Vol. 6, No. 2, November 2018).
- Pradipta, A. A., Prasetyo, Y. A., & Ambarsari, N. (2015). Pengembangan Web E-Commerce Bojana Sari Menggunakan Metode *Prototype*. *EProceedings of Engineering*, 2(1), 1042–1056.
- Randi V. Palit, dkk. (2015). "Rancangan Sistem Informasi Keuangan Gereja Berbasis Web Di Jemaat GMIM Bukit Moria Malalayang. "*Laporan Penelitian. UNSRAT.*
- Soegijono, M. (1993). Wawancara Sebagai Salah Satu Metode Pengumpulan Data. *Media Penelitian Dan Pengembangan Kesehatan*, 3(1), 17–21. <https://doi.org/10.22435/mpk.v3i1.930>.
- Sutiyono dan Santi (2020). "Membangun Sistem Informasi Pendaftaran Siswa Baru Berbasis Web Dengan Metode Mdd (Model Driven Development) Di Raudhatul Athfal Nahjussalam". *Jurnal Sistem Informasi, J-SIKA* (Volume 02 Nomor 01, Juli 2020: 50-56).
- Triyono L, (2016). Sistem Informasi Akademik Kampus Berbasis Web dengan Laravel 5. Yogyakarta. Lokomedia.