



INSTITUT TEKNOLOGI DEL

Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android

DOKUMEN TUGAS AKHIR

Oleh:

11317004 Ronatiur Febriani Lumban Gaol
11317040 Feny Bertarida. M.S. Simanjuntak
11317041 Chorintians Lucky Panjaitan

FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO
PROGRAM STUDI DIII TEKNOLOGI INFORMASI
LAGUBOTI
JULI 2020



INSTITUT TEKNOLOGI DEL

Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android

DOKUMEN TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
A.md.Kom
Program Studi Teknologi Informasi**

Oleh :

**11317004 Ronatiur Febriani Lumban Gaol
11317040 Feny Bertarida. M.S. Simanjuntak
11317041 Chorintians Lucky Panjaitan**

**FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO
PROGRAM STUDI DIII TEKNOLOGI INFORMASI**

**LAGUBOTI
JULI 2020**

HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Ronatiur Febriani Lumban Gaol

NIM : 11317004

Tanda Tangan :



Tanggal : 29 Juli 2020

Nama : Feny Bertarida. M.S. Simanjuntak

NIM : 11317040

Tanda Tangan :



Tanggal : 29 Juli 2020

Nama : Chorintians Lucky Panjaitan

NIM : 11317041

Tanda Tangan :



Tanggal : 29 Juli 2020

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh : :

1. Nama : Ronatiur Febriani Lumban Gaol
NIM : 11317004
Program Studi : DIII Teknologi Informasi

2. Nama : Feny Bertarida Simanjuntak
NIM : 11317040
Program Studi : DIII Teknologi Informasi

3. Nama : Chorintians Lucky Panjaitan
NIM : 11317041
Program Studi : DIII Teknologi Informasi

Judul Tugas Akhir : Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android

Telah berhasil dipertahankan di hadapannya dewan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Diploma III program studi Diploma III Teknologi Informasi Fakultas Informatika dan Teknik Elektro Institut Teknologi Del.

DEWAN PENGUJI

Pembimbing : Teamsar Muliadi Panggabean, S.Kom, PGCert ()

Penguji : Roy Deddy Hasiholan Lumban Tobing, S.T., M.ICT ()

Penguji : Togu Novriansyah Turnip, S.S.T., M.IM ()

Ditetapkan di : Laguboti
Tanggal : 29 Juli 2020

KATA PENGANTAR

Puji dan syukur kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul "**Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android**" dengan baik.

Adapun maksud dan tujuan dari penulisan Tugas Akhir ini adalah untuk memenuhi persyaratan kelulusan Program Studi Diploma III pada jurusan Teknologi Informasi Fakultas Informatika dan Teknik Elektro di Institut Teknologi Del.

Penyusunan Tugas Akhir ini tidak lepas dari bantuan berbagai pihak, maka dalam kesempatan ini penulis menyampaikan ucapan terimakasih kepada:

1. Kedua orang tua serta keluarga dari ketiga penulis Tugas Akhir ini, karena senantiasa memberi dukungan dan doa untuk mendukung penulis dalam menulis Tugas Akhir ini.
2. Bapak Teamsar Muliadi Panggabean, S.Kom, PGCert sebagai dosen pembimbing yang telah bersedia untuk meluangkan waktu untuk membimbing, memberikan masukan serta arahan selama pengerjaan Tugas Akhir ini.
3. Bapak Roy Deddy Hasiholan Lumban Tobing, S.T., M.ICT dan Bapak Togu Novriansyah Turnip, S.S.T., M.IM selaku dosen penguji Tugas Akhir yang telah memberikan banyak saran yang membangun kepada penulis dalam menyelesaikan Tugas Akhir ini.
4. Seluruh dosen dan staf Institut Teknologi Del yang telah memberikan materi perkuliahan yang dapat membantu dalam menyelesaikan Tugas Akhir, beserta pihak yang tidak dapat disebutkan satu per satu yang telah memberi dukungan, masukan dan ikut membantu dalam proses pengerjaan Tugas Akhir ini.

Penulis menyadari bahwa dalam menyusun laporan ini masih menemui banyak kesulitan, laporan ini juga masih memiliki banyak kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari semua pihak untuk dapat membuat laporan ini menjadi lebih baik lagi. Akhir kata, penulis mengucapkan terimakasih dan semoga penulisan laporan Tugas Akhir ini dapat bermanfaat bagi semua pihak yang membutuhkan.

Sitoluama, 29 Juli 2020

Penulis (D3TI-15)

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Institut Teknologi Del, saya yang bertanda tangan di bawah ini:

Nama : Ronatiur Febriani Lumban Gaol
NIM : 11317004
Fakultas/Program Studi : FITE / DIII Teknologi Informasi
Nama : Feny Bertarida Simanjuntak
NIM : 11317040
Fakultas/Program Studi : FITE/ DIII Teknologi Informasi
Nama : Chorintians Lucky Panjaitan
NIM : 11317041
Fakultas/Program Studi : FITE/ DIII Teknologi Informasi
Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Institut Teknologi Del Hak Bebas Royalti Non Eksklusif (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul: **“Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android”**.

Dengan Hak Bebas Royalty Noneksklusif ini Institut Teknologi Del berhak menyimpan, mengalih/media-format dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik hak cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Laguboti

Pada tanggal :

Yang menyatakan

(Ronatiur F Lumban Gaol)

(Feny B.M.S.Simanjuntak)

(Chorintians L Panjaitan)

ABSTRAK

| | | |
|---------------|---|--|
| Nama | : | Ronatiur Febriani Lumban Gaol |
| Program Studi | : | DIII Teknologi Informasi |
| | | |
| Nama | : | Feny Bertarida. M. S. Simanjuntak |
| Program Studi | : | DIII Teknologi Informasi |
| | | |
| Nama | : | Chorintians Lucky Panjaitan |
| Program Studi | : | DIII Teknologi Informasi |
| | | |
| Judul | : | Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android. |

Toba Samosir merupakan salah satu tempat yang termasuk destinasi wisata di Indonesia. Saat ini telah banyak objek wisata yang dapat dikunjungi di Toba Samosir. Banyaknya objek wisata menyebabkan pengunjung wisata semakin banyak pula. Hal ini menjadi sebuah keuntungan bagi masyarakat daerah Toba Samosir dimana masyarakat dapat memanfaatkan pengunjung wisata sebagai potensi menambah penghasilan. Bagi masyarakat yang memiliki sumber daya yang lebih, dapat menyewakan rumahnya sebagai homestay bagi para pengunjung wisata. Namun, terkadang dalam pencarian *homestay* sulit bagi pengunjung untuk dapat menemukan *homestay* yang sesuai dengan kebutuhan mereka. Hal ini menyebabkan diperlukannya sebuah aplikasi yang menampung informasi *homestay* Toba Samosir. Aplikasi ini akan menampilkan sejumlah 74 *homestay* yang terdaftar pada Dinas Pariwisata Toba Samosir. Aplikasi ini akan memberikan layanan pemesanan bagi pengunjung dan akan menampilkan informasi seputar jumlah kamar serta fasilitas / layanan yang disediakan setiap *homestay*. Sedangkan bagi pemilik *homestay*, aplikasi ini akan membantu mereka dalam mempromosikan *homestay* mereka masing-masing, sehingga dengan aplikasi ini diharapkan mampu meningkatkan pendapatan bagi pemilik *homestay*.

Kata Kunci:

Android, *Homestay*, Toba Samosir

ABSTRACT

| | | |
|---------------|---|--|
| Name | : | Ronatiur Febriani Lumban Gaol |
| Study Program | : | DIII Teknologi Informasi |
| Name | : | Feny Bertarida. M. S. Simanjuntak |
| Study Program | : | DIII Teknologi Informasi |
| Name | : | Chorintians Lucky Panjaitan |
| Study Program | : | D II Teknologi Informasi |
| Title | : | Rancang Bangun Aplikasi Homestay Daerah Toba Samosir Berbasis Android. |

Toba Samosir is one of the places including tourist destinations in Indonesia. Currently there are many attractions that can be visited in Toba Samosir. The number of attractions causes more and more tourist visitors. This has become an advantage for the people of the Toba Samosir area where people can take advantage of tourism visitors as a potential to increase income. For people who have more resources, they can rent their house as a homestay for tourist visitors. However, sometimes in the search for homestays it is difficult for visitors to be able to find a homestay that suits their needs. This causes the need for an application that holds Toba Samosir homestay information. This application will feature 74 homestays registered with the Toba Samosir Tourism Office. This application will provide booking services for visitors and will display information about the number of rooms and facilities / services provided by each homestay. Whereas for homestay owners, this application will assist them in promoting their respective homestays, so that this application is expected to be able to increase income for homestay owners.

Key Words:

Android, Homestay, Toba Samosir

DAFTAR ISI

| | |
|--|-------------|
| HALAMAN PENGESAHAN | ii |
| KATA PENGANTAR..... | iii |
| ABSTRAK..... | vi |
| DAFTAR ISI | viii |
| DAFTAR TABEL | xi |
| DAFTAR GAMBAR..... | xiv |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 3 |
| 1.3 Tujuan | 3 |
| 1.4 Lingkup | 3 |
| 1.5 Pendekatan | 4 |
| 1.5.1 Objek Perancangan Aplikasi..... | 4 |
| 1.5.2 Jenis Data | 4 |
| 1.5.3 Sumber Data..... | 4 |
| 1.5.4 Metode Pengumpulan Data..... | 4 |
| 1.5.5 Metodologi Pengembangan Sistem | 5 |
| 1.6 Sistematika Penyajian..... | 7 |
| BAB II TINJAUAN PUSTAKA | 8 |
| 2.1 Pariwisata..... | 8 |
| 2.2 <i>Homestay</i> | 9 |
| 2.2.1 Pengertian <i>Homestay</i> | 9 |
| 2.2.2 Kriteria dan Persyaratan <i>Homestay</i> | 10 |
| 2.3 Android | 13 |
| 2.4 Mapbox API..... | 14 |
| 2.5 Android Kotlin | 14 |
| 2.6 Web Service | 15 |
| 2.7 Java Development Kit (JDK) | 15 |
| 2.8 Hyper Text Transfer Protocol (HTTP) | 15 |
| 2.9 PHP..... | 16 |
| 2.10 Database..... | 16 |

| | | |
|---|---|------------|
| 2.11 | Design Pattern MVP (Model View Presenter)..... | 17 |
| 2.12 | Representational State Transfer (REST) | 17 |
| 2.13 | Metode <i>Prototype</i> | 18 |
| 2.14 | Latitude dan Longitude..... | 19 |
| BAB III ANALISIS..... | | 20 |
| 3.1. | Metode Pengumpulan Data..... | 20 |
| 3.2 | Metode Pengembangan Sistem | 20 |
| 3.2.1 | Analisis dan Penentuan Kebutuhan | 20 |
| BAB IV DESIGN | | 54 |
| 4.1 | Design interface Low Fidelity..... | 54 |
| 4.1.1 | Design interface Admin | 54 |
| 4.1.2 | Design interface Pengunjung | 61 |
| 4.1.3 | User Interface Pemilik <i>Homestay</i> | 68 |
| 4.2 | Design Prototype High Fidelity | 71 |
| 4.2.1 | Design Prototype <i>Admin</i> | 71 |
| 4.2.2 | Design Prototype Pengunjung | 77 |
| 4.2.3 | Design Prototype Pemilik <i>Homestay</i> | 86 |
| 4.3 | Evaluasi Prototype..... | 91 |
| BAB V IMPLEMENTASI..... | | 92 |
| 5.1 | Implementasi | 92 |
| 5.1.1 | Tools | 92 |
| 5.1.2 | Data..... | 93 |
| 5.1.3 | Persiapan Pembangunan Aplikasi | 93 |
| 5.1.4 | Perancangan Sistem..... | 94 |
| 5.1.5 | Implementasi Fungsi | 95 |
| 5.1.6 | Implementasi Code | 96 |
| 5.1.7 | Implementasi Web service dengan Springboot java | 106 |
| BAB VI TESTING | | 114 |
| 6.1 | Testing | 114 |
| 6.1.1 | Tujuan Pengujian..... | 114 |
| 6.1.2 | Skenario Pengujian | 114 |
| BAB VII HASIL DAN PEMBAHASAN | | 131 |

| | |
|---|------------|
| 7.1 Hasil Admin | 131 |
| 7.1.1 Hasil Mengelola Data Homestay..... | 131 |
| 7.1.2 Hasil Pendaftaran Homestay | 133 |
| 7.1.3 Hasil Daftar Pengguna..... | 134 |
| 7.1.4 Hasil Pemesanan Homestay | 134 |
| 7.2 Hasil Pengunjung | 137 |
| 7.2.1 Splashscreen Pengunjung | 137 |
| 7.2.2 Login Pengunjung | 137 |
| 7.2.3 Register Pengunjung..... | 137 |
| 7.2.4 Home Pengunjung | 137 |
| 7.2.5 List Homestay | 138 |
| 7.2.6 Detail Homestay | 138 |
| 7.2.7 Tampilan Lokasi..... | 138 |
| 7.2.8 Pemesanan..... | 138 |
| 7.2.9 Rincian Akhir | 139 |
| 7.2.10 Pembayaran | 139 |
| 7.3 Hasil Pemilik Homestay | 140 |
| 7.3.1 Hasil Splashscreen..... | 140 |
| 7.3.2 Hasil Melakukan Login | 140 |
| 7.3.3 Hasil Melakukan Registrasi | 141 |
| 7.3.4 Hasil Mengelola Kamar | 141 |
| BAB VIII KESIMPULAN DAN SARAN..... | 142 |
| 8.1 Kesimpulan | 142 |
| 5.2 Saran | 142 |
| Daftar Pustaka..... | 145 |
| LAMPIRAN | xx |

DAFTAR TABEL

| | |
|---|----|
| Tabel 1 Use Case Scenario <i>Login</i> | 30 |
| Tabel 2 Use Case Scenario Registrasi Akun | 31 |
| Tabel 3 Use Case Scenario Pendaftaran <i>Homestay</i> | 32 |
| Tabel 4 Use Case Scenario Konfirmasi Pemesanan <i>Homestay</i> | 33 |
| Tabel 5 Use Case Scenario Konfirmasi Pendaftaran <i>Homestay</i> | 34 |
| Tabel 6 Use Case Scenario Menambah Data Kamar <i>Homestay</i> | 36 |
| Tabel 7 Use Case Scenario Update Kamar <i>Homestay</i> | 37 |
| Tabel 8 Use Case Scenario Delete Data Kamar <i>Homestay</i> | 38 |
| Tabel 9 Use Case Scenario Cetak Laporan Pemesanan <i>Homestay</i> | 39 |
| Tabel 10 Use Case Scenario Tambah <i>Homestay</i> | 40 |
| Tabel 11 Use case scenario delete <i>homestay</i> | 42 |
| Tabel 12 Use Case Scenario Update <i>homestay</i> | 43 |
| Tabel 13 Use Case Scenario Melihat <i>Homestay</i> | 44 |
| Tabel 14 Use Case Scenario Mencari Homestay..... | 45 |
| Tabel 15 Use Case Scenario Melihat Lokasi Homestay | 46 |
| Tabel 16 Use case scenario melakukan pemesanan | 47 |
| Tabel 17 Tipe Data <i>Homestay</i> | 49 |
| Tabel 18 Tipe data pemesanan | 50 |
| Tabel 19 Type Data Kamar | 50 |
| Tabel 21 Type Data pemilik_homestay | 51 |
| Tabel 22 Type Data Admin | 51 |
| Tabel 23 Type Data Pengunjung | 51 |
| Tabel 24. Tools Developing Environment | 92 |
| Tabel 25 Identifikasi Aktor | 94 |
| Tabel 26 Implementasi Fungsi Web (Admin) | 95 |
| Tabel 27 Implementasi Fungsi Pengunjung | 95 |
| Tabel 28 Implementasi Fungsi Pemilik Homestay | 96 |
| Tabel 29 Implementasi code create homestay..... | 96 |
| Tabel 30 Implementasi code fungsi update | 97 |
| Tabel 31 Implementasi code fungsi view..... | 98 |

| | |
|--|-----|
| Tabel 32 Implementasi Code Fungsi Cetak Data | 99 |
| Tabel 33 Implementasi Code Fungsi Hapus Pengguna | 100 |
| Tabel 34 Implementasi Code Fungsi Konfirmasi Pendaftaran | 101 |
| Tabel 35 Implementasi Code Fungsi Hapus Pendaftaran | 101 |
| Tabel 36 Implementasi Code Fungsi Konfirmasi pemesanan..... | 101 |
| Tabel 37 Implementasi Code Registrasi Akun Pengunjung..... | 102 |
| Tabel 38 Implementasi Code Login Pengunjung | 102 |
| Tabel 39 Implementasi Code Melihat Homestay | 103 |
| Tabel 40 Implementasi Code Melakukan Pemesanan Homestay..... | 103 |
| Tabel 41 Implementasi Code Melakukan Pembayaran Homestay | 104 |
| Tabel 42 Implementasi Code Melihat Lokasi Homestay | 104 |
| Tabel 43 Implementasi Code Mengupdate Data Kamar | 104 |
| Tabel 44 Implementasi Code Melihat Detail Kamar | 105 |
| Tabel 45 Implementasi Code Menambah Kamar | 105 |
| Tabel 46 Implementasi Code Melihat Daftar Kamar..... | 105 |
| Tabel 47 Setup application properties..... | 107 |
| Tabel 48 Model entity | 107 |
| Tabel 49 Repository | 108 |
| Tabel 50 Implementasi service | 108 |
| Tabel 51 Respons..... | 109 |
| Tabel 52 Controller | 110 |
| Tabel 53 Swagger Config..... | 110 |
| Tabel 54 Skenario Pengujian Login..... | 114 |
| Tabel 55 Skenario Pengujian Registrasi Akun | 116 |
| Tabel 56 Skenario Pengujian Pendaftaran Homestay | 117 |
| Tabel 57 Skenario Pengujian Melihat Homestay | 118 |
| Tabel 58 Scenario Pengujian Melihat Data Homestay | 119 |
| Tabel 59 Scenario Pengujian Mengelola Data Homestay..... | 120 |
| Tabel 60 Scenario Pengujian Melakukan Pemesanan Homestay | 121 |
| Tabel 61 Scenario Pengujian Konfirmasi Pendaftaran Homestay..... | 122 |
| Tabel 62 Scenario Pengujian Konfirmasi Pemesanan Homestay..... | 124 |
| Tabel 63 Scenario Pengujian Melihat Lokasi Homestay | 125 |

| | |
|--|------|
| Tabel 64 Scenario Pengujian Mencari Homestay..... | 126 |
| Tabel 65 Scenario Pengujian Mengelola Data Kamar | 127 |
| Tabel 66 Scenario Pengujian Mengelola Laporan Pemesanan Homestay | 128 |
| Tabel 67 Scenario Pengujian Melakukan Pembayaran..... | 129 |
| Tabel 68 Data <i>Homestay</i> Tobasa | xxiv |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 1 Arsitektur Android..... | 13 |
| Gambar 2 Metode Prototype | 18 |
| Gambar 3 Bisnis Proses Current system | 21 |
| Gambar 4 Bisnis Proses Melihat <i>Homestay</i> | 22 |
| Gambar 5 Bisnis Proses Pemesanan <i>Homestay</i> | 23 |
| Gambar 6 Bisnis Proses Autentikasi Akun | 23 |
| Gambar 7 Bisnis Proses Pendaftaran <i>Homestay</i> | 24 |
| Gambar 8 Bisnis Proses Mengelola Data Kamar | 25 |
| Gambar 9 Bisnis Proses Konfirmasi Pemesanan <i>Homestay</i> | 25 |
| Gambar 10 Bisnis Proses Mengelola Daftar <i>Homestay</i> | 26 |
| Gambar 11 Bisnis Proses Konfirmasi Pendaftaran <i>Homestay</i> | 26 |
| Gambar 12 Bisnis Proses Mengelola Laporan Pemesanan | 27 |
| Gambar 13 Bisnis Proses Melihat Lokasi | 27 |
| Gambar 14 Bisnis Proses Mencari Homestay | 28 |
| Gambar 15 Use Case Diagram | 29 |
| Gambar 16 ERD | 48 |
| Gambar 17 Conceptual Data Model | 48 |
| Gambar 18 Physical Data Model..... | 49 |
| Gambar 19 <i>Design interface</i> Login | 54 |
| Gambar 20 <i>Design interface</i> Home | 55 |
| Gambar 21 <i>Design interface</i> Tambah <i>Homestay</i> | 56 |
| Gambar 22 <i>Design interface</i> Data <i>Homestay</i> | 56 |
| Gambar 23 <i>Design interface</i> Detail <i>Homestay</i> | 57 |
| Gambar 24 <i>Design interface</i> Edit <i>Homestay</i> | 57 |
| Gambar 25 <i>Design interface</i> Pendaftaran <i>Homestay</i> | 58 |
| Gambar 26 <i>Design interface</i> Detail Pendaftaran <i>Homestay</i> | 58 |
| Gambar 27 <i>Design interface</i> Daftar Pengguna..... | 59 |
| Gambar 28 <i>Design interface</i> Pemesanan | 59 |
| Gambar 29 <i>Design interface</i> Konfirmasi Pemesanan..... | 60 |
| Gambar 30 <i>Design interface</i> Splashscreen..... | 61 |
| Gambar 31 <i>Design interface</i> Login | 61 |

| | |
|---|----|
| Gambar 32 <i>Design interface Register</i> | 62 |
| Gambar 33 <i>Design interface Home</i> | 62 |
| Gambar 34 <i>Design interface List Homestay</i> | 63 |
| Gambar 35 <i>Design interface Detail Homestay</i> | 63 |
| Gambar 36 <i>Design interface Pemesanan</i> | 64 |
| Gambar 37 <i>Design interface Rincian Pemesanan</i> | 64 |
| Gambar 38 <i>Design interface Rincian Akhir</i> | 65 |
| Gambar 39 <i>Design interface Pembayaran</i> | 65 |
| Gambar 40 <i>Design interface Transfer DP</i> | 66 |
| Gambar 41 <i>Design interface Bayar di Tempat</i> | 66 |
| Gambar 42 <i>Design interface Transfer Lunas</i> | 67 |
| Gambar 43 <i>Design interface Splashscreen</i> | 68 |
| Gambar 44 <i>Design interface Login</i> | 68 |
| Gambar 45 <i>Design interface Register</i> | 69 |
| Gambar 46 <i>Design interface Home</i> | 69 |
| Gambar 47 <i>Design interface Tambah Homestay</i> | 69 |
| Gambar 48 <i>Design interface Kelola Homestay</i> | 69 |
| Gambar 49 <i>Design interface Tambah Kamar</i> | 70 |
| Gambar 50 <i>Design interface Informasi Homestay</i> | 70 |
| Gambar 51 <i>Design interface Daftar Kamar</i> | 70 |
| Gambar 52 <i>Design interface Edit Kamar</i> | 70 |
| Gambar 53 <i>Design interface Edit Informasi Homestay</i> | 71 |
| Gambar 54 <i>Design interface Beranda</i> | 71 |
| Gambar 55 <i>Design interface Daftar Homestay</i> | 72 |
| Gambar 56 <i>Design interface Tambah Homestay</i> | 72 |
| Gambar 57 <i>Design interface Lihat Homestay</i> | 73 |
| Gambar 58 <i>Design interface Edit Homestay</i> | 73 |
| Gambar 59 <i>Design interface Cetak Data Homestay</i> | 74 |
| Gambar 60 <i>Design interface Pendaftaran Homestay</i> | 74 |
| Gambar 61 <i>Design interface Konfirmasi Pendaftaran Homestay</i> | 75 |
| Gambar 62 <i>Design interface Daftar Pengguna</i> | 75 |
| Gambar 63 <i>Design interface Pemesanan Homestay</i> | 76 |

| | |
|--|-----|
| Gambar 64 <i>Design interface</i> Konfirmasi Pemesanan..... | 77 |
| Gambar 65 <i>Design interface</i> Cetak Laporan Pemesanan..... | 77 |
| Gambar 66 <i>Design interface</i> Register..... | 78 |
| Gambar 67 <i>Design interface</i> Login..... | 78 |
| Gambar 68 <i>Design interface</i> Splashscreen..... | 79 |
| Gambar 69 <i>Design interface</i> Home | 80 |
| Gambar 70 <i>Design interface</i> List Homestay | 80 |
| Gambar 71 <i>Design interface</i> Informasi Homestay | 81 |
| Gambar 72 <i>Design interface</i> Pemesanan Homestay | 82 |
| Gambar 73 <i>Design interface</i> Rincian Pemesanan Homestay | 82 |
| Gambar 74 <i>Design interface</i> Rincian Akhir Pemesanan Homestay | 83 |
| Gambar 75 <i>Design interface</i> Pembayaran..... | 84 |
| Gambar 76 <i>Design interface</i> Bayar di Tempat..... | 85 |
| Gambar 77 <i>Design interface</i> Transfer Lunas Pembayaran | 85 |
| Gambar 78 <i>Design interface</i> Transfer DP Pembayaran..... | 86 |
| Gambar 79 <i>Design Interface</i> Register Homestay | 86 |
| Gambar 80 <i>Design interface</i> Edit Homestay..... | 87 |
| Gambar 81 <i>Design interface</i> Hapus Homestay | 87 |
| Gambar 82 <i>Design interface</i> Tambah Kamar..... | 88 |
| Gambar 83 <i>Design interface</i> Daftar Kamar | 88 |
| Gambar 84 <i>Design interface</i> Edit Kamar | 89 |
| Gambar 85 <i>Design interface</i> Delete Kamar | 89 |
| Gambar 86 <i>Design interface</i> Notifikasi | 90 |
| Gambar 87 Rancangan Sistem..... | 94 |
| Gambar 88 Struktur Maven..... | 106 |
| Gambar 89 Hasil Mengeola Data Homestay | 131 |
| Gambar 90 Hasil Tambah Homestay | 132 |
| Gambar 91 Hasil Lihat Detail Homestay | 132 |
| Gambar 92 Hasil Edit Homestay | 132 |
| Gambar 93 Hasil Hapus Homestay..... | 133 |
| Gambar 94 Hasil Pendaftaran Homestay | 133 |
| Gambar 95 Hasil Memberikan Persetujuan | 134 |

| | |
|---|-----|
| Gambar 96 Hasil Daftar Pengguna | 134 |
| Gambar 97 Hasil Pemesanan Homestay | 135 |
| Gambar 98 Hasil Konfirmasi Pemesanan | 135 |
| Gambar 99 Hasil Cetak Laporan pemesanan | 136 |
| Gambar 100 Splashscreen Pengunjung..... | 137 |
| Gambar 101 Login Pengunjung..... | 137 |
| Gambar 102 Register Pengunjung | 137 |
| Gambar 103 Home Pengunjung | 137 |
| Gambar 104 List Homestay..... | 138 |
| Gambar 105 Detail Homestay | 138 |
| Gambar 106 Tampilan Lokasi..... | 138 |
| Gambar 107 Pemesanan..... | 138 |
| Gambar 108 Rincian Akhir | 139 |
| Gambar 109 Pembayaran | 139 |
| Gambar 110 Hasil Tampilan Splash Screen..... | 140 |
| Gambar 111 Hasil Tampilan Login | 140 |
| Gambar 112 Hasil Tampilan Register..... | 141 |
| Gambar 113 Hasil Daftar Kamar | 141 |

BAB I

PENDAHULUAN

Pada Bab pendahuluan akan diuraikan latar belakang penggerjaan Tugas Akhir, rumusan masalah, tujuan, lingkup penggerjaan Tugas Akhir, pendekatan yang dilakukan dan sistematika penyajian dokumen.

1.1 Latar Belakang

Indonesia merupakan negara yang kaya akan pesona dan panorama alamnya. Keindahan alam yang dimiliki Indonesia mengundang pengunjung lokal maupun mancanegara ke lokasi – lokasi pariwisata yang ada. Kabupaten Toba Samosir yang terletak di Provinsi Sumatera Utara merupakan salah satu dari tujuh kabupaten yang mengelilingi Danau Toba sebagai danau terluas di Indonesia. Dalam pengembangan kawasan pariwisata, Danau Toba merupakan salah satu dari 88 objek wisata yang termasuk kedalam Kawasan Strategis Pariwisata Nasional (KSPN) berdasarkan peraturan pemerintah Nomor 50 tahun 2011 tentang Rencana Induk Pembangunan Kepariwisataan Nasional tahun 2010-2025, sehingga menjadi prioritas dalam pembangunan kepariwisataan [1].

Didukung oleh sumber daya alam dan keindahan Danau Toba, sektor ini berpotensial menjadi objek wisata andalan di Kabupaten Toba Samosir. Keberadaan pariwisata Toba Samosir menjadi salah satu faktor berkembangnya perekonomian daerah Toba Samosir dikarenakan sifatnya yang memiliki efek lanjutan yang panjang terhadap faktor pendukung pariwisata. Setiap pengunjung yang ingin melakukan perjalanan wisata kesuatu daerah dengan tujuan rekreasi, pengembangan pribadi, ataupun mempelajari keunikan daya tarik wisata dipastikan akan membutuhkan fasilitas, layanan serta informasi. Salah satu fasilitas yang dibutuhkan pengunjung adalah penginapan di daerah wisata tersebut. Kebutuhan penginapan menjadi salah satu hal menarik untuk dikaji yang disebabkan banyaknya pengunjung yang ingin menginap dan bersosialisasi dengan warga sekitar seperti menjadikan rumah- rumah penduduk sekitar dijadikan sebagai penginapan atau yang sering disebut dengan *homestay*. *Homestay* merupakan tempat tinggal penduduk di desa wisata yang sebagian kamarnya disewakan kepada pengunjung dengan keunggulan dapat melakukan interaksi aktif antara pengunjung

dengan pemilik rumah (Kemenpar 2016). Kabupaten Toba Samosir merupakan salah satu tujuan wisata yang berada dikawasan Danau Toba yang memiliki potensi pengembangan wisata yang baik dengan adanya berbagai objek wisata dan atraksi wisata seperti potensi alam, dan potensi budaya.

Banyaknya objek wisata membuat pengunjung yang berkunjung ke daerah wisata Toba Samosir yang memerlukan waktu yang cukup lama untuk menjelajahi objek wisata yang terdapat di Kabupaten Toba Samosir sehingga pengunjung membutuhkan penginapan selama liburan. Di Kabupaten Toba Samosir, khususnya di Kecamatan Balige yang memiliki objek wisata Pantai Lumban Bul-bul sedang dicanangkan program pengembangan desa wisata berbasis *homestay* melihat fenomena pariwisata yang semakin berkembang dan tingkat kunjungan wisata yang meningkat. Maka *homestay* adalah pilihan yang tepat untuk pariwisata berbasis masyarakat, disamping harga yang murah dan terjangkau sekaligus dapat meningkatkan tingkat pendapatan masyarakat.

Kabupaten Toba Samosir yang terdiri dari 16 kecamatan dan 244 desa memiliki 74 *homestay* dalam perkembangan bisnisnya. *Homestay* sudah dikenal oleh sebagian masyarakat. Namun informasi dari 74 *homestay* tersebut belum tersebar dengan baik. Sehingga masyarakat luas belum mengetahui detail tentang *homestay* tersebut mulai dari alamatnya, cara pemesanan kamar serta fasilitas yang terdapat di *homestay* tersebut. Pada saat ini, belum ada sistem/aplikasi yang menyediakan informasi *homestay* di seluruh daerah Toba Samosir sehingga pengunjung yang ingin menginap di *homestay* kesulitan karena kurangnya aksesibilitas informasi. Untuk mengetahui letak *homestay* pengunjung hanya dapat melihat pamflet yang ada di sekitar desa wisata yang dikunjunginya kemudian mendatangi langsung *homestay* yang akan ditempatinya. Maka untuk mengatasi masalah tersebut penulis akan membuat aplikasi *homestay* berbasis android yang menyediakan informasi *homestay* daerah Kabupaten Toba Samosir seperti harga per kamar, lokasi *homestay* pada *maps*, fasilitas yang tersedia beserta *point of interest* dari *homestay* tersebut melalui aplikasi mobile yang akan dibangun.

1.2 Rumusan Masalah

Adapun rumusan permasalahan yang akan dibahas dalam pembangunan aplikasi ini yaitu:

1. Bagaimana merancang dan membangun aplikasi yang menyajikan informasi *homestay*?
2. Bagaimana pengunjung dapat mengetahui informasi mengenai homestay di kabupaten Toba Samosir?

1.3 Tujuan

Tujuan yang ingin dicapai dari pelaksanaan Tugas Akhir ini adalah untuk membangun aplikasi *homestay* daerah Toba Samosir berbasis android yang menyajikan informasi *homestay* seperti harga kamar, lokasi *homestay* pada *maps*, fasilitas yang tersedia, beserta *point of interest* yang terdapat di *homestay* tersebut. Selain itu pengunjung dapat melakukan pemesanan *homestay* melalui aplikasi dan melakukan transaksi pembayaran.

1.4 Lingkup

Lingkup masalah yang ingin dicapai dalam pelaksanaan tugas akhir ini adalah:

1. Pembangunan aplikasi *homestay* daerah Toba Samosir berbasis android dengan bahasa pemrograman kotlin.
2. Aplikasi *homestay* Toba Samosir yang akan dibangun ditujukan untuk memperkenalkan *homestay* Toba Samosir yang berisi daftar *homestay* yang dikelompokkan berdasarkan kecamatan dan menyajikan informasi detail *homestay* seperti harga kamar, lokasi *homestay* pada *maps* fasilitas yang tersedia dan *point of interest* yang terdapat di *homestay* tersebut.
3. Aplikasi yang dibangun dapat diakses dengan menggunakan internet.
4. Pengunjung dapat melakukan pemesanan *homestay* melalui aplikasi yang dibangun.
5. Aplikasi dibangun menggunakan metodologi *prototype*.

1.5 Pendekatan

Metodologi rancang bangun aplikasi *homestay* merupakan suatu cara/prosedur yang digunakan untuk mengumpulkan data dengan perantara teknik tertentu. Dalam pengerjaan Tugas Akhir ini akan menggunakan metodologi sebagai berikut:

1.5.1 Objek Perancangan Aplikasi

Objek yang digunakan dalam perancangan aplikasi ini adalah *homestay* daerah Toba Samosir yaitu terdiri dari 74 *homestay* dari 16 kecamatan. Namun dalam pembangunan aplikasi ini, penulis menggunakan 12 homestay dari 74 homestay yang ada di Kabupaten Toba Samosir.

1.5.2 Jenis Data

Jenis data yang digunakan dalam rancang bangun aplikasi ini adalah data kuantitatif. Jenis data kuantitatif yaitu data yang dalam bentuk angka seperti harga sewa per kamar dan data jumlah kamar yang tersedia.

1.5.3 Sumber Data

1. Data Primer

Data primer adalah data yang diperoleh langsung dari observasi dan *survey* langsung ke *homestay* untuk mengetahui data-data yang akan digunakan pada aplikasi. Data tersebut meliputi harga sewa kamar, jumlah kamar tersedia, fasilitas yang tersedia, dan *point of interest* yang terdapat di *homestay* tersebut.

2. Data Sekunder

Data yang diperoleh dalam bentuk informasi dari jurnal, dan pustaka.

1.5.4 Metode Pengumpulan Data

Metode yang dilakukan dalam pengumpulan data adalah dengan melakukan wawancara langsung dengan pemilik *homestay* untuk mengetahui masalah yang dihadapi dalam pemesanan *homestay*. Untuk pengumpulan data, penulis melakukan 3 (tiga) kali pertemuan dengan pemilik *homestay*.

1.5.4.1 Observasi dan Wawancara

Pada tahap ini, penulis melakukan wawancara kepada pemilik *homestay*.

Pertanyaan yang diberikan menerapkan 5W + 1H, seperti berikut:

1. *What* (Apa fasilitas yang dimiliki *homestay*?)
2. *When* (Kapan *homestay* didirikan?)
3. *Where* (Dimana lokasi *homestay*?)
4. *Who* (Siapa pemilik *homestay*?)
5. *Why* (Mengapa *homestay* dibangun?)
6. *How* (Bagaimana proses pemesanan dan pembayaran *homestay*?)

1.5.4.2 Melakukan Pertemuan Kedua

Pada tahap ini penulis mengunjungi lokasi *homestay* untuk menanyakan apakah pemilik *homestay* dapat memahami desain *prototype* aplikasi yang sudah dirancang oleh penulis. Tahap ini menggunakan desain yang bersifat *low fidelity*. Penulis menggunakan aplikasi Balsamiq Mockup 3 sebagai *tools* desain.

1.5.4.3 Melakukan Pertemuan Ketiga

Pada tahap ini penulis kembali mengunjungi lokasi *homestay* untuk menanyakan apakah pemilik *homestay* dapat memahami desain *prototype* aplikasi yang dirancang oleh penulis. Tahapan ini menggunakan desain yang bersifat *high fidelity* karena sudah menggunakan warna dan sudah mengandung deskripsi desain yang dibuat. Pada tahap ini penulis menggunakan *tools* Adobe XD.

1.5.5 Metodologi Pengembangan Sistem

Metodologi yang dilakukan dalam merancang bangun aplikasi *homestay* daerah Toba Samosir ini dilakukan dengan menggunakan metode *prototype* berdasarkan siklus hidup perangkat lunak. Dalam pengembangan sistem informasi dikenal berbagai macam metode salah satunya adalah metode *prototype*. *Prototype* disebut juga desain aplikasi yang cepat (*rapid application design/RAD*) karena menyederhanakan dan mempercepat desain sistem. Pengembangan yang cepat dan pengujian terhadap model kerja (*prototype*) dari aplikasi baru melalui proses interaksi berulang-ulang yang biasa digunakan. Pendekatan ini terdiri dari beberapa tahapan yang dilakukan sehingga menghasilkan sebuah aplikasi yang dapat

menjawab permasalahan yang telah diuraikan pada latar belakang Tugas Akhir ini. Adapun tahapan dari metode *prototype* adalah:

1. Analisa Kebutuhan

Penulis dan pemilik *homestay* sama-sama mendiskusikan apa yang menjadi kebutuhan dari *homestay* yang ada saat ini. Dan mendiskusikan apa keinginan pemilik *homestay* yang akan dijadikan fungsi pada aplikasi yang akan dibangun.

2. Mendesain *Prototype*

Mendesain *prototype* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pemilik *homestay* yaitu membuat desain *interface* dari aplikasi yang akan dibangun.

3. Evaluasi *Prototype*

Evaluasi ini dilakukan oleh pemilik *homestay*, dan akan diputuskan apakah desain *prototype* yang dibangun sudah sesuai dengan kebutuhan pemilik *homestay*.

4. *Coding* Sistem

Pada tahap ini desain *prototype* yang sudah dibuat dan dievaluasi akan dijadikan sebagai gambaran untuk membangun aplikasi dan dimasukkan ke dalam codingan.

5. Menguji Sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, pengujian akan dilakukan dahulu sebelum aplikasi digunakan.

6. Evaluasi Sistem

User mengevaluasi apakah sistem yang sudah jadi sesuai dengan kebutuhan yang diharapkan *user*.

7. *Deployment*

Aplikasi *Homestay* Daerah Toba Samosir yang telah diimplementasikan akan di *deploy* dalam bentuk .apk sehingga dapat dilakukan instalasi pada *device mobile* berbasis android.

1.6 Sistematika Penyajian

Dokumen Tugas Akhir ini terdiri dari enam Bab yang disusun secara sistematis, yakni sebagai berikut:

Bab I Pendahuluan

Bab I Pendahuluan, diuraikan penjelasan mengenai latar belakang masalah dan alasan mengapa topik ini dibahas, tujuan, lingkup, pendekatan, sistematika penyajian dalam Tugas Akhir ini.

Bab II Tinjauan Pustaka

Bab II Tinjauan Pustaka menjelaskan mengenai dasar - dasar teori yang relevan dengan topik tugas akhir yang digunakan.

Bab III Analisis

Bab III Analisis menjelaskan mengenai analisis kebutuhan dan proses yang dilakukan untuk pengembangan aplikasi.

Bab IV Design

Bab IV Design menunjukkan desain aplikasi yang dibuat oleh penulis.

Bab V Implementasi

Bab V Implementasi menunjukkan implementasi pembangunan aplikasi.

Bab VI Testing

Bab VI Testing menjelaskan proses pengujian aplikasi yang di sudah dibangun.

Bab VII Hasil dan Pembahasan

Bab VI Hasil dan Pembahasan menjelaskan mengenai hasil pengujian yang sudah dilakukan sebelumnya dan juga membahas kekurangan yang terdapat dalam aplikasi.

Bab VIII Kesimpulan dan Saran

Kesimpulan dan Saran menjelaskan mengenai kesimpulan yang dapat diambil dari hasil analisa dan juga saran yang dibutuhkan untuk merancang aplikasi tersebut.

BAB II

TINJAUAN PUSTAKA

Pada bab ini diuraikan dasar teori yang relevan dengan objek kajian Tugas Akhir.

2.1 Pariwisata

Pariwisata disebut sebagai suatu perpindahan dari suatu tempat ke tempat yang lain, dimana sifat yang dimiliki bersifat sementara dengan pelakunya berupa perorangan maupun kelompok sebagai proses mencari keseimbangan atau keserasian dan kebahagiaan hidup dalam dimensi sosial, budaya, alam dan ilmu. Menurut Undang-Undang RI No.10 tahun 2009 tentang kepariwisataan adalah bahwa wisata merupakan sebuah kegiatan mobilisasi yang dilakukan pribadi maupun sekelompok orang dengan tujuan mengunjungi suatu tempat tertentu untuk tujuan rekreasi, pengembangan pribadi, maupun mempelajari keunikan wisata yang dikunjungi dalam waktu sementara [2].

Pariwisata adalah satu dari industri gaya baru, yang berpotensi meningkatkan perekonomian secara cepat dalam hal kesempatan kerja, pendapatan, taraf hidup dan dalam mengaktifkan sektor produksi lain di dalam negara penerima pengunjung (Wahab, 2003: 5). Sektor pariwisata termasuk satu dari sekian banyak sektor strategis dalam menggerakkan perekonomian Indonesia dan berperan besar dalam perekonomian global. Berlangsungnya revolusi 3T, *transport, telecommunication* dan *tourism* [3].

Pariwisata memiliki fasilitas yang menunjang perkembangan pariwisata, dimana fasilitas wisata ditujukan sebagai bentuk pelayanan dan membantu pengunjung lebih mudah dalam melakukan kegiatan atau aktivitas untuk mendapat pengalaman rekreasi dan memberikan pelayanan kepada wisatawan.

Menurut Lawson dan Baud-Bovy dalam buku berjudul *tourism and recreation handbook of planning and design* (1997:17) terdapat pembagian fasilitas menjadi 2 tipe yakni:

1. Fasilitas dasar bagi setiap resort atau tempat rekreasi, dengan memberi layanan kepada pengunjung secara umum seperti akomodasi, makanan, minuman, hiburan, dan infrastruktur dasar pengelolaan sebuah objek wisata.

2. Fasilitas khusus sesuai dengan karakteristik yang tersedia yang menunjukkan karakter alamiah sebuah objek wisata.

Menurut Roger A Lancaster (1983:51) dalam buku standardisasi fasilitas berpendapat bahwa standar fasilitas adalah “sebagai jumlah fasilitas rekreasi dengan segala kelengkapannya yang perlu disediakan bagi kebutuhan masyarakat untuk berbagai macam aktivitas rekreasi”. Lancaster juga menyebutkan beberapa syarat yang dipadat dijadikan sebagai dasar panduan dalam penentuan fasilitas wisata, seperti:

1. Realistik.
2. Logis dan bermanfaat.
3. Berdasar pada analisa yang tepat berdasarkan informasi terbaik yang diperoleh.

2.2 *Homestay*

2.2.1 Pengertian *Homestay*

Homestay merupakan sebuah tempat tinggal (rumah) yang dimana sebagian dari kamar pemilik rumah disewakan kepada tamu dalam kurun waktu tertentu guna mempelajari budaya setempat atau suatu rutinitas tertentu. *Homestay* biasanya berada terletak di daerah kawasan wisata dengan tujuan untuk disewakan kepada pengunjung yang secara langsung dapat melihat kehidupan masyarakat sehari-hari, melihat pemandangan, dan menjalani kehidupan penduduk lokal [4].

Tujuan diadakannya *homestay* adalah untuk mengakomodasikan pengunjung disuatu daerah/desa bersamaan dengan pemilik rumah(*host family*) sehingga pengunjung dapat belajar tentang *gaya hidup* lokal dan budaya. Dalam pengoperasiannya, ada hal yang membuat bangunan baru diluar bangunan rumah tinggal yang ada sebelumnya, ada pula yang memanfaatkan rumah tinggal yang ada sebelumnya, ada pula yang menggunakan rumah tinggal *lama* dan dihuni secara bersama-sama (Koestanto, 2016).

2.2.2 Kriteria dan Persyaratan *Homestay*

Berikut ini kriteria dan persyaratan *homestay* berdasarkan ASEAN *Homestay* standar yang berpengaruh dengan pemahaman potensi dan konsep desain, yaitu [4]:

1. Lokasi dan aksesibilitas

Lokasi *homestay* harus dapat dicapai oleh transportasi apa saja.

2. Akomodasi

- a. Rumah

Struktur rumah harus dalam keadaan baik dan aman, kamar tidur untuk tamu harus terpisah dari kamar tidur pemilik rumah tersebut, harus ada minimal satu kamar mandi dalam bagi pengunjung serta persediaan listrik dan air bersih yang mencukupi.

- b. Kamar Tidur

Menyediakan perabotan atau fasilitas kamar, maksimal empat kamar dari total jumlah kamar yang ada di rumah tersebut dimana status kamar tidak digunakan oleh anggota keluarga dan menyediakan tipe ranjang yang standar dan tepat seperti *single bed* ataupun *double bed* serta kasur dan bantal.

- c. Kamar Mandi/Toilet

Kamar mandi yang disediakan bertipe kloset duduk atau jongkok di dalam rumah dengan fasilitas yang standar seperti pintu dan air bersih yang memadai setiap waktu.

3. Aktivitas

- a. Aktivitas desa dan komunitas

Mengoptimalkan dan menampilkan sumber daya local, misalnya:

1. Warisan budaya
2. Usaha lokal
3. Sumber daya alam
4. Perancangan dan pelaksana kegiatan harus mampu meningkatkan interaksi aktif antara masyarakat lokal dengan wisatawan.

- b. Aktivitas sekitar

Kunjungan ke sebuah pertunjukan/pameran wisata populer di area sekitar dikaitkan dengan *homestay* dimana *homestay* sebagai basis dan

berkolaborasi dengan desa di area sekitar untuk menambah variasi kegiatan dan menciptakan efek ganda.

4. *Host* (pemilik rumah/penyedia)

1. Desa/ Kampung dan Komunitas

1. *Homestay* terletak tidak jauh dari tempat pariwisata alam dan budaya yang ada di sekitarnya.
2. Ada pusat/area yang digunakan sebagai basis operasi *homestay* dan aktivitas seperti cara penyambutan, pertunjukan budaya.

2. Penyedia *Homestay* (tuan rumah)

1. Penyedia *homestay* tidak boleh memiliki catatan kriminal dan dalam kondisi sehat (tidak menderita penyakit menular).
2. Penyedia *homestay* harus menghadiri dan menyelesaikan kursus atau *workshop basic homestay*

5. *Sustainability Principle* (Prinsip-prinsip berkelanjutan)

1. Prinsip *economic sustainability*

1. *Employment*

Pengelola *homestay* harus melakukan rekrutmen dan mempekerjakan stafnya dari masyarakat sekitar, mengalokasikan insentif dan bonus kepada staff dengan kinerja yang baik dan menyediakan donasi untuk usaha- usaha mikro untuk masyarakat sekitar ikut serta dalam aktivitas yang terkait dengan *homestay*.

2. *Purchasing*

Menyarankan agar pemilik *homestay* membeli produk-produk usaha kecil lokal, meminta pengunjung membeli produk usaha kecil lokal dan mengatur area penjualan kerajinan tangan di sekitar *homestay*.

3. *Local product/attraction*

Mempromosikan festival lokal dan mengunjungi pasar tradisional terdekat, makanan tradisional dan pertunjukan seni budaya kepada wisatawan.

2. Prinsip *environment sustainability*

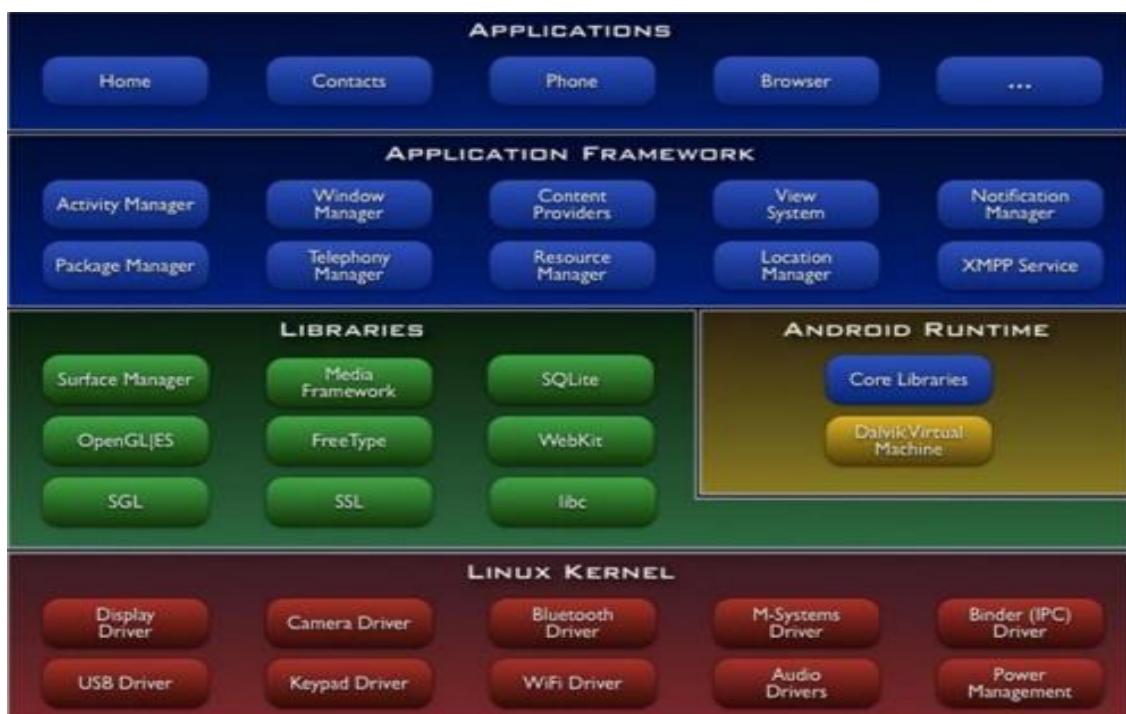
Mengurangi dampak negatif dari pengunjung terutama pada lingkungan alam dan budaya, tidak mendukung pengunjung yang merusak flora dan fauna lokal, menjaga kebersihan pantai, hutan, sungai dan taman.

3. Prinsip *sociocultural sustainability*

Membuat pusat informasi dan *display* tentang kebudayaan lokal dipusat *homestay* dan di tempat umum, memberi informasi kepada pengunjung untuk harus menghormati adat-istiadat, kebiasaan dan perilaku setempat.

2.3 Android

Aplikasi *homestay* daerah Toba Samosir yang akan dibangun pada perangkat *mobile* menggunakan Sistem Operasi android. *Platform* android dipilih dikarenakan sistem operasi ini paling banyak digunakan oleh *user smartphone* dan dapat dibawa kemana – mana. Android merupakan sebuah Sistem Operasi pada *smartphone* dan juga *pc tablet*. android merupakan sebuah sistem operasi yang telah diintegrasikan dengan *Google Mail*, *Google Maps*, *Google Search*, *Google Docs*, dan produk layanan Google lainnya. Integrasi teknologi *mobile* GPS dapat memudahkan dikembangkannya aplikasi *mobile* informasi objek wisata secara *real time* yang lebih interaktif [5]. Android pertama kali dirilis pada tahun 2008, sejak saat itu *google* terus melakukan pengembangan untuk sistem operasi android dan telah merilis lebih dari 8 versi android, pada saat ini versi terbaru android adalah versi 9.0 yang diberi nama *pie* [6]. Pada sektor aplikasi, telah tercatat lebih dari 2, 8 juta aplikasi telah terdaftar dalam android *store* pada tahun 2016. Android menjadi hal yang menarik bagi perusahaan teknologi yang membutuhkan barang jadi, biaya rendah dan kostumisasi OS untuk perangkat berteknologi tinggi (Imaduddin and Permana, 2017).



Gambar 1 Arsitektur Android
Source : (Kariyodiharjo, 2011)

2.4 Mapbox API

MapBox API adalah salah satu layanan penyedia peta digital untuk berbagai platform.

Data Mapbox biasanya diperoleh berdasarkan sumber data terbuka seperti *Open Street Map* dan NAS dan sumber berbayar lainnya seperti Digital Globe. Teknologi Mapbox dirancang dengan menggunakan Node.Js, CouchDB, Mapnik, GDAL dan Leafletjs. MapBox, yang mendukung GeoJSON – *encoding* objek geografis dalam bentuk JSON [7].

2.5 Android Kotlin

Kotlin adalah bahasa pemrograman berbasis *Java Virtual Machine* (JVM) yang dikembangkan oleh Jetbrains [8]. Kotlin merupakan bahasa yang pragmatis untuk android yang mengkombinasikan *object oriented (OO)* dan pemrograman fungsional. Kotlin merupakan bahasa pemrograman yang interoperabilitas yang membuat bahasa ini dapat digabungkan dalam satu project dengan bahasa pemrograman java. Keuntungan yang akan didapatkan dalam pengembangan aplikasi menggunakan kotlin diatas *platform JVM* sebagai berikut:

1. Dapat mengatasi *NullPointerException* yang umumnya terdapat pada Java.
2. Penulisan kode lebih ringkas dan mudah dibaca dibandingkan kode yang ditulis menggunakan bahasa Java.
3. Mudah dipelajari.
4. Dukungan IDE untuk mempermudah dalam pemrograman.

Pada acara Google I/O 2017, *Google* mengumumkan bahasa pemrograman baru yang resmi digunakan untuk pengembangan aplikasi android yaitu bahasa kotlin. Bahasa Kotlin adalah bahasa pemrograman yang dikembangkan oleh *JetBrains*. Bahasa Kotlin memiliki fitur-fitur bahasa *modern* yang dibandingkan dengan Java. Bahasa pemrograman kotlin dinilai dapat memberikan *value* yang lebih dibandingkan bahasa Java.

2.6 Web Service

Web service merupakan sebuah konsep suatu sistem yang dibangun dalam mendukung interoperabilitas interaksi mesin-mesin melalui sebuah jaringan [9]. *Web service* juga berfungsi sebagai RPC (*Remote Procedure Call*) yang dapat melakukan pendefinisian fungsi pada aplikasi web dan mengekspos sebuah API atau UI melalui web. Kelebihan *web service* adalah sebagai jembatan penghubung dengan *database* tanpa menggunakan *driver database* dan tanpa mengetahui jenis DBMS serta membantu proses pertukaran data lebih mudah. Dua buah metode yang dimiliki *web service* berpusat pada layanan dan sumber daya informasi dalam perkembangannya adalah: SOAP (*Simple Object Access Protocol*) dan REST (*Representational State Transfer*).

2.7 Java Development Kit (JDK)

Java Development Kit (JDK) digunakan untuk mengembangkan perangkat lunak berbasis *Java* yang merupakan sekumpulan perangkat lunak, sedangkan JRE digunakan untuk menjalankan program *Java* dan merupakan sebuah implementasi dari *Java Virtual Machine*. JDK biasanya berisi satu atau lebih JRE dan alat pengembangan lain seperti *compiler java*, *bundling*, *debugger*, *development libraries* [10].

2.8 Hyper Text Transfer Protocol (HTTP)

Hyper Text Transfer Protocol adalah sebuah protokol komunikasi dengan konsep *request-response* antara *client* dan *server*. Sebagai sebuah protocol, HTTP dapat menentukan prosedur-prosedur komunikasi seperti format komunikasi dan cara berkomunikasi (Hidayatullah dan Jauhari, 2015).

Menurut Pratama (2014), dalam mengakses suatu situs khususnya *website*, protokol yang umum digunakan oleh pengguna jaringan komputer adalah protokol HTTP. Dimana dalam komunikasi server-client, HTTP paling sedikit menjalankan tiga buah fungsi, sebagai berikut:

1. Menentukan *response web server* (*server*) terhadap *request web browser* (*client*).
2. Membantu *web browser* menyajikan data dan informasi yang diterima dari *web server* berdasarkan permintaan *client*.

3. Membantu menerjemahkan pesan dan perintah yang berasal dari client dan ditujukan ke *server*, serta tanggapan yang dikirim oleh *server* kepada *client* (berdasarkan permintaan dari *client*).

2.9 PHP

PHP (*Hypertext Preprocessor*) adalah sebuah bahasa *script* yang dapat ditanam maupun disisipkan ke dalam halaman web. Umumnya pembangunan web menggunakan PHP, baik web yang bersifat statis, dinamis, manajemen konten web, maupun jenis web yang lain. Karena diproses pada komputer server, PHP disebut sebagai sebuah bahasa pemrograman server *side*. Berbeda dengan beberapa bahasa seperti Javascript yang diproses pada web browser (*client side*). Penggunaan PHP yang secara bebas tanpa membayar menjadikan PHP bersifat *Open Source*. Mudah dan populernya PHP pada saat ini sudah menjadi standar bagi para programmer web di seluruh dunia (MADCOMS, 2016).

2.10 Database

Basis data telah dirancang sebaik mungkin sehingga dapat melakukan manipulasi terhadap data, mengambil dan mencari data secara cepat dan tepat sehingga basis data biasa diartikan sebagai kumpulan data yang terintegrasi dan data tersebut. Selain berisi data, *database* juga berisi metadata [10]. Pada proses pengumpulan, analisis dan penggunaan informasi disebut juga sebagai pengumpulan dan analisis kebutuhan (*Requirement Collection and Analysis*) untuk mengidentifikasi kebutuhan *user* terhadap suatu sistem baru. Perancangan basis data (*database design*) merupakan sebuah proses membuat desain baru dimana proses ini mendukung operasional dan tujuan suatu lembaga perusahaan[11]. Pemanfaatan *database* dapat digunakan sebagai tempat menyimpan data atau mengubah dan menampilkan data kembali dengan cepat dan tepat.

Menurut Pratama (2016), *Database* dapat menjadi sebuah wadah bagi data maupun informasi, dimana wadah tersebut memenuhi fungsi-fungsi berikut:

1. Tempat menyimpanan data.
2. Tempat memproses/memanipulasi data menjadi informasi.
3. Tempat mengolah data menjadi informasi.
4. Tempat mengakses data dan informasi.

2.11 Design Pattern MVP (Model View Presenter)

Design pattern adalah sebuah hasil solusi yang dirancang sebagai bentuk penyelesaian masalah yang bersifat teknis dengan menyediakan satu arsitektur aplikasi. Dalam pengembangan aplikasi *homestay* ini menggunakan *design pattern Model View Presenter* (MVP). Tujuan pengimplementasian MVP adalah untuk memisahkan bagian *data presentation*, *user interface* (UI), dan *business logic* [12]. Teknik pemrograman yang mengarah pada *page event model* adalah ciri khas MVP. *Model* disebut dengan *domain level object*, yang disebut dengan *business logic*. Menjaga data dan fungsi/ *method* pada aplikasi untuk secara konsisten dapat diakses merupakan tugas utama *model*. Yang bertanggung jawab terhadap representasi antarmuka dengan pengguna adalah *view*. Sedangkan *presenter* bertanggung jawab untuk memanipulasi *model* sebagai respon terhadap *input* pengguna.

2.12 Representational State Transfer (REST)

REST merupakan salah satu gaya arsitektur yang memiliki aturan seperti keseragaman antar muka, sehingga dengan diterapkannya aturan tersebut pada web services, maka mampu memaksimalkan kinerja *web services* terutama pada performa, skalabilitas, dan kemudahan dalam proses modifikasi [13]. Protokol HTTP yang bersifat *stateless* juga digunakan oleh REST. Perintah HTTP yang digunakan adalah perintah GET, POST, PUT dan DELETE. Informasi yang diterima lebih mudah dibaca dan diparsing pada sisi *client* karena hasil yang dikirimkan server berbentuk format XML maupun JSON sederhana tanpa memakai protokol pemaketan data. Cara akses yang mudah dan hasil data yang memiliki format JSON sangat cocok digunakan sebagai *backend* dalam penerapan *web service* karena ukuran file yang menjadi lebih kecil.

Dalam arsitektur REST, server yang mengikuti arsitektur REST menyediakan akses ke sumber data dan klien yang mengambil data. Berbagai bentuk format yang digunakan RESR untuk menyajikan data, seperti teks, JSON dan XML, seperti:

1. *GET* sebagai penyediakan akses untuk membaca sumber data.
2. *PUT* sebagai fungsi yang dipakai dalam memperbarui data yang tersedia.
3. *DELETE* untuk menghapus data.

4. *POST* untuk membuat data baru.

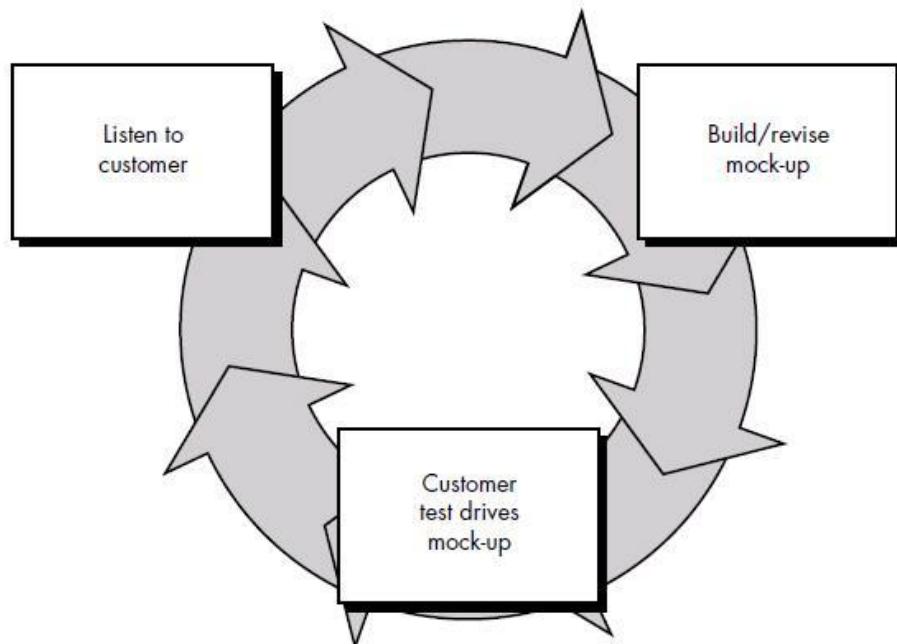
2.13 Metode *Prototype*

Prototype merupakan versi awal dari sistem perangkat lunak yang digunakan untuk mendemonstrasikan konsep-konsep, percobaan rancangan, dan menemukan lebih banyak masalah dan solusi yang memungkinkan [13].

Sistem dengan model *prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik.

Beberapa keunggulan dalam menggunakan metode *prototype* [14]:

1. Dalam penyamaan persepsi pembuatan model untuk sistem yang akan menjadi dasar pengembangan sistem operasionalnya, hal yang pertama dilakukan adalah melakukan komunikasi antar *developer* dan pengguna.
2. Keterlibatan dan partisipasi pengguna secara aktif dalam menentukan model sistem dan sistem operasionalnya sehingga nanti pengguna akan merasa puas karena sistem yang dibangun sudah sesuai dengan keinginan dan permintaannya.
3. Kualitas sistem yang dibangun sesuai dengan keinginan *user*.



Gambar 2 Metode *Prototype*
(Sumber: Khosrow-Pour, 2005)

Gambar di atas menjelaskan bahwa metode *prototype* dimulai dengan mencatat semua kebutuhan pengguna. Pertemuan antara *developer* dan pengguna bertujuan untuk menetapkan tujuan akhir terhadap perangkat lunak dan melakukan identifikasi terhadap setiap persyaratan yang diperlukan. Kemudian, *developer* akan membuat sebuah deksripsi berupa contoh aplikasi yang akan dapat direpresentasikan kepada pengguna. Fokus gambaran di atas terletak pada representasi aspek dari aplikasi yang akan terlihat oleh pengguna [15].

2.14 Latitude dan Longitude

Bagian bumi terbelah secara horizontal menjadi dua bagian yakni bagian utara dan bagian selatan oleh garis lintang atau garis khayal. Garis lintang 0 derajat atau garis khatulistiwa berperan sebagai pembelah bumi. Di atas khatulistiwa terdapat garis yang disebut sebagai Lintang Utara (LU) dan di bawah khatulistiwa terdapat garis yang disebut Lintang Selatan (LS). Garis lintang di Kutub Utara atau Kutub Selatan berkisar 0-90 derajat dengan besaran tertinggi garis lintang mencapai 90 derajat. Umumnya, penotasian sudut lintang dengan menggunakan huruf Yunani *phi*.

Garis Bujur (*Longitude*) merupakan garis khayal yang terbentang di belahan bumi secara vertikal sehingga bumi terbagi menjadi dua bagian, yaitu Timur dan Barat, dimana bagian ini juga terhubung dengan kutub Utara dan Selatan. Garis yang tegak lurus dengan garis khatulistiwa menjadi patokan waktu di dunia yang berada di Royal Observatory Greenwich, Inggris. Garis bujur ini dilambangkan dengan huruf Yunani lambda (λ) dan biasa dinyatakan dalam bentuk derajat, menit dan detik [16]. Dalam Pembangunan aplikasi homestay daera toba samosir ini, penulis menggunakan titik *latitude* dan *longitude* untuk menentukan lokasi *homestay* pada *maps*.

BAB III ANALISIS

Pada bab analisis menjelaskan mengenai hasil analisis terhadap pembangunan Aplikasi Pariwisata Toba Samosir yang mencakup analisis pengumpulan data, *current system*, *target system*, *user characteristics*, *use case diagram*, dan *use case scenario*.

3.1. Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan *developer* dengan melakukan wawancara kepada beberapa pemilik *homestay* yang berada di sekitaran Balige, Toba Samosir. Wawancara dilakukan untuk memperoleh informasi mengenai data *homestay* seperti harga, fasilitas yang disediakan, jumlah kamar, foto *homestay*, dan tempat yang berada di sekitaran *homestay*. Data-data yang didapatkan dari wawancara dapat dilihat pada lampiran di halaman belakang. Wawancara yang dilakukan sebanyak 3 kali wawancara dengan pemilik *homestay* di kabupaten Toba dan melakukan pengujian terhadap desain prototype yang telah dikembangkan.

3.2 Metode Pengembangan Sistem

Metode pengembangan sistem pada aplikasi *homestay* yaitu menggunakan metode *prototype*. Berikut tahapan metode *prototype* yang dilakukan *developer* dalam merancang bangun aplikasi *homestay* daerah Toba Samosir.

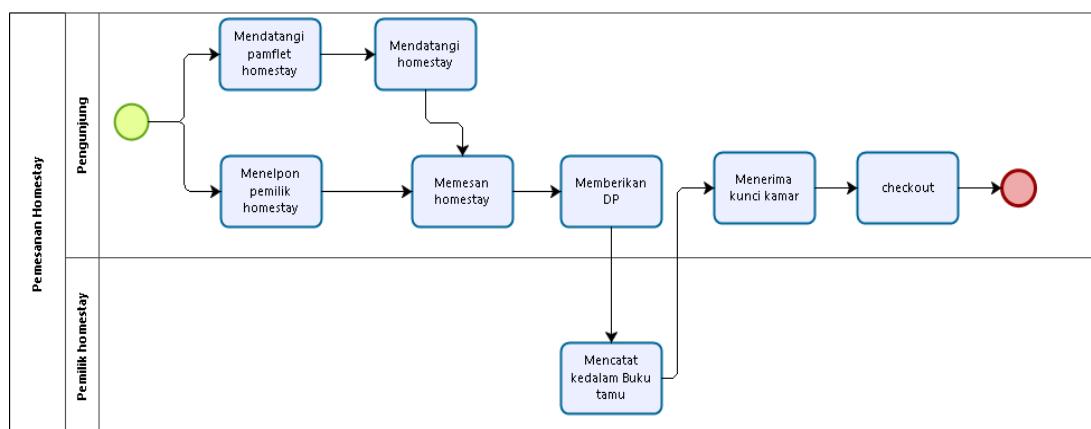
3.2.1 Analisis dan Penentuan Kebutuhan

Pada tahap analisis dan penentuan kebutuhan untuk mengidentifikasi kebutuhan *functional system* yang akan dibangun dengan pemodelan *use case diagram*, *use case scenario*, *current system*, dan *target system*. Sub Bab ini juga mendefinisikan *main function* sesuai dengan kesepakatan antara *user* dengan *developer* ketika melakukan *user requirement*. Fungsi utama aplikasi *homestay* daerah Toba Samosir adalah untuk membantu dan memudahkan pengunjung mencari tempat penginapan berupa *homestay* di daerah wisata yang dikunjunginya. Selain itu terdapat fungsi pemesanan *homestay* melalui aplikasi tersebut, dan untuk pendaftaran *homestay* yang dimilikinya melalui aplikasi tersebut.

3.2.1.1 Current System

Berdasarkan hasil *observasi* dan hasil wawancara dengan *client* sistem yang berjalan saat ini adalah pengunjung melihat pamflet *homestay* atau bertanya kepada masyarakat di sekitaran daerah wisata, kemudian apabila pengunjung memperoleh informasi dari masyarakat setempat maka pengunjung akan menelpon pemilik *homestay* untuk memesan kamar *homestay*, setelah itu pengunjung datang langsung ke *homestay* tersebut dan melakukan pembayaran *homestay*.

Berikut ini bisnis proses pemesanan *homestay* yang berjalan saat ini:

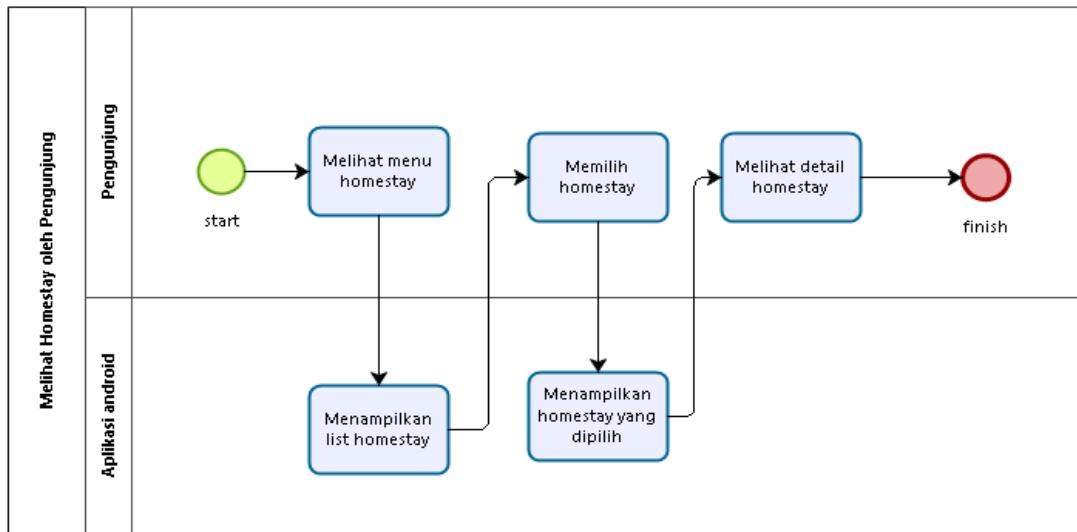


Gambar 3 Bisnis Proses Current system

3.2.1.2 Target System

Pada subbab ini menggambarkan target *system* aplikasi yang akan dibangun sesuai dengan kebutuhan pengunjung dan pemilik *homestay* setelah melakukan *user gathering*.

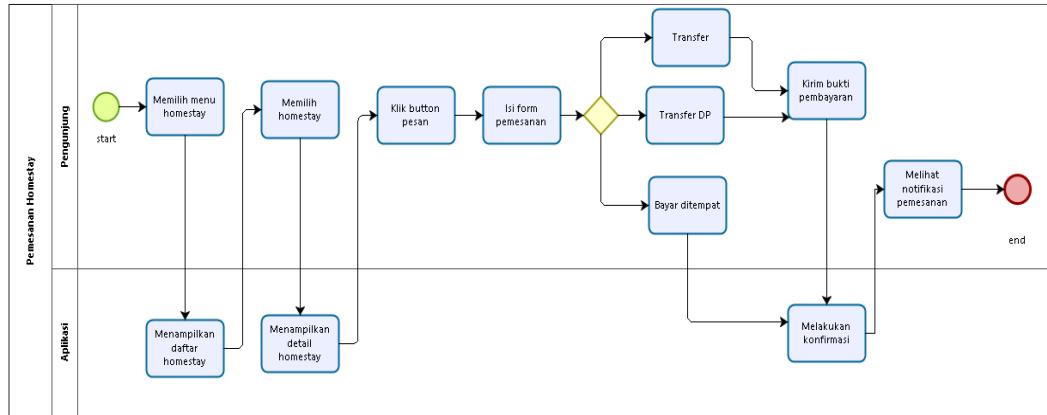
1. Melihat *Homestay*



Gambar 4 Bisnis Proses Melihat *Homestay*

Gambar 4 ini menjelaskan tentang proses pengunjung melihat *homestay* di daerah Toba Samosir. Pengunjung akan membuka aplikasi dan akan melihat halaman utama yang berisi daftar *homestay* daerah Toba Samosir. Kemudian pengunjung memilih salah satu *homestay*, aplikasi akan menampilkan detail *homestay* yang dilihatnya.

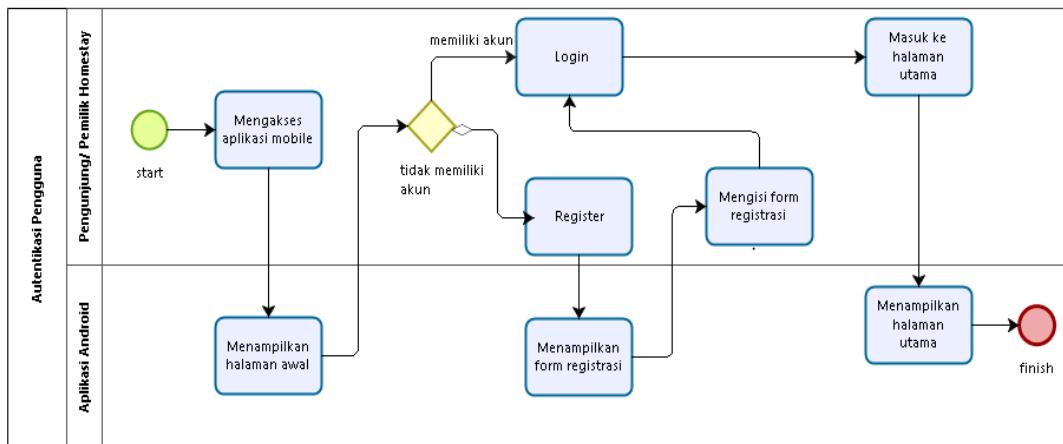
2. Pemesanan Homestay



Gambar 5 Bisnis Proses Pemesanan Homestay

Gambar 5 ini menjelaskan tentang proses pengunjung melakukan pemesanan *homestay*. Pengunjung akan *login* terlebih dahulu, kemudian aplikasi akan menampilkan halaman utama. Pengunjung akan memilih *homestay* dan melihat detailnya. Apabila *pengunjung* ingin memesan *homestay*, pengunjung akan mengklik tombol pesan. Aplikasi akan menampilkan form pemesanan *homestay*. Kemudian akan ada 2 pilihan pembayaran yaitu bayar di tempat atau transfer. Jika pengunjung memilih bayar ditempat maka pengunjung akan memberikan bayaran secara langsung, tetapi jika pengunjung memilih transfer maka pengunjung harus mengirim bukti pembayayaran

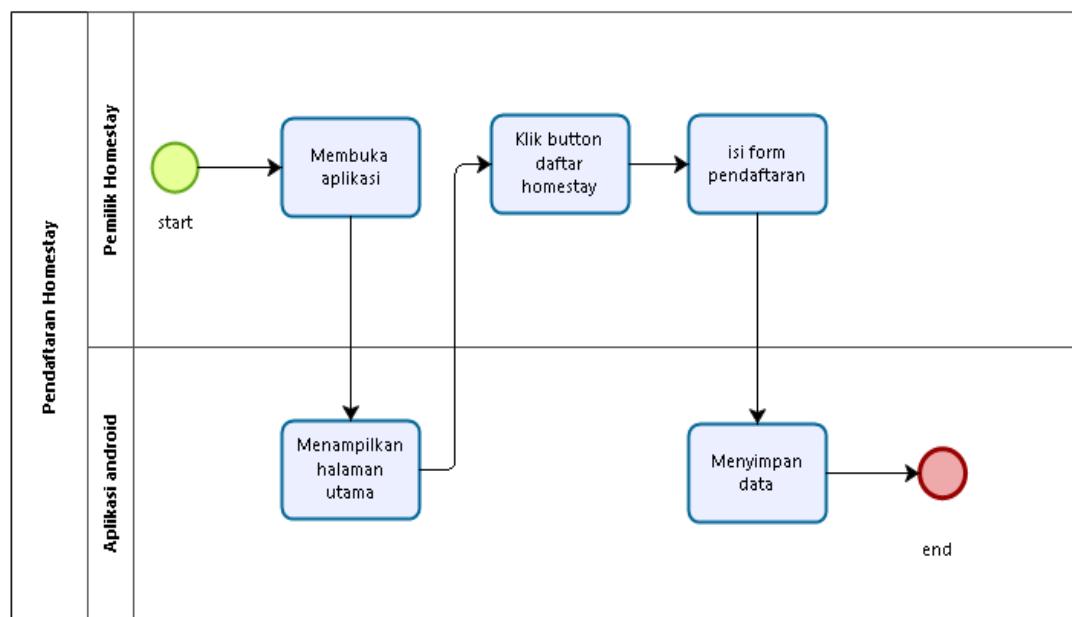
3. Autentikasi (Register dan Login) oleh Pengguna



Gambar 6 Bisnis Proses Autentikasi Akun

Gambar 6 ini menjelaskan tentang proses pengguna mengakses aplikasi untuk memperoleh informasi. Pengguna harus *login* terlebih dahulu dengan mendaftarkan diri melalui *register*. Jika *register* telah selesai, pengguna dapat *login* dan menuju halaman utama aplikasi.

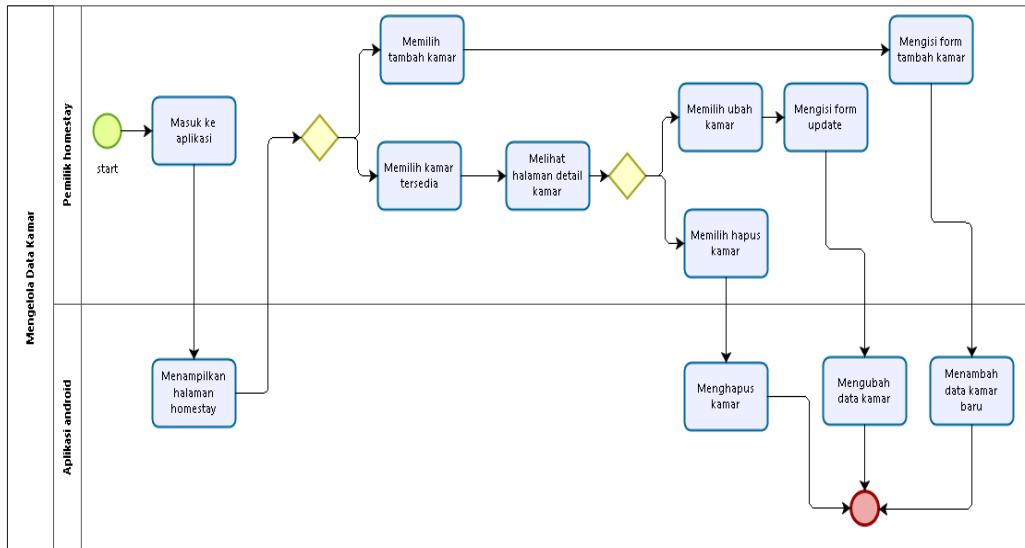
4. Pendaftaran *Homestay*



Gambar 7 Bisnis Proses Pendaftaran *Homestay*

Gambar 7 ini menjelaskan tentang proses pendaftaran *homestay*. Pemilik *homestay* akan memilih tombol daftar kemudian akan menampilkan form daftar *homestay*. Pemilik usaha akan mengisi form pendaftaran dan menunggu konfirmasi pendaftaran *homestay*.

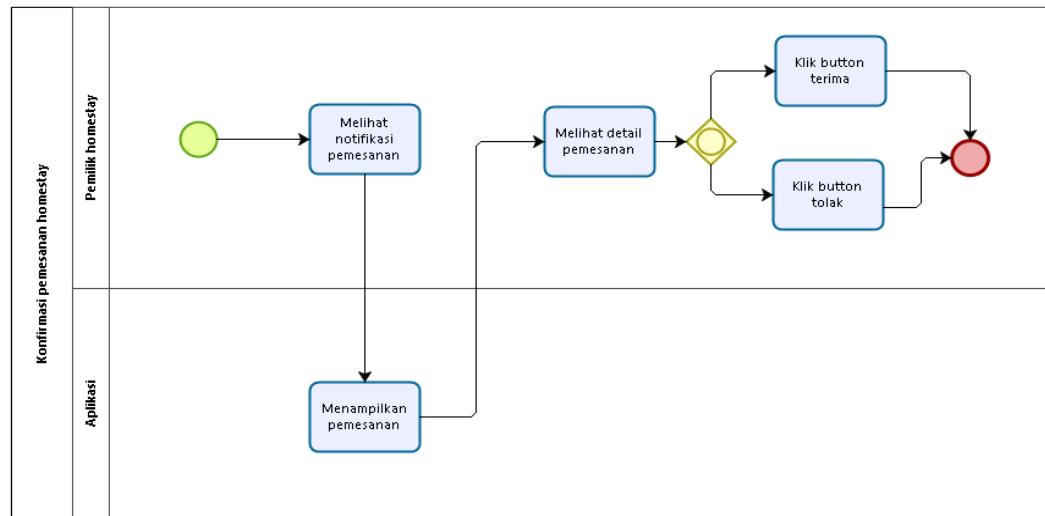
5. Mengelola Data Kamar



Gambar 8 Bisnis Proses Mengelola Data Kamar

Gambar 8 ini menjelaskan tentang proses pemilik *homestay* mengelola data kamar. Pemilik *homestay* akan mengelola data kamar yang dimilikinya. Pemilik *homestay* dapat melakukan *create, update, delete*.

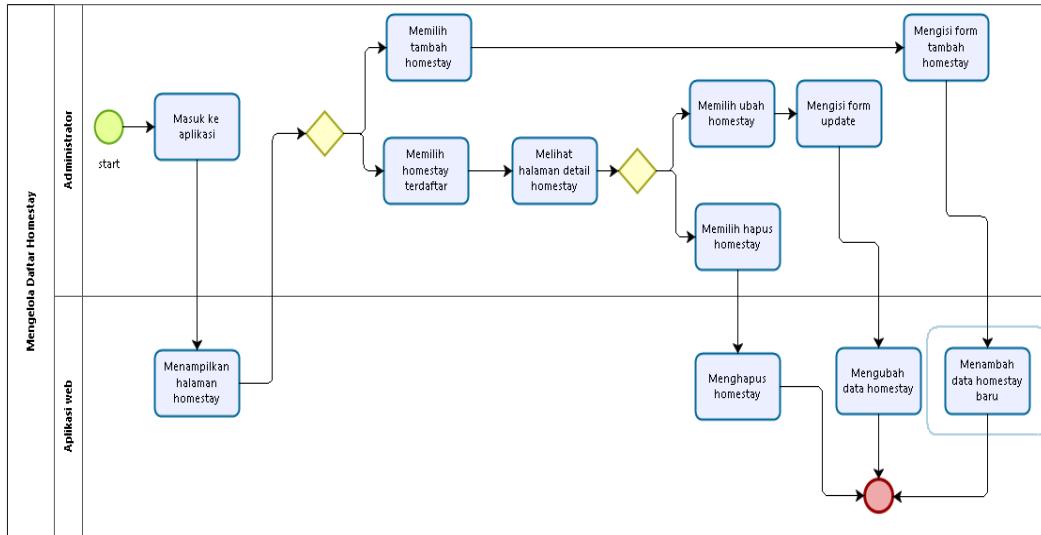
6. Konfirmasi pemesanan *homestay*



Gambar 9 Bisnis Proses Konfirmasi Pemesanan *Homestay*

Gambar 9 ini menjelaskan tentang proses pemilik *homestay* melakukan konfirmasi pemesanan *homestay*.

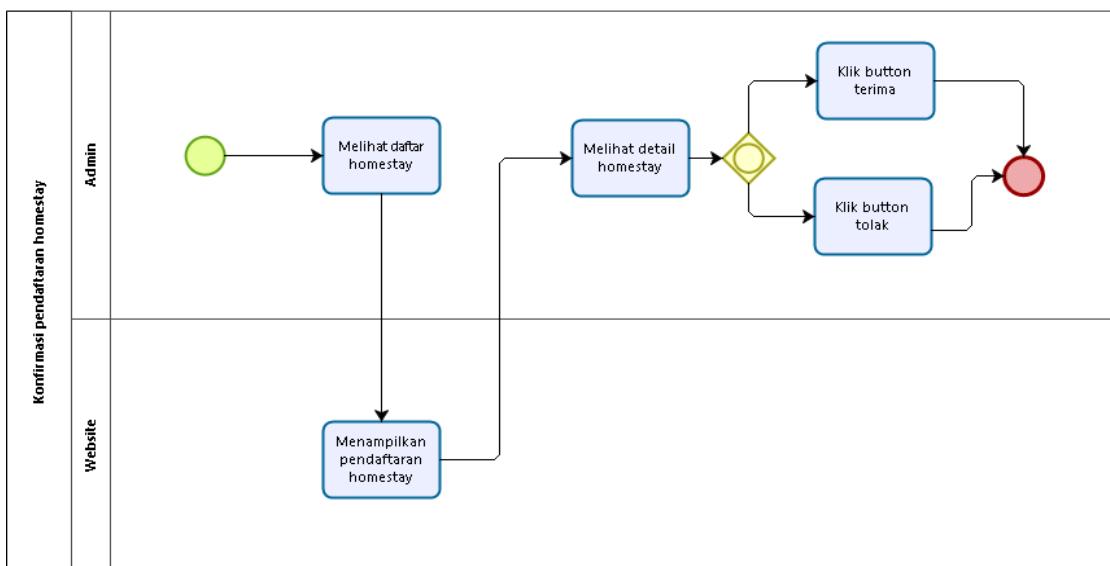
7. Mengelola daftar homestay



Gambar 10 Bisnis Proses Mengelola Daftar Homestay

Gambar 10 ini menjelaskan admin mengelola daftar homestay daerah Toba Samosir. Dimana admin dapat melakukan *create, update, delete homestay*.

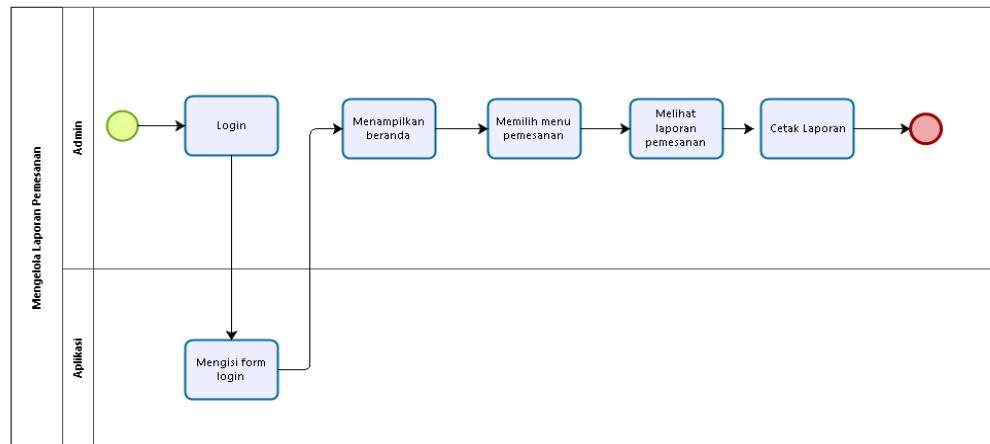
8. Kofirmasi pendaftaran homestay



Gambar 11 Bisnis Proses Konfirmasi Pendaftaran Homestay

Gambar 11 ini menjelaskan admin melakukan konfirmasi pendaftaran homestay.

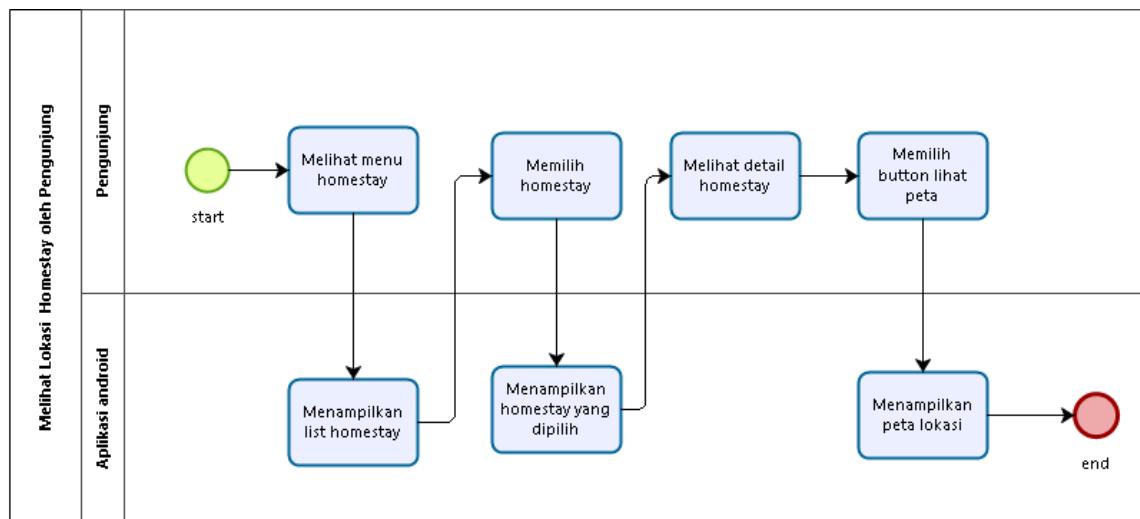
9. Mengelola laporan pemesanan *homestay*



Gambar 12 Bisnis Proses Mengelola Laporan Pemesanan

Gambar 12 ini menjelaskan tentang proses admin mengelola laporan pemesanan *homestay*. Admin akan mengelola laporan pemesanan *homestay* yang dimilikinya. Pemilik *homestay* dapat mencetak laporan pemesanan *homestay*.

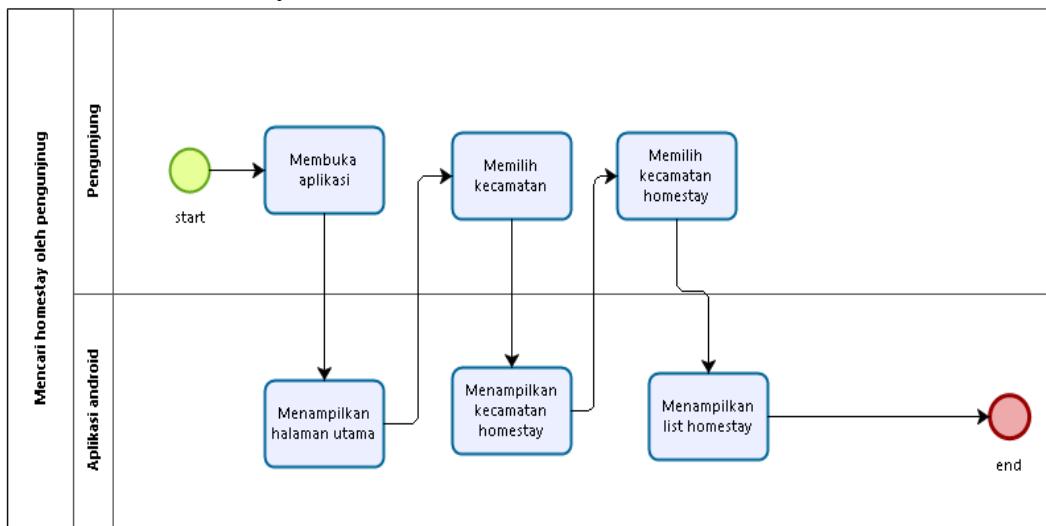
10. Melihat lokasi *homestay*



Gambar 13 Bisnis Proses Melihat Lokasi

Gambar 13 menjelaskan bagaimana pengunjung dapat melihat lokasi *homestay*.

11. Mencari Homestay berdasarkan kecamatan



Gambar 14 Bisnis Proses Mencari Homestay

3.2.1.3 User Characteristics

Berikut adalah karakteristik *user* aplikasi *homestay* daerah Toba Samosir:

1. User Group 1

| | |
|------------------|---|
| User | : Admin |
| Role | : Mengelola daftar <i>homestay</i> (<i>create, update, delete</i>), konfirmasi pendaftaran <i>homestay</i> , konfirmasi pemesanan <i>homestay</i> |
| Prerequisite | : <i>Login</i> |
| Task Description | : Mengelola daftar <i>homestay</i> |

2. User Group 2

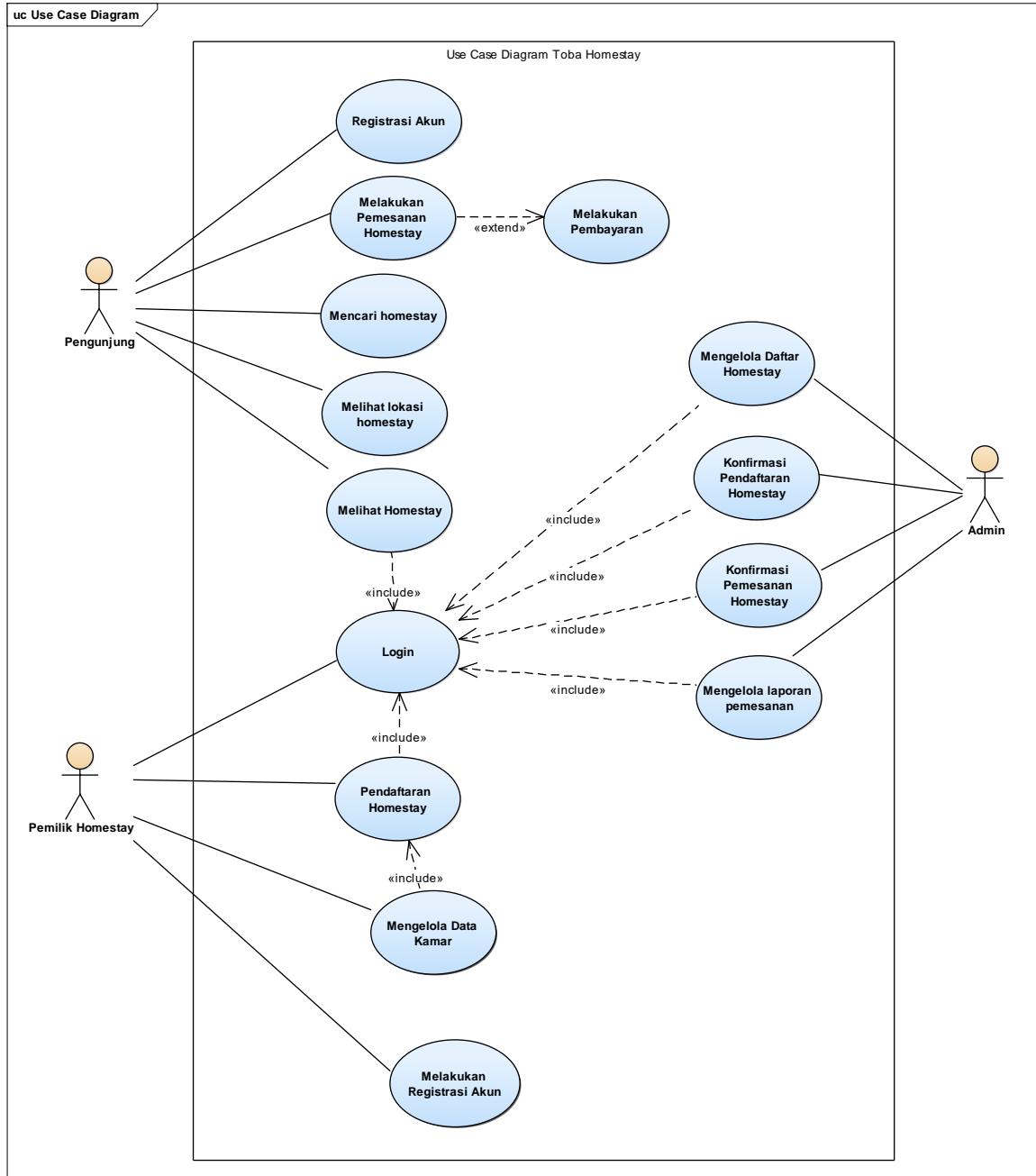
| | |
|------------------|--|
| User | : Pemilik <i>homestay</i> |
| Role | : Mengelola data <i>homestay</i> , laporan pemesanan <i>homestay</i> |
| Prerequisite | : <i>Login</i> |
| Task Description | : Mengelola data <i>homestay</i> yang dimilikinya |

3. User Group 3

| | |
|------------------|----------------------------------|
| User | : Pengunjung |
| Role | : Memesan <i>homestay</i> |
| Prerequisite | : <i>Login</i> dan memiliki akun |
| Task Description | : Pemesanan <i>homestay</i> |

3.2.1.4 Use Case Diagram

Pada subbab ini akan digambarkan fungsi yang dapat dilakukan *user* dalam aplikasi yang akan dibangun.



Gambar 15 Use Case Diagram

3.2.1.5 Use Case Scenario

Pada subbab ini menjelaskan mengenai use case scenario fungsi utama aplikasi yang akan dibangun

3.2.1.5.1 Use Case Scenario Login

Tabel 1 Use Case Scenario Login

| | | |
|----------------------|---|---|
| Use Case Id | UCD1 | |
| Use Case Name | <i>Login</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas <i>login</i> | |
| Actor | <i>Admin</i> , pemilik <i>homestay</i> , pengunjung | |
| Precondition | <i>Admin</i> , pemilik <i>homestay</i> , pengunjung telah memiliki <i>username</i> dan <i>password</i> | |
| Included Use Case | - | |
| Basic Flow of Events | <i>User Action</i> | <i>System response</i> |
| | 1. Membuka aplikasi | |
| | | 2. Sistem menampilkan <i>form login</i> |
| | 3. Memasukkan <i>username</i> dan <i>password</i> dengan nilai benar, serta melakukan <i>action login</i> | |
| | | 4. Memvalidasi <i>username</i> dan <i>password user</i> |
| | 5. Masuk ke aplikasi | |
| | | 6. Menampilkan halaman <i>home</i> |
| Error Flow | <i>User Action</i> | <i>System Response</i> |

| | | |
|----------------|---|--|
| | 1. Memasukkan <i>username</i> dan <i>password</i> dengan nilai salah, serta melakukan <i>action login</i> | |
| | 2. Memvalidasi <i>username</i> dan <i>password user</i> | |
| | 3. Menerima notifikasi <i>login</i> gagal, dan melakukan <i>login kembali</i> | |
| Post Condition | Masuk ke halaman halaman utama | |

3.2.1.5.2 Use Case Scenario Registrasi Akun

Tabel 2 Use Case Scenario Registrasi Akun

| | | |
|----------------------|---|-------------------------------------|
| Use Case Id | UCD2 | |
| Use Case Name | <i>Registrasi Akun</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas <i>Registrasi Akun</i> | |
| Actor | Pemilik <i>homestay</i> , pengunjung | |
| Precondition | Pemilik <i>homestay</i> , pengunjung telah memiliki <i>username</i> dan <i>password</i> | |
| Included Use Case | - | |
| Basic Flow of Events | <i>User Action</i> | <i>System response</i> |
| | 1. Membuka aplikasi | |
| | | 2. Sistem menampilkan halaman utama |
| | 3. Memilih tombol register | |
| | | 4. Menampilkan <i>form register</i> |
| | 5. Mengisi form register | |

| | | |
|----------------|--|-------------------------------------|
| | | 6. Menampilkan halaman <i>login</i> |
| Error Flow | <i>User Action</i> | <i>System Response</i> |
| | 1. Tidak mengisi form register dengan baik dan benar | |
| | | 2. Memvalidasi data <i>user</i> |
| | 3. Menerima notifikasi <i>registrasi gagal</i> , dan melakukan <i>register kembali</i> | |
| Post Condition | Masuk ke halaman halaman utama | |

3.2.1.5.3 Use Case Scenario Pendaftaran *Homestay*

Tabel 3 Use Case Scenario Pendaftaran *Homestay*

| | | |
|----------------------|--|-------------------------------------|
| Use Case Id | UCD3 | |
| Use Case Name | Pendaftaran <i>homestay</i> | |
| Brief Description | Use case ini menjelaskan aktivitas Pendaftaran <i>homestay</i> | |
| Actor | Pemilik <i>homestay</i> | |
| Precondition | Pemilik <i>homestay</i> telah memiliki <i>username</i> dan <i>password</i> | |
| Included Use Case | - | |
| Basic Flow of Events | <i>User Action</i> | <i>System response</i> |
| | 1. Membuka aplikasi | |
| | | 2. Sistem menampilkan halaman utama |
| | 3. Memilih tombol daftar | |

| | | |
|----------------|---|---|
| | | 4. Menampilkan form Pendaftaran <i>homestay</i> |
| | 5. Mengisi form Pendaftaran <i>homestay</i> | |
| | | 6. Menampilkan halaman utama |
| Error Flow | <i>User Action</i> | <i>System Response</i> |
| | 1. Tidak mengisi form register dengan baik dan benar | |
| | | 2. Memvalidasi data Pendaftaran <i>homestay</i> |
| | 3. Menerima notifikasi Pendaftaran <i>homestay</i> gagal, dan melakukan Pendaftaran kembali | |
| Post Condition | Masuk ke halaman halaman utama | |

3.2.1.5.4 Use Case Scenario Konfirmasi Pemesanan *Homestay*

Tabel 4 Use Case Scenario Konfirmasi Pemesanan *Homestay*

| | | |
|----------------------|--|------------------------|
| Use Case Id | UCD4 | |
| Use Case Name | Konfirmasi Pemesanan | |
| Brief Description | <i>Use case</i> ini menjelaskan fitur konfirmasi pemesanan <i>homestay</i> | |
| Actor | Admin | |
| Precondition | Admin telah <i>login</i> ke dalam sistem | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <i>User Action</i> | <i>System Response</i> |

| | | |
|----------------------------|-----------------------------------|--|
| | 1. Memilih menu pemesanan | |
| | | 2. Menampilkan daftar pemesanan |
| | 3. Melihat detail pemesanan | |
| | | 4. Menampilkan detail pemesanan |
| | 5. Memilih tombol <i>accepted</i> | |
| | | 6. Mengubah status permintaan pesanan menjadi <i>accepted</i> dan mengirim notifikasi <i>accepted</i> kepada <i>user</i> |
| Alternative flow of events | <i>User Action</i> | <i>System Response</i> |
| | 1. Kamar tidak tersedia | |
| | 2. Memilih tombol <i>declined</i> | |
| | | 3. Mengubah status permintaan pesanan menjadi <i>declined</i> dan mengirim notifikasi ditolak kepada pengunjung. |
| Post Condition | pemesanan berhasil dikonfirmasi | |

3.2.1.5.5 Use Case Scenario Konfirmasi Pendaftaran *Homestay*

Tabel 5 Use Case Scenario Konfirmasi Pendaftaran *Homestay*

| | |
|-------------------|--|
| Use Case Id | UCD5 |
| Use Case Name | Mengelola Konfirmasi Pendaftaran <i>Homestay</i> |
| Brief Description | <i>Use case</i> ini menjelaskan fitur konfirmasi pendaftaran <i>homestay</i> |
| Actor | Admin |

| | | |
|----------------------------|---|---|
| Precondition | Admin telah <i>login</i> ke dalam sistem | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <i>User Action</i> | <i>System Response</i> |
| | 1. Memilih menu pendaftaran <i>homestay</i> | |
| | | 2. Menampilkan daftar pendaftaran <i>homestay</i> . |
| | 3. Melihat detail pendaftaran <i>homestay</i> | |
| | | 4. Menampilkan detail pendaftaran <i>homestay</i> |
| | 5. Memilih tombol <i>accepted</i> | |
| | | 6. Mengubah status permintaan pesanan menjadi <i>accepted</i> dan mengirim notifikasi <i>accepted</i> kepada <i>user</i> |
| Alternative flow of events | <i>User Action</i> | <i>System Response</i> |
| | 1. Tidak sesuai dengan kriteria | |
| | 2. Memilih tombol <i>declined</i> | |
| | | 3. Mengubah status permintaan daftar <i>homestay</i> menjadi <i>declined</i> dan mengirim notifikasi ditolak kepada pengunjung. |

| | |
|----------------|-----------------------------------|
| Post Condition | pendaftaran berhasil dikonfirmasi |
|----------------|-----------------------------------|

3.2.1.5.6 Use Case Scenario Mengelola Kamar *Homestay*

1. Fungsi Menambah Data kamar *Homestay*

Tabel 6 Use Case Scenario Menambah Data Kamar *Homestay*

| | | |
|----------------------|--|-------------------------------|
| Use Case Id | UCD6 | |
| Use Case Name | Mengelola Data <i>Homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan penambahan data data <i>homestay</i> | |
| Actor | Pemilik <i>homestay</i> | |
| Precondition | Pemilik <i>homestay</i> telah <i>login</i> ke dalam sistem | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <p><i>User Action</i></p> <p>1. Memilih menu <i>homestay</i></p> <p>2. Menampilkan <i>homestay</i></p> <p>3. Memilih tombol <i>add</i></p> <p>4. Menampilkan <i>form add</i> kamar <i>homestay</i></p> <p>5. Mengisi <i>form add</i> kamar <i>homestay</i></p> <p>6. Memilih tombol <i>save</i>.</p> <p>7. Menyimpan data yang ditambahkan</p> | <p><i>System Response</i></p> |

| Error Flow | User Action | System Response |
|----------------|--|--|
| | 1. Mengisi <i>form</i> dan pilih tombol <i>add</i> | |
| | | 2. Menampilkan pesan error di <i>form</i> tambah datas |
| Post Condition | Data kamar <i>homestay</i> berhasil ditambahkan | |

2. Fungsi Update Data Kamar *Homestay*

Tabel 7 Use Case Scenario Update Kamar *Homestay*

| Use Case Id | UCD7 | |
|----------------------|--|---------------------------------------|
| Use Case Name | Mengelola Data <i>Homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan <i>update</i> data <i>homestay</i> | |
| Actor | Pemilik <i>homestay</i> | |
| Precondition | Pemilik <i>homestay</i> telah <i>login</i> ke dalam aplikasi | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | User Action | System Response |
| | 1. Memilih menu <i>homestay</i> | |
| | | 2. Menampilkan <i>homestay</i> |
| | 3. Memilih <i>homestay</i> yang ingin diupdate | |
| | | 4. Menampilkan detail <i>homestay</i> |
| | 5. Melakukan update data <i>homestay</i> | |

| | | |
|----------------|---|---|
| | 6. Memilih tombol <i>save</i> . | |
| | | 7. Menyimpan data yang di <i>update</i> |
| Error Flow | <i>User Action</i> | <i>System Response</i> |
| | 1. Mengisi <i>form</i> dan pilih tombol <i>update</i> | |
| | | 2. Menampilkan pesan error di <i>form</i> tambah diatas |
| Post Condition | Data <i>homestay</i> berhasil di <i>update</i> | |

3. Fungsi Delete Data Kamar *Homestay*

Tabel 8 Use Case Scenario Delete Data Kamar *Homestay*

| | | |
|----------------------|--|--------------------------------------|
| <i>Use case Id</i> | UCD8 | |
| <i>Use case Name</i> | Mengelola Data Kamar <i>Homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas menghapus data informasi <i>homestay</i> | |
| Actor | Pemilik <i>homestay</i> | |
| Precondition | Pemilik <i>homestay</i> telah <i>login</i> ke dalam sistem | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <i>User Action</i> | <i>System Response</i> |
| | 1. Memilih menu <i>homestay</i> | |
| | | 2. Menampilkan kamar <i>homestay</i> |
| | 3. Memilih tombol <i>delete</i> . | |

| | | |
|----------------|--|-----------------------------|
| | | 4. Menampilkan <i>alert</i> |
| | 5. Menyetujui menghapus data <i>homestay</i> | |
| | | 6. Data berhasil dihapus |
| Post Condition | Data <i>homestay</i> berhasil dihapus | |

3.2.1.5.7 Use Case Scenario Mengelola Laporan Pemesanan *Homestay*

Tabel 9 Use Case Scenario Cetak Laporan Pemesanan *Homestay*

| | | |
|----------------------|--|---|
| Use Case Id | UCD9 | |
| Use Case Name | Mengelola Informasi Laporan Pemesanan | |
| Brief Description | <i>Use case</i> ini menjelaskan cetak data informasi Laporan Pemesanan | |
| Actor | <i>Admin</i> | |
| Precondition | <i>Admin</i> telah <i>login</i> ke dalam aplikasi | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <p><i>User Action</i></p> <p>1. Memilih menu Laporan Pemesanan</p> | <p><i>System Response</i></p> <p>2. Menampilkan Laporan Pemesanan</p> |
| | <p>3. Memilih Laporan Pemesanan yang ingin dicetak</p> | |

| | | |
|----------------|--|--|
| | | 4. Menampilkan detail Laporan Pemesanan |
| | 5. Mencetak data Laporan Pemesanan | |
| | 6. Memilih tombol cetak. | |
| | | 7. Mengubah laporan kedalam pdf dan cetak laporan |
| Error Flow | <i>User Action</i> | <i>System Response</i> |
| | 1. Mencetak <i>form</i> dan pilih tombol cetak | |
| | | 2. Menampilkan pesan error laporan tidak dapat dicetak |
| Post Condition | Data Laporan Pemesanan berhasil dicetak | |

3.2.1.5.8 Use Case Scenario Mengelola Daftar *Homestay*

1. Fungsi tambah *homestay*

Tabel 10 Use Case Scenario Tambah *Homestay*

| | | |
|----------------------|---|------------------------|
| Use Case Id | UCD10 | |
| Use Case Name | Mengelola Informasi <i>homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan penambahan data <i>informasi homestay</i> | |
| Actor | <i>Admin</i> | |
| Precondition | <i>Admin</i> telah <i>login</i> ke dalam sistem | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <i>User Action</i> | <i>System Response</i> |

| | | |
|----------------|--|--|
| | 1. Memilih menu <i>homestay</i> . | |
| | 2. Menampilkan <i>homestay</i> | |
| | 3. Memilih tombol <i>add</i> | |
| | 4. Menampilkan <i>form add homestay</i> . | |
| | 5. Mengisi <i>form add homestay</i> . | |
| | 6. Memilih tombol <i>save</i> . | |
| | | 7. Menyimpan data yang ditambahkan |
| Error Flow | <i>User Action</i> | <i>System Response</i> |
| | 1. Mengisi <i>form</i> dan pilih tombol <i>add</i> | |
| | | 2. Menampilkan pesan error di <i>form tambah datas</i> |
| Post Condition | Data <i>homestay</i> berhasil ditambahkan | |

2. Fungsi delete *homestay*

Tabel 11 Use case scenario delete *homestay*

| <i>Use case Id</i> | UCD11 | |
|----------------------|--|-------------------------------------|
| <i>Use case Name</i> | Mengelola Informasi <i>homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas menghapus data informasi <i>homestay</i> | |
| Actor | Admin | |
| Precondition | Admin dan pemilik hotel telah <i>login</i> ke dalam sistem | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | <i>User Action</i> | <i>System Response</i> |
| | 1. Memilih menu <i>homestay</i> | |
| | | 2. Menampilkan list <i>homestay</i> |
| | 3. Memilih tombol <i>delete</i> . | |
| | | 4. Menampilkan <i>alert</i> |
| | 5. Menyetujui menghapus data <i>homestay</i> | |
| | | 6. Data berhasil dihapus |
| Post Condition | Data <i>homestay</i> berhasil dihapus | |

3. Fungsi Update *homestay*

Tabel 12 Use Case Scenario Update *homestay*

| Use Case Id | UCD12 | |
|----------------------|--|---------------------------------------|
| Use Case Name | Mengelola Informasi <i>homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan <i>update</i> data informasi <i>homestay</i> | |
| Actor | <i>Admin</i> | |
| Precondition | <i>Admin</i> telah <i>login</i> ke dalam aplikasi | |
| Included Use Case | <i>Login</i> | |
| Basic Flow of Events | User Action | System Response |
| | 1. Memilih menu <i>homestay</i> . | |
| | | 2. Menampilkan <i>homestay</i> |
| | 3. Memilih <i>homestay</i> yang ingin diupdate | |
| | | 4. Menampilkan detail <i>homestay</i> |
| | 5. Mengupdate data <i>homestay</i> | |
| | 6. Memilih tombol <i>save</i> . | |
| Error Flow | | 7. Menyimpan data yang diupdate |
| | User Action | System Response |

| | | |
|----------------|---|--|
| | 1. Mengisi <i>form</i> dan pilih tombol <i>update</i> | |
| | 2. Menampilkan pesan error di <i>form</i> tambah diatas | |
| Post Condition | Data <i>homestay</i> berhasil di <i>update</i> | |

3.2.1.5.9 Use Case Scenario Melihat Homestay

Tabel 13 Use Case Scenario Melihat Homestay

| | | |
|----------------------|--|--|
| Use Case Id | UCD13 | |
| Use Case Name | Melihat informasi <i>homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas melihat informasi <i>homestay</i> | |
| Actor | Pengunjung | |
| Precondition | - | |
| Included Use Case | - | |
| Basic Flow of Events | <p><i>User Action</i></p> <ol style="list-style-type: none"> 1. Membuka Aplikasi <i>homestay</i> Toba Samosir 2. Memilih menu <i>homestay</i> 3. Memilih salah satu <i>homestay</i> | <p><i>System Response</i></p> <ol style="list-style-type: none"> 1. Menampilkan home 2. Menampilkan list <i>homestay</i> |
| | 5. Memilih salah satu <i>homestay</i> | |

| | | |
|----------------|------------------------------------|---------------------------------------|
| | | 6. Menampilkan <i>detail homestay</i> |
| Post Condition | Detail <i>homestay</i> ditampilkan | |

3.2.1.5.10 Use Case Scenario Mencari Homestay

Tabel 14 Use Case Scenario Mencari Homestay

| | | |
|----------------------|---|---|
| Use Case Id | UCD14 | |
| Use Case Name | Mencari Homestay | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas cari homestay | |
| Actor | Pengunjung | |
| Precondition | - | |
| Included Use Case | - | |
| Basic Flow of Events | <p><i>User Action</i></p> <p>5. Membuka Aplikasi <i>homestay</i> Toba Samosir</p> <p>7. Memilih kecamatan homestay</p> <p>9. Memilih homestay</p> | <p><i>System Response</i></p> <p>6. Menampilkan home</p> <p>8. Menampilkan list <i>homestay</i></p> <p>6. Menampilkan detail homestay</p> |
| Post Condition | Berhasil melakukan pencarian <i>homestay</i> | |

3.2.1.5.12 Use Case Scenario Melihat Lokasi Homestay

Tabel 15 Use Case Scenario Melihat Lokasi Homestay

| | | |
|----------------------|---|---|
| Use Case Id | UCD15 | |
| Use Case Name | Melihat Lokasi <i>Homestay</i> | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas melihat lokasi <i>homestay</i> | |
| Actor | Pengunjung | |
| Precondition | - | |
| Included Use Case | - | |
| Basic Flow of Events | <p style="text-align: center;"><i>User Action</i></p> <p>1. Membuka Aplikasi <i>homestay</i> Toba Samosir</p> <p>2. Menampilkan home</p> <p>3. Memilih homestay</p> <p>4. Menampilkan detail <i>homestay</i></p> <p>5. Memilih button lihat peta</p> <p>6. Menampilkan lokasi <i>homestay</i></p> | <p style="text-align: center;"><i>System Response</i></p> |
| Post Condition | Berhasil menampilkan lokasi <i>homestay</i> | |

3.2.1.5.13 Use Case Scenario Melakukan Pemesanan

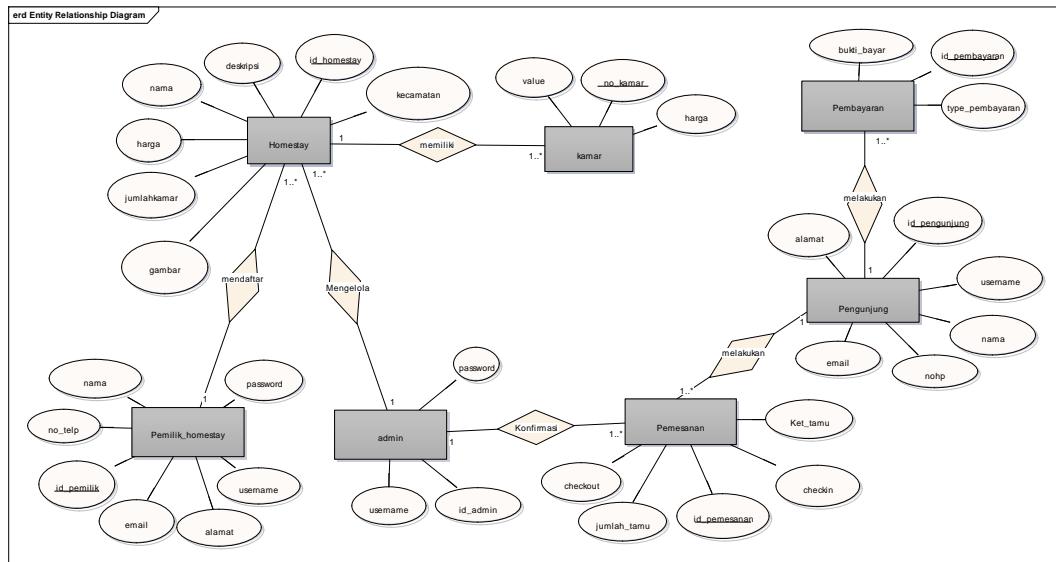
Tabel 16 Use case scenario melakukan pemesanan

| | | |
|----------------------|---|---|
| Use Case Id | UCD16 | |
| Use Case Name | Melakukan pemesanan homestay | |
| Brief Description | <i>Use case</i> ini menjelaskan aktivitas melakukan pemesanan homestay | |
| Actor | Pengunjung | |
| Precondition | - | |
| Included Use Case | - | |
| Basic Flow of Events | <p style="text-align: center;"><i>User Action</i></p> <p>1. Membuka Aplikasi <i>homestay Toba Samosir</i></p> <p>2. Memilih homestay</p> <p>3. Memilih button pesan</p> <p>4. Mengisi form pemesanan</p> <p>5. Mengisi form pemesanan</p> <p>6. Menampilkan detail <i>homestay</i></p> <p>7. Menampilkan form pemesanan</p> | <p style="text-align: center;"><i>System Response</i></p> |
| Post Condition | Berhasil melakukan pemesanan | |

3.2.1.6 Data Requirement

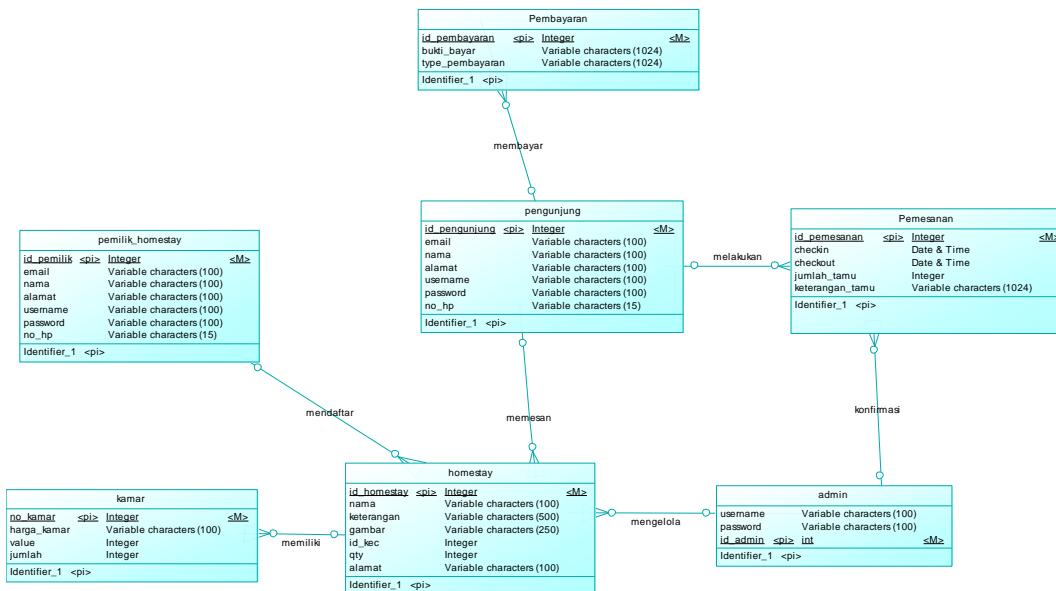
3.2.1.6.1 ERD

Pada subbab ini penulis menggunakan *Entity Relationship Diagram* (ERD) dengan notasi Chen yaitu untuk menggambarkan relasi antar entity dalam basis data pada aplikasi *homestay*.



Gambar 16 ERD

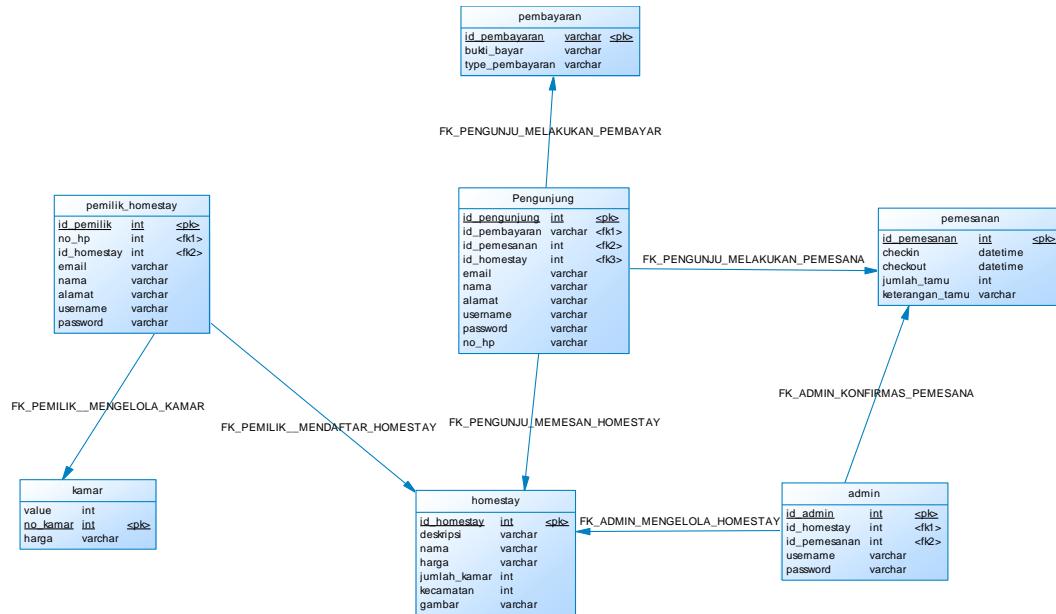
3.2.1.6.2 Conceptual Data Model



Gambar 17 Conceptual Data Model

Conceptual Data Model adalah perancangan basis data yang berdasarkan pengumpulan data dan analisis yang digambarkan secara *logic* berdasarkan hasil analisa. Untuk topik pengembangan aplikasi *homestay* ini disimpulkan entity-entity yang digunakan adalah *homestay*, kamar, pengunjung, pemilik *homestay*, admin, pemesanan, dan pembayaran.

3.2.1.6.3 Physical Data Model



Gambar 18 Physical Data Model

Physical Data Model merupakan gambaran fisik dari database yang akan dibuat dengan mempertimbangkan DBMS yang akan digunakan.

3.2.1.6.4 Struktur Database Aplikasi

1. Tabel *homestay*

Digunakan untuk menyimpan data *homestay*.

Nama tabel: *homestay*

Primary key: id_*homestay*

Tabel 17 Tipe Data Homestay

| Nama field | Tipe | Keterangan |
|---------------------|---------------|--------------------------|
| Id_ <i>homestay</i> | int | Id tabel <i>homestay</i> |
| nama | Varchar (100) | Nama <i>homestay</i> |
| kecamatan | Varchar (70) | Lokasi <i>homestay</i> |

| | | |
|-----------|--------------|-----------------------|
| deskripsi | Varchar(255) | Deskripsi homestay |
| foto | Varchar(255) | Foto homestay |
| fasilitas | Varchar(200) | Fasilitas homestay |
| qty | int | Jumlah kamar homestay |
| Harga | Int | Harga homestay |

2. Tabel pemesanan_homestay

Digunakan untuk menyimpan data pemesanan.

Nama tabel: pemesanan_homestay

Primary key: id_pemesanan

Tabel 18 Tipe data pemesanan

| Nama field | Tipe | Keterangan |
|-----------------|--------------|--------------------|
| Id_pemesanan | int | Id tabel pemesanan |
| checkin | datetime | Checkin |
| checkout | datetime | Checkout |
| Jumlah_tamu | int | Jumlah tamu |
| Keterangan_tamu | Varchar(100) | Keterangan tamu |

3. Tabel kamar

Digunakan untuk menyimpan data kamar.

Nama tabel: kamar

Primary key: no_kamar

Tabel 19 Type Data Kamar

| Nama field | Tipe | Keterangan |
|-------------|-------------|----------------|
| Harga_kamar | Varchar(70) | Harga kamar |
| No_kamar | int | Id tabel kamar |
| value | int | Harga kamar |
| jumlah | int | Jumlah kamar |

4. Tabel pemilik_homestay

Digunakan untuk menyimpan data pemilik_homestay

Nama tabel: pemilik_homestay

Primary key: pemilik_homestay

Tabel 20 Type Data pemilik_homestay

| Nama field | Tipe | Keterangan |
|------------|--------------|-----------------------|
| Id_pemilik | int | Id tabel <i>users</i> |
| Nama | Varchar (70) | Nama <i>users</i> |
| email | Varchar (90) | email <i>users</i> |
| alamat | Varchar(255) | alamat <i>users</i> |
| role | int | Role <i>users</i> |
| nohp | varchar | No hp <i>users</i> |
| password | varchar | Password <i>users</i> |

5. Tabel Admin

Digunakan untuk menyimpan data admin

Nama tabel: pendaftaran *Primary key*: id_admin

Tabel 21 Type Data Admin

| Nama field | Tipe | Keterangan |
|--------------|--------------|--------------------|
| Id_admin | int | Id tabel admin |
| Id_homestay | int | Id <i>homestay</i> |
| Id_pemesanan | int | Id pemesanan |
| username | Varchar(100) | Username homestay |
| password | Varchar(90) | Password homestay |

6. Tabel pengunjung

Digunakan untuk menyimpan data pengunjung

Nama tabel: pengunjung

Primary key: id_pengunjung

Tabel 22 Type Data Pengunjung

| Nama field | Tipe | Keterangan |
|---------------|--------------|-----------------------|
| Id_pengunjung | int | Id tabel <i>users</i> |
| Nama | Varchar (70) | Nama <i>users</i> |
| email | Varchar (90) | email <i>users</i> |
| alamat | Varchar(255) | alamat <i>users</i> |
| role | int | Role <i>users</i> |
| nohp | varchar | No hp <i>users</i> |
| password | varchar | Password <i>users</i> |

7. Tabel Pembayaran

Digunakan untuk menyimpan data pembayaran.

Nama tabel: pembayaran

Primary key: id_pembayaran

| Nama field | Tipe | Keterangan |
|------------------|--------------|--------------------------|
| Id_pembayaran | int | Id pembayaran |
| Id_pemesanan | int | Foreign key id pemesanan |
| Bukti_pembayaran | Varchar(200) | Bukti pembayaran |
| Type_pembayaran | Varchar(200) | Type pembayaran |

BAB IV DESIGN

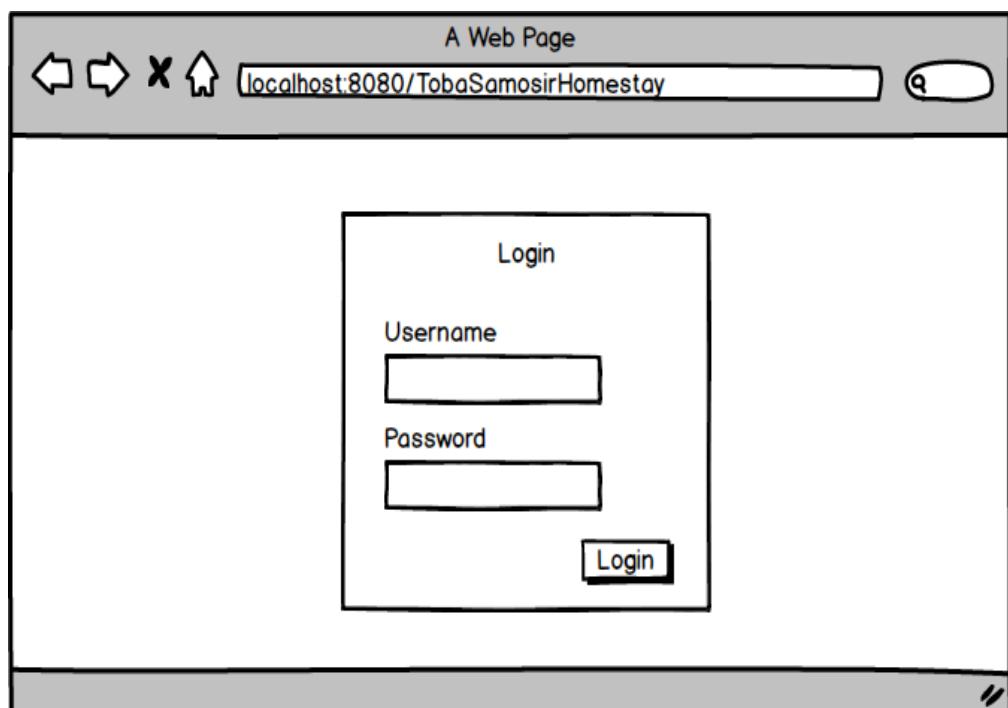
4.1 Design interface Low Fidelity

Pada sub bab ini dijelaskan bahwa penulis melakukan pertemuan dengan *user* untuk melihat *mockup wireframe*. Pada tahap ini penulis melakukan pengujian dan menunjukkan hasil *design* berupa *mockup wireframe* untuk menunjukkan kerangka dan *flowchart* aplikasi yang akan dibangun.

4.1.1 Design interface Admin

1. Design interface Login

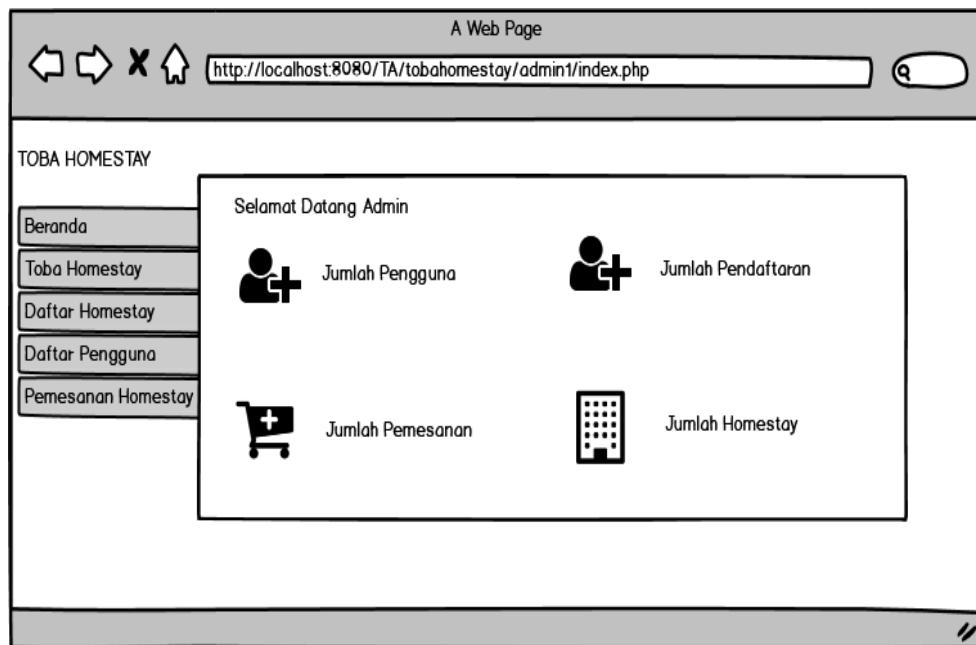
Berikut design interface *admin* untuk fungsi *login* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 19 Design interface Login

2. Design interface Home

Berikut design interface *admin* untuk menu *home* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 20 Design interface Home

3. Design interface Tambah Homestay

Berikut design interface *admin* untuk fungsi tambah *homestay* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

A Web Page
http://localhost:8080/TA/tobahomestay/admin1/tambah_homestay.php

TOBA HOMESTAY

Beranda
Toba Homestay
Daftar Homestay
Daftar Pengguna
Pemesanan Homestay

Tambah Homestay

Nama Homestay
[REDACTED]

Jumlah
[REDACTED]

Deskripsi
[REDACTED]

Kecamatan
[REDACTED]

Fasilitas
[REDACTED]

POI
[REDACTED]

Gambar 21 Design interface Tambah Homestay

4. Design interface Data Homestay

Berikut design interface *admin* untuk menu data *homestay* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan.

A Web Page
http://localhost:8080/TA/tobahomestay/admin1/data_homestay.php

TOBA HOMESTAY

Beranda
Toba Homestay
Daftar Homestay
Daftar Pengguna
Pemesanan Homestay

Data Homestay

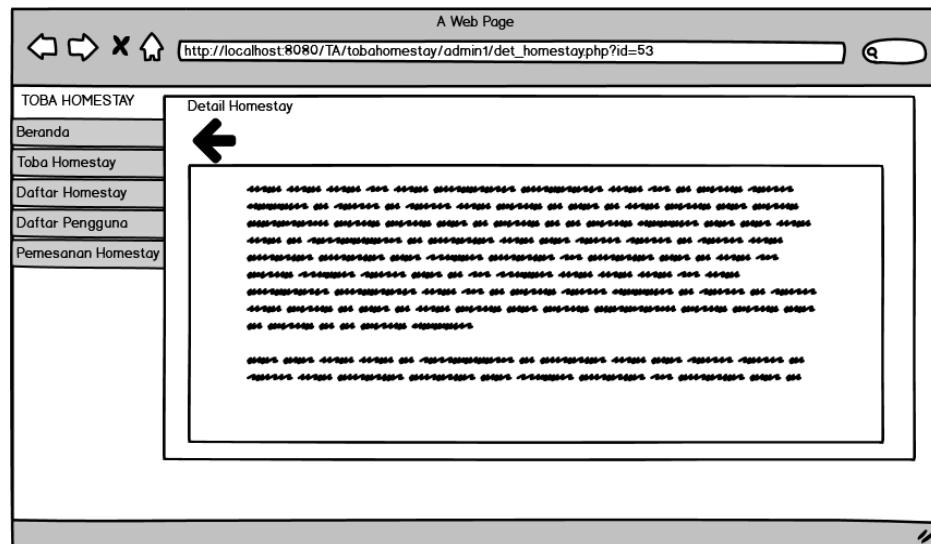
Tambah Homestay

| No | Id Homestay | Nama Homestay | Kecamatan | Jlh Kamar | Fasilitas | Alamat | Pol | Opsi |
|----|-------------|---------------|-----------|-----------|-----------|--------|-----|------|
| | | | | | | | | |

Gambar 22 Design interface Data Homestay

5. Design interface Detail Homestay

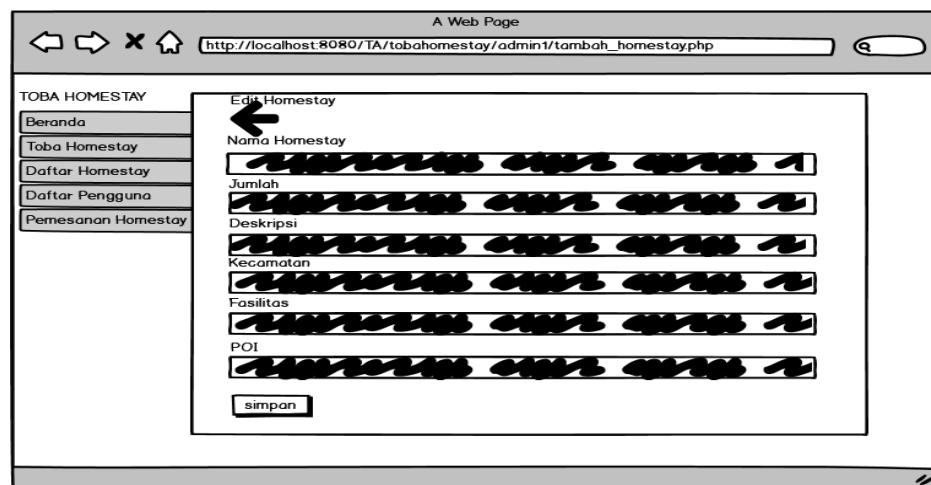
Berikut design interface *admin* untuk fungsi detail *homestay* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 23 Design interface Detail Homestay

6. Design interface Edit Homestay

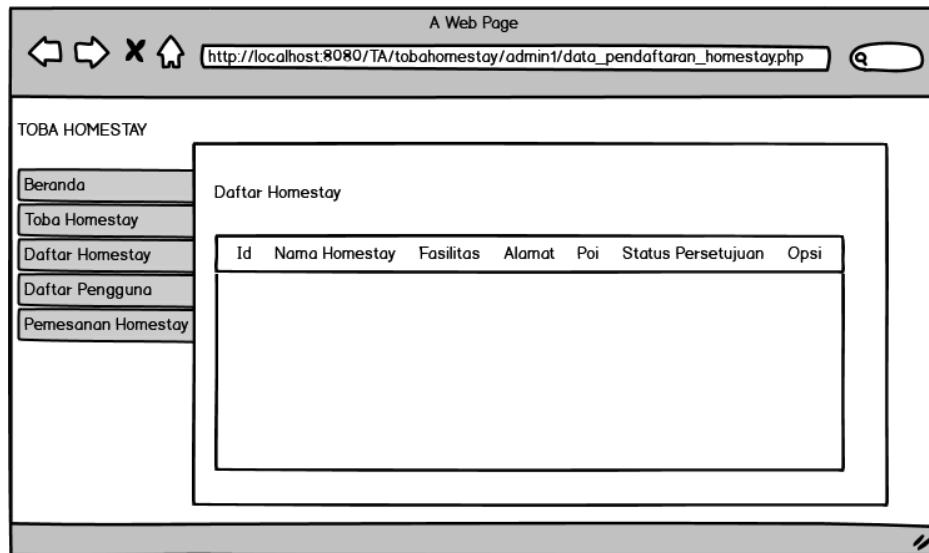
Berikut design interface *admin* untuk fungsi edit *homestay* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 24 Design interface Edit Homestay

7. Design interface Daftar Homestay

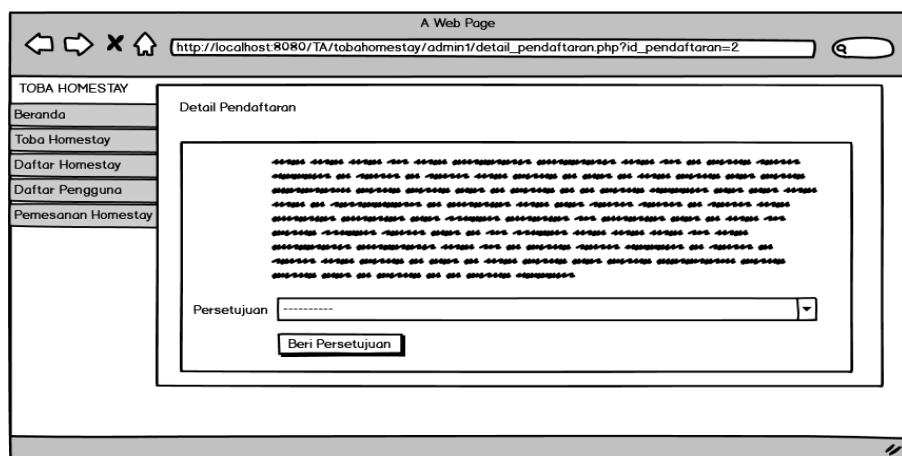
Berikut design interface *admin* untuk fungsi daftar *homestay* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 25 Design interface Pendaftaran Homestay

8. Design interface Detail Pendaftaran Homestay

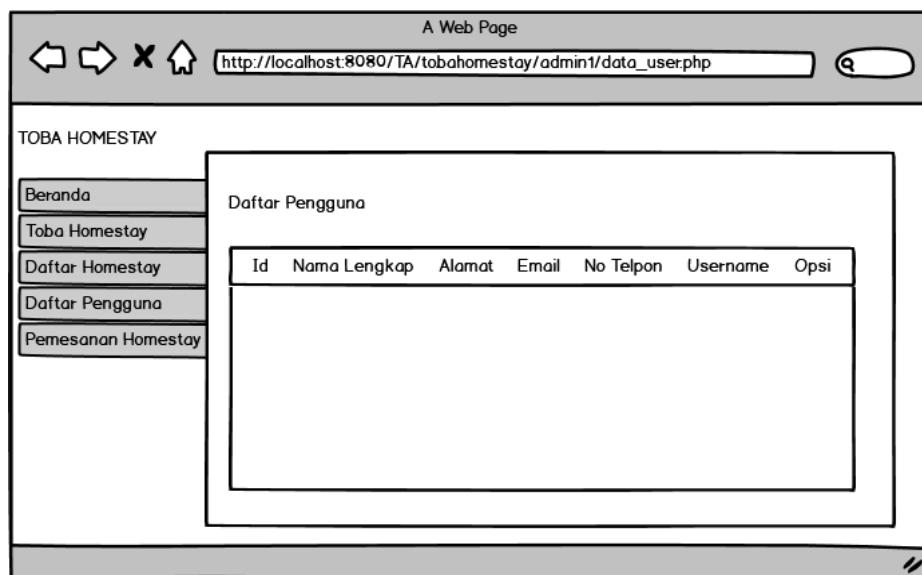
Berikut design interface *admin* untuk fungsi detail pendaftaran *homestay* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan.



Gambar 26 Design interface Detail Pendaftaran Homestay

9. Design interface Daftar Pengguna

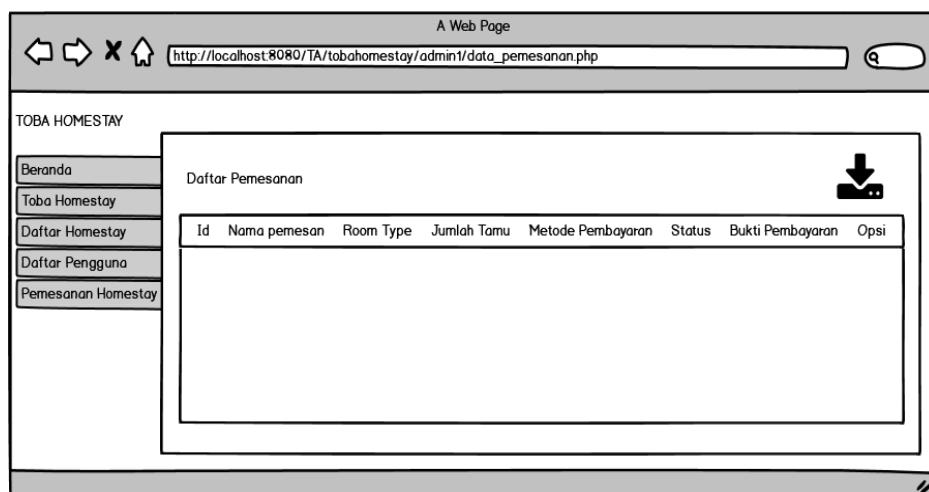
Berikut design interface *admin* untuk menu daftar pengguna yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 27 Design interface Daftar Pengguna

10. Design interface Pemesanan

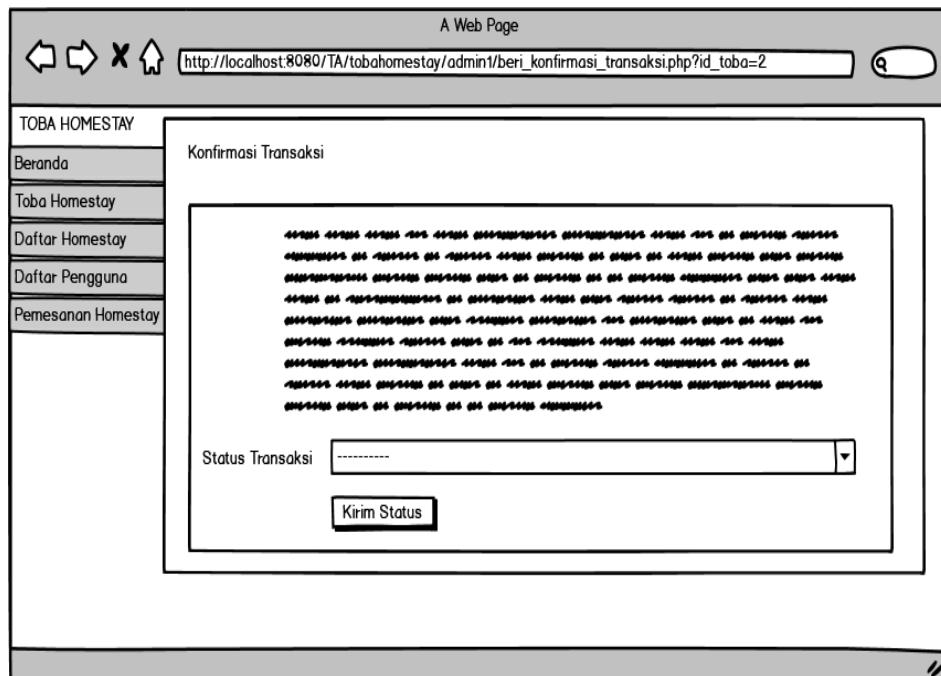
Berikut design interface *admin* untuk fungsi *login* yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 28 Design interface Pemesanan

11. Design interface Konfirmasi Pemesanan

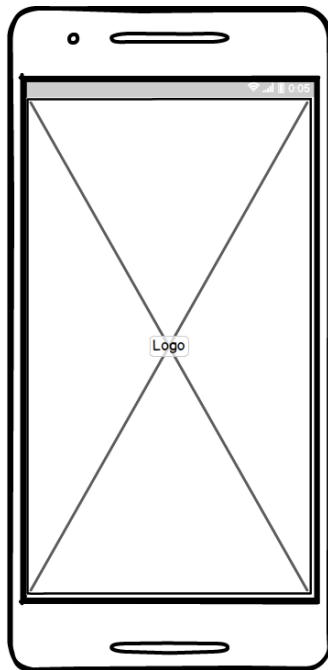
Berikut design interface *admin* untuk fungsi konfirmasi pemesanan yang berupa *design low fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



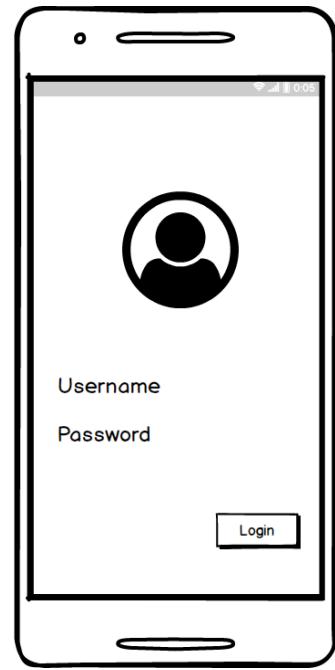
Gambar 29 *Design interface Konfirmasi Pemesanan*

4.1.2 Design interface Pengunjung

1. *Design interface* Splashscreen
 2. *Design interface* Login
- Berikut design interface splashscreen pengunjung
- Berikut design interface login pengunjung



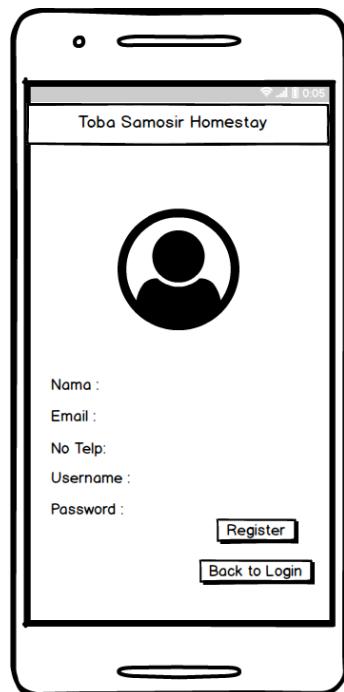
Gambar 30 *Design interface* Splashscreen



Gambar 31 *Design interface* Login

3. Design interface Register

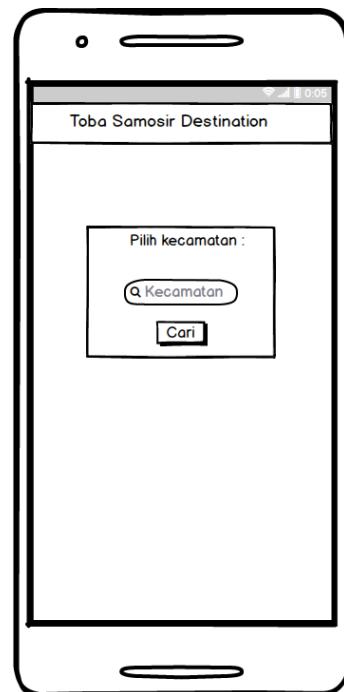
Berikut design interface register pengunjung



Gambar 32 Design interface Register

4. Design interface Home

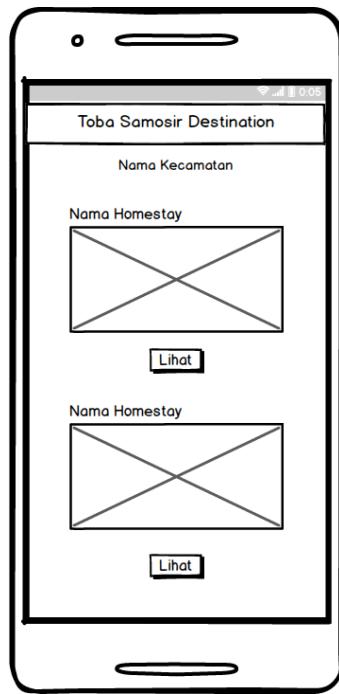
Berikut design interface home pengunjung



Gambar 33 Design interface Home

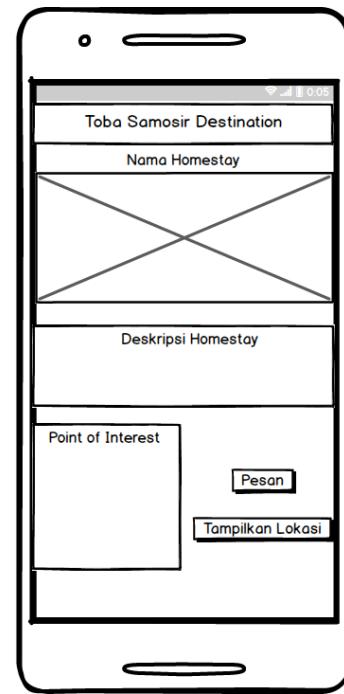
5. *Design interface List Homestay*

Berikut design interface list homestay pengunjung



6. *Design interface Detail Homestay*

Berikut design interface detail homestay pengunjung

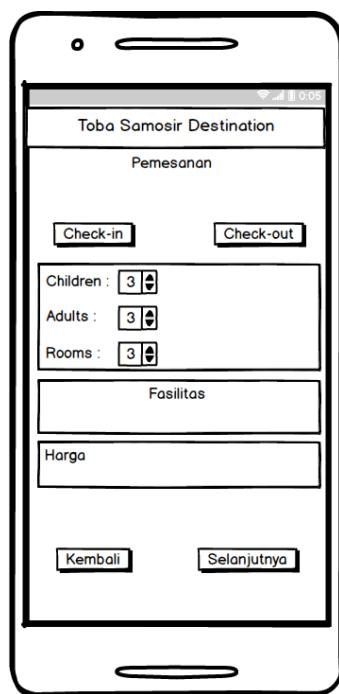


Gambar 34 Design interface List Homestay

Gambar 35 Design interface Detail Homestay

7. Design interface Pemesanan

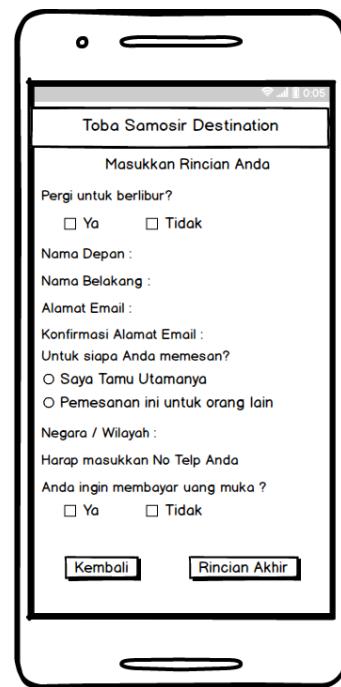
Berikut design interface pemesanan pengunjung



Gambar 36 Design interface Pemesanan

8. Design interface Rincian Pemesanan

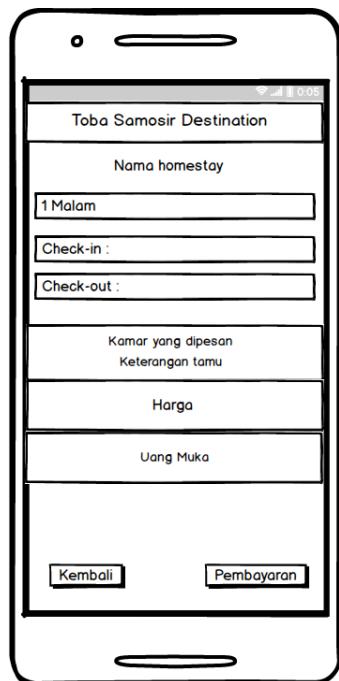
Berikut design interface rincian pemesanan pengunjung



Gambar 37 Design interface Rincian Pemesanan

9. *Design interface Rincian Akhir*

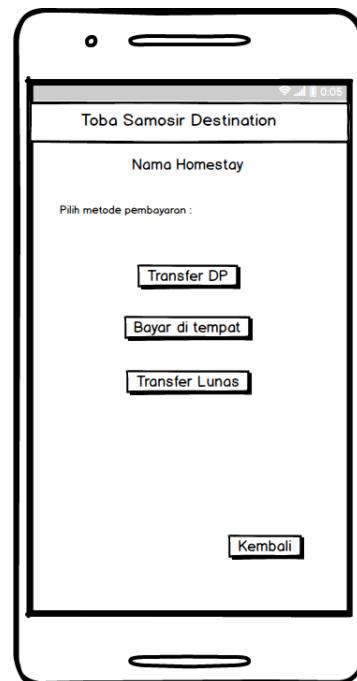
Berikut design interface rincian akhir pengunjung



Gambar 38 *Design interface Rincian Akhir*

10. *Design interface Pembayaran*

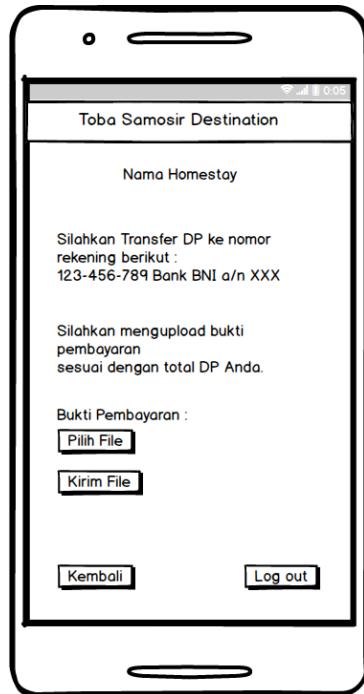
Berikut design interface pembayaran pengunjung



Gambar 39 *Design interface Pembayaran*

11. *Design interface Transfer DP*

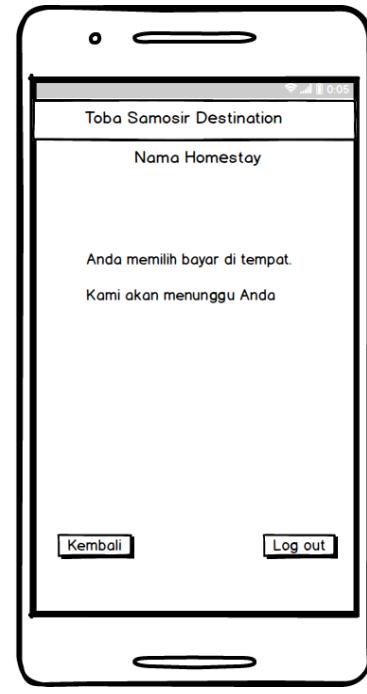
Berikut design interface Transfer DP pengunjung



Gambar 40 Design interface Transfer DP

12. *Design interface Bayar di Tempat*

Berikut design interface bayar ditempat pengunjung



Gambar 41 Design interface Bayar di Tempat

13. *Design interface Transfer Lunas*

Berikut design interface transfer lunas pengunjung

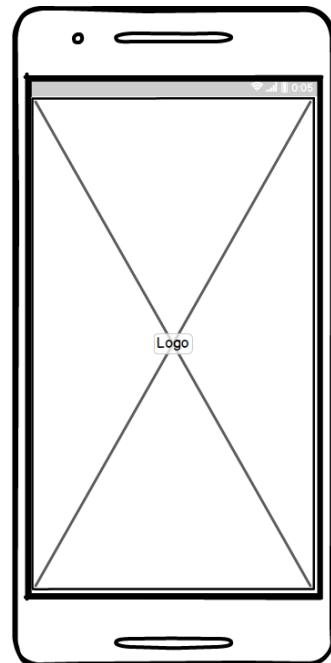


Gambar 42 Design interface Transfer Lunas

4.1.3 Design Interface Pemilik Homestay

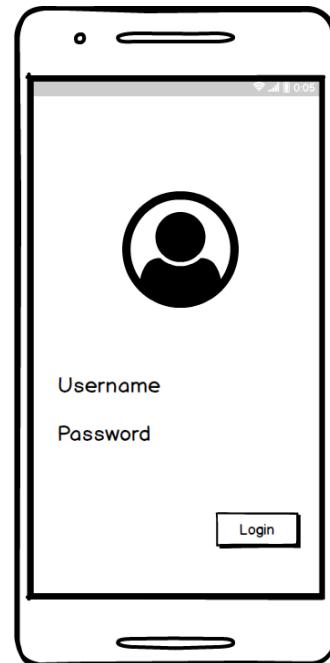
1. *Design interface* Splashscreen
2. *Design interface* Login

Berikut design interface
splashscreen pemilik homestay



Gambar 43 *Design interface*
Splashscreen

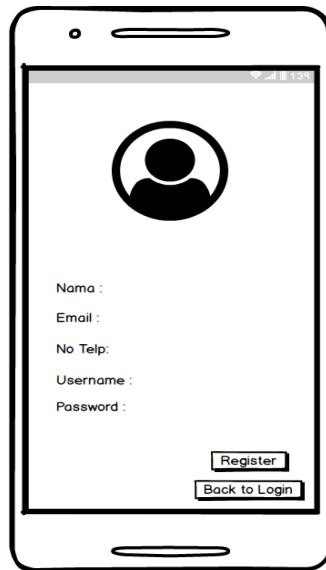
Berikut design interface login
pemilik homestay



Gambar 44 *Design interface* Login

3. *Design interface Register*

Berikut design interface register pemilik homestay



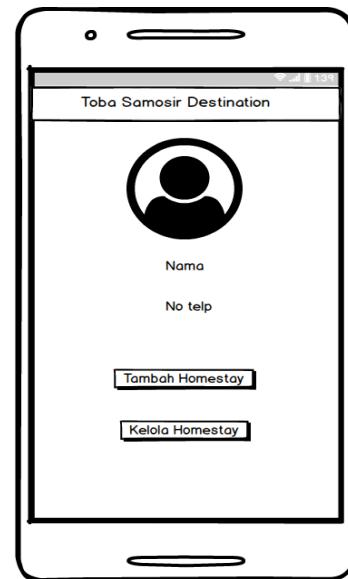
Gambar 45 Design interface Register

5. *Design interface Tambah Homestay*

Berikut design interface tambah homestay pemilik homestay

4. *Design interface Home*

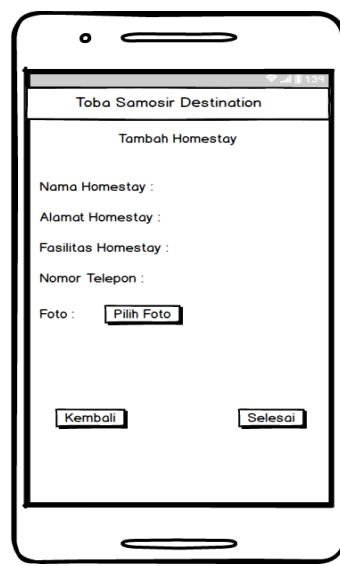
Berikut design interface home pemilik homestay



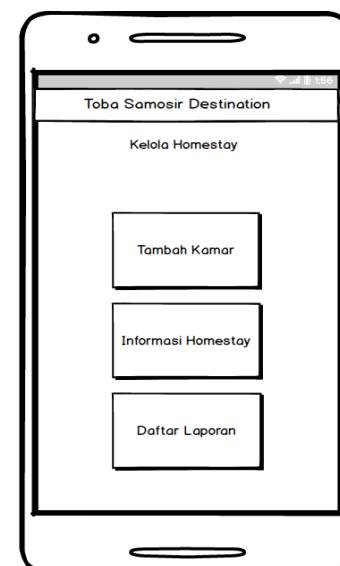
Gambar 46 Design interface Home

6. *Design interface Kelola Homestay*

Berikut design interface kelola homestay pemilik homestay



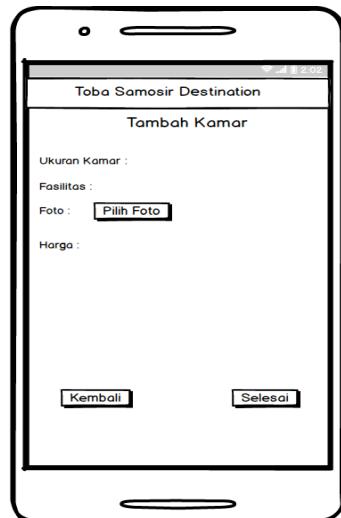
Gambar 47 Design interface Tambah Homestay



Gambar 48 Design interface Kelola Homestay

7. Design interface Tambah Kamar

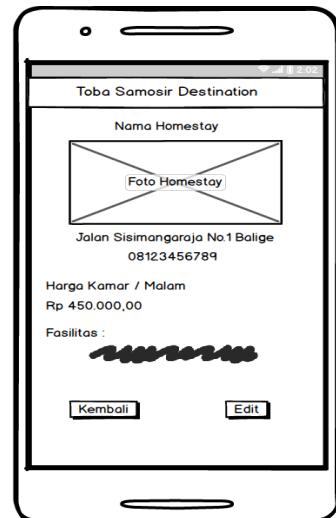
Berikut design interface tambah kamar pemilik homestay



Gambar 49 Design interface Tambah Kamar

8. Design interface Informasi Homestay

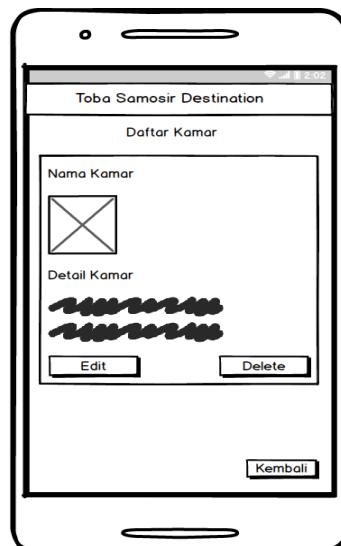
Berikut design interface informasi homestay pemilik homestay



Gambar 50 Design interface Informasi Homestay

9. Design interface Daftar Kamar

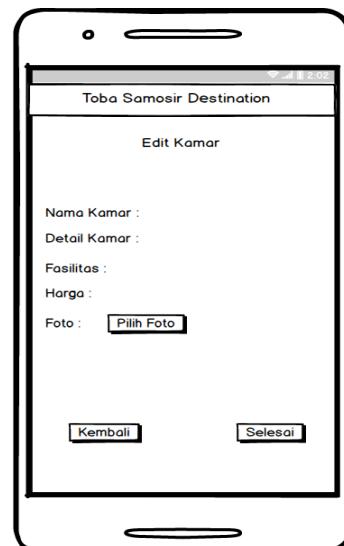
Berikut design interface daftar kamar pemilik homestay



Gambar 51 Design interface Daftar Kamar

10. Design interface Edit Kamar

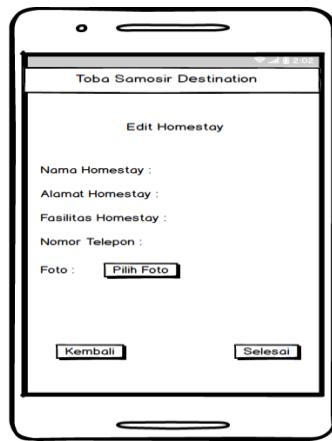
Berikut design interface edit kamar pemilik homestay



Gambar 52 Design interface Edit Kamar

11. Design interface Edit Informasi Homestay

Berikut design interface edit informasi pemilik homestay



**Gambar 53 Design interface Edit
Informasi Homestay**

4.2 Design Prototype High Fidelity

Pada pertemuan ini *user* melakukan pengujian terhadap design *prototype* yang akan dibangun.

4.2.1 Design Prototype Admin

1. Design interface Beranda

Berikut design interface *admin* untuk fungsi *login* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 54 Design interface Beranda

2. Design interface Daftar Homestay

Berikut design interface *admin* untuk fungsi daftar *homestay* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| Pendaftaran Homestay | | | | | | |
|----------------------|---------------|---------------|--------|--------------------|--------------------|------|
| ID | Nama homestay | Fasilitas | Alamat | Poi | Status Persetujuan | Opsi |
| 2 | Evelyn | 1.Kamar Mandi | Silaen | 1. Lumban Bulbul 1 | Di tolak | |

Gambar 55 Design interface Daftar Homestay

3. Design interface Tambah Homestay

Berikut design interface *admin* untuk fungsi tambah *homestay* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

Gambar 56 Design interface Tambah Homestay

4. Design interface Lihat Homestay

Berikut design interface *admin* untuk fungsi *login* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| Detail Homestay | |
|-----------------|--|
| ID Homestay | 53 |
| Nama Homestay | Teb Jose Homestay |
| Jumlah Kamar | 3 |
| Deskripsi | Aman, nyaman dan lingkungan asri |
| Kecamatan | 2 |
| Gambar | |
| Fasilitas | Kamar mandi luar |
| Alamat | Pantai Bul-bul |
| POI | 1. Pantai Bul-bul SGM 2. Museum Batak |

Gambar 57 Design interface Lihat Homestay

5. Design interface Edit Homestay

Berikut design interface *admin* untuk fungsi edit *homestay* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

Gambar 58 Design interface Edit Homestay

6. Design interface Cetak Data Homestay

Berikut design interface *admin* untuk fungsi *cetak data homestay* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| ID Homestay | Nama Homestay | Jumlah | Keterangan | Alamat | Poi |
|-------------|-------------------|--------|---|--|--|
| 6 | Blessing Homestay | 2 | Lokasi strategis, dekat pantai, cocok digunakan untuk acara keluarga | Pantai Bul-Bul | Pantai Lumban Bul-bul 50 m |
| 43 | Homestay Jos N Mo | 3 | Homestay Jos N Mo merupakan homestay dengan harga yang murah, nyaman seperti di rumah sendiri dan interaksi budaya kekeluargaan | Jl. Sutoemo No. 15 | Depan kantor camat 500 m sekolah unggulan Yusop 400 m makam Sisingamangaraja 500 m Museum Batake TB Center |
| 45 | Mikha Homestay | 3 | Yang SL mari datang ke homestay kami, cocok digunakan untuk keluarga | jl.guru somsalaing basuma bange kelurahan lumban dolok | Pantai Lumban Bul-bul |

[Print](#)

Gambar 59 Design interface Cetak Data Homestay

7. Design interface Pendaftaran Homestay

Berikut design interface *admin* untuk fungsi pendaftaran *homestay* yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| Id | Nama homestay | Fasilitas | Alamat | Poi | Status Persetujuan | Opsi |
|----|---------------|---------------|--------|--------------------|---|--|
| 2 | Evelyn | 1.Kamar Mandi | Silaen | 1. Lumban Bulbul 1 | Di tolak | |

Gambar 60 Design interface Pendaftaran Homestay

8. Design interface Konfirmasi Pendaftaran Homestay

Berikut design interface *admin* untuk fungsi konfirmasi pendaftaran yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| Detail Pendaftaran | |
|--------------------|-----------------------------------|
| ID | 2 |
| Nama Homestay | Evelyn |
| Alamat | Silaen |
| POL | 1. Lumban Bulbul 1 |
| Fasilitas | 1.Kamar Mandi |
| Deskripsi | Homestay ini didirikan tahun 2019 |
| Persetujuan | |

Beri persetujuan

Gambar 61 *Design interface Konfirmasi Pendaftaran Homestay*

9. Design interface Daftar Pengguna

Berikut design interface *admin* untuk fungsi daftar pengguna yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| Daftar Pengguna | | | | | | |
|-----------------|----------------------|----------------|-----------------------------|---------------|----------|------|
| ID | Nama Lengkap | Alamat | Email | Nomor Telepon | username | Opsi |
| 3 | rona | sigordang | ronatutfebriani09@gmail.com | 082360414241 | rona | |
| 14 | Marlina Simangunsong | Lumban Bul-bul | marlina@gmail.com | 08236041421 | marlinaa | |

Gambar 62 *Design interface Daftar Pengguna*

10. Design interface Pemesanan Homestay

Berikut design interface *admin* untuk fungsi pemesanan yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



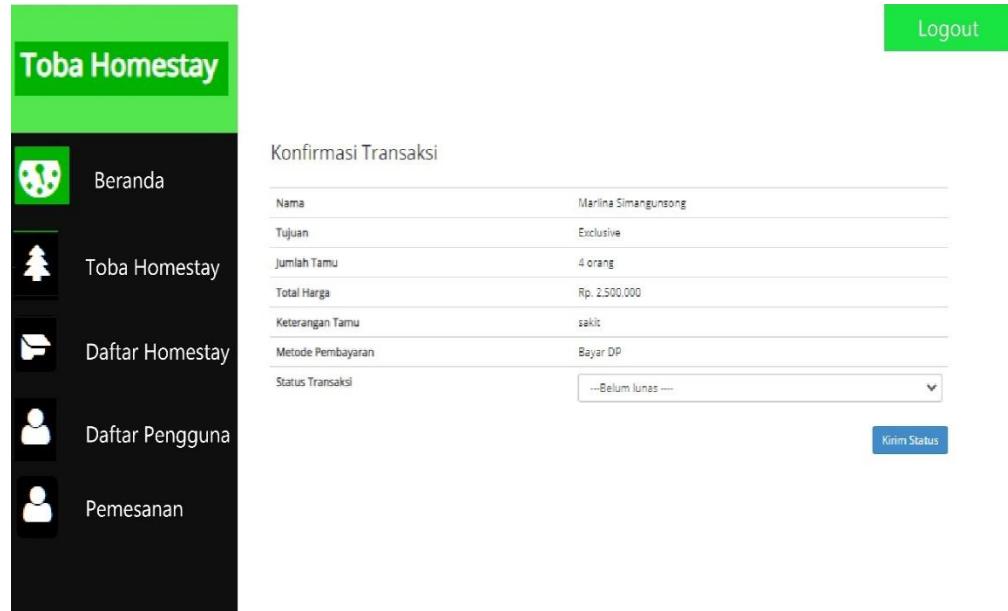
The screenshot shows the Toba Homestay admin interface. At the top right is a green 'Logout' button. On the left is a sidebar with a green header 'Toba Homestay' containing five items: 'Beranda' (with a house icon), 'Toba Homestay' (with a tree icon), 'Daftar Homestay' (with a camera icon), 'Daftar Pengguna' (with a person icon), and 'Pemesanan' (with a person icon). The main area has a white header 'Daftar Pemesanan'. Below it is a table with the following data:

| ID | Nama Pemesan | Room Type | Jumlah Tamu | Metode Pembayaran | Total Harga | Status | Bukti Pembayaran | Opsi |
|----|------------------------------|-----------|-------------|-------------------|-------------|-------------|------------------|--------------------------------------|
| 2 | Marline Simangunsong | Exclusive | 4 | Bayar DP | 2500000 | Belum Lunes | JPG | <button>→ Beri Konfirmasi</button> |
| 1 | Ronabur Febriani LumbangGeol | ttt | 4 | ss | 2500000 | Lunes | sss | <button>✓ Transaksi Selesai</button> |

Gambar 63 Design interface Pemesanan Homestay

11. Design interface Konfirmasi Pemesanan

Berikut design interface *admin* untuk fungsi konfirmasi pemesanan yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan



Gambar 64 Design interface Konfirmasi Pemesanan

12. Design interface Cetak Laporan Pemesanan

Berikut design interface *admin* untuk fungsi cetak laporan pemesanan yang berupa *design high fidelity* yang ditunjukkan dengan *user* ketika melakukan pertemuan

| Laporan Pemesanan | | | | | | | | |
|-------------------|---------------------------------|-----------|-------------|-------------------|-------------|-------------|------------------|--|
| ID Pemesanan | Nama Pemesan | Room Type | Jumlah Tamu | Metode Pembayaran | Total Harga | Status | Bukti Pembayaran | |
| 1 | Ronatiur Febriani LumbanGaol | ttt | 4 | ss | 2500000 | Lunas | sss | |
| 2 | Marlina Simangunsong | Exclusive | 4 | Bayar DP | 2500000 | Belum lunas | JPG | |

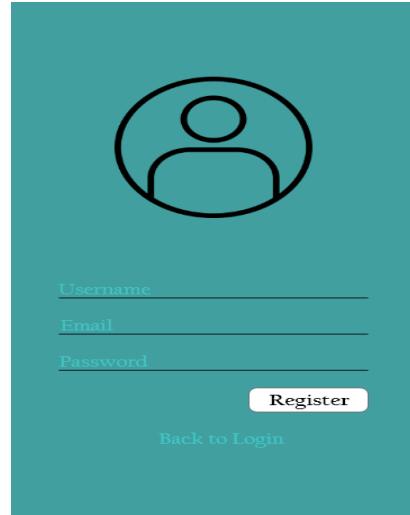
Print

Gambar 65 Design interface Cetak Laporan Pemesanan

4.2.2 Design Prototype Pengunjung

1. Design interface Register

Berikut *design interface register* dengan menggunakan *high fidelity* untuk pengunjung



Gambar 66 Design interface Register

2. *Design interface Login*

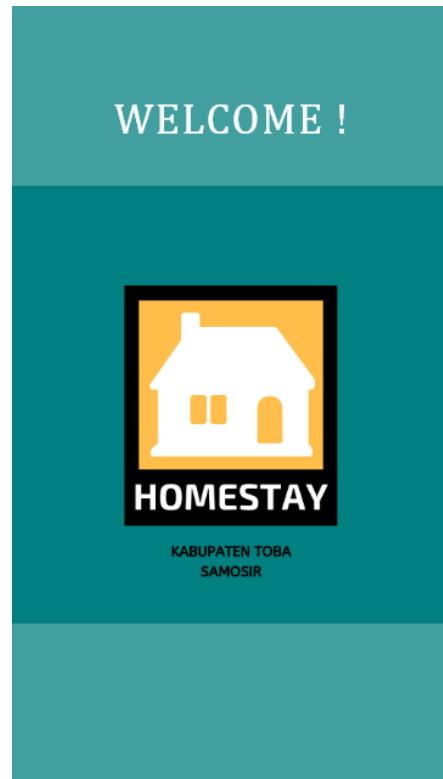
Berikut *design interface register* dengan menggunakan *high fidelity* untuk pengunjung



Gambar 67 Design interface Login

3. *Design interface Splashscreen*

Berikut *design interface Splashscreen* dengan menggunakan *high fidelity* untuk pengunjung



Gambar 68 *Design interface* Splashscreen

4. Design interface Home

Berikut *design interface* home dengan menggunakan *high fidelity* untuk pengunjung



Gambar 69 Design interface Home

5. Design interface List Homestay

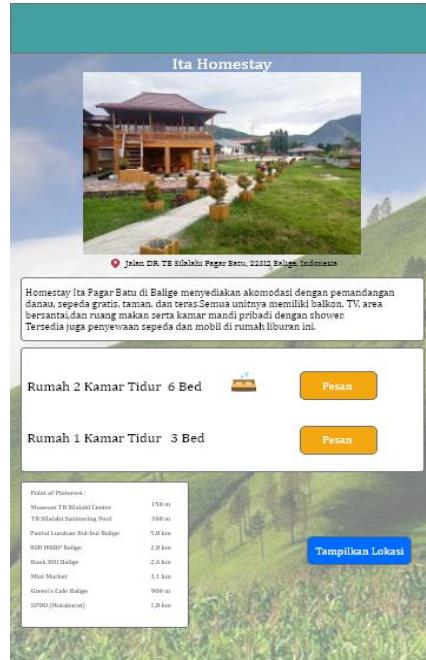
Berikut *design interface* List Homestay dengan menggunakan *high fidelity* untuk pengunjung



Gambar 70 Design interface List Homestay

6. Design interface Informasi Homestay

Berikut *design interface* Informasi Homestay dengan menggunakan *high fidelity* untuk pengunjung



Gambar 71 *Design interface* Informasi Homestay

7. Design interface Pemesanan Homestay

Berikut *design interface* pemesanan Homestay dengan menggunakan *high fidelity* untuk pengunjung



Gambar 72 Design interface Pemesanan Homestay

8. Design interface Rincian Pemesanan Homestay

Berikut *design interface* rincian pemesanan Homestay dengan menggunakan *high fidelity* untuk pengunjung

Gambar 73 Design interface Rincian Pemesanan Homestay

9. *Design interface Rincian Akhir Pemesanan Homestay*

Berikut *design interface* rincian akhir pemesanan *Homestay* dengan menggunakan *high fidelity* untuk pengunjung



Gambar 74 Design interface Rincian Akhir Pemesanan Homestay

10. Design interface Pembayaran

Berikut *design interface* pembayaran Homestay dengan menggunakan *high fidelity* untuk pengunjung



Gambar 75 Design interface Pembayaran

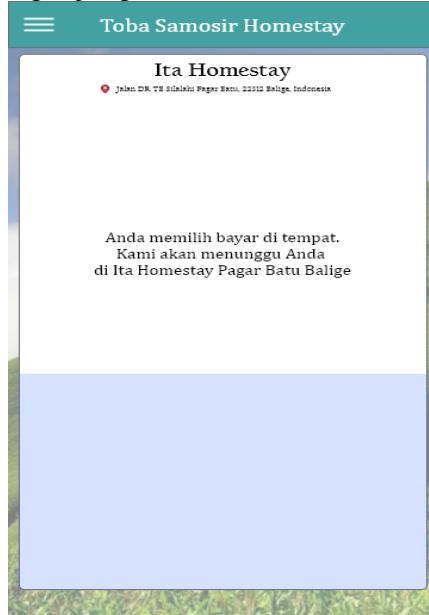
11. Design interface Peta Lokasi

Berikut *design interface* peta lokasi Homestay dengan menggunakan *high fidelity* untuk pengunjung



12. Design Interface Bayar di Tempat

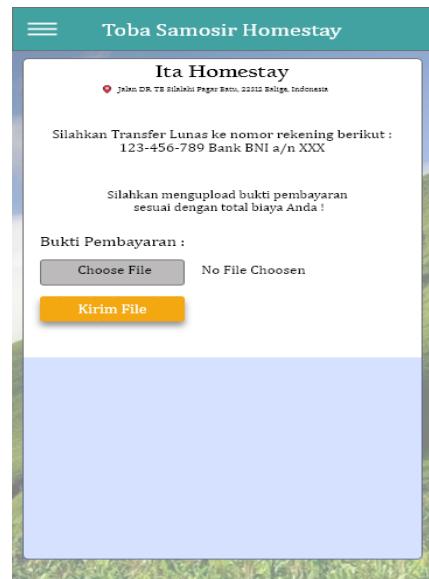
Berikut *design interface* bayar ditempat *Homestay* dengan menggunakan *high fidelity* untuk pengunjung



Gambar 76 Design interface Bayar di Tempat

13. Design interface Transfer Lunas Pembayaran

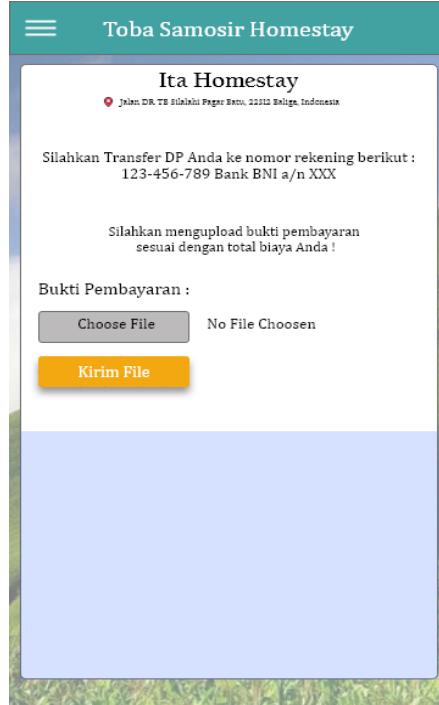
Berikut *design interface* transfer lunas *homestay* dengan menggunakan *high fidelity* untuk pengunjung



Gambar 77 Design interface Transfer Lunas Pembayaran

14. Design interface Transfer DP Pembayaran

Berikut *design interface* transfer DP Homestay dengan menggunakan *high fidelity* untuk pengunjung

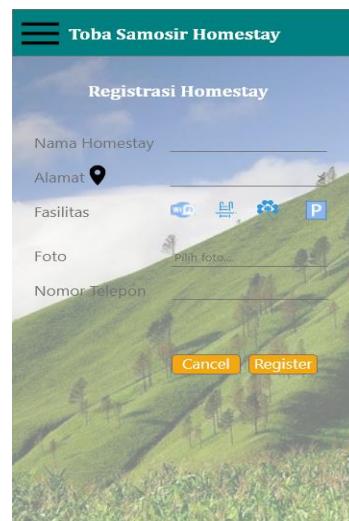


Gambar 78 Design interface Transfer DP Pembayaran

4.2.3 Design Prototype Pemilik Homestay

1. Design interface Register Homestay

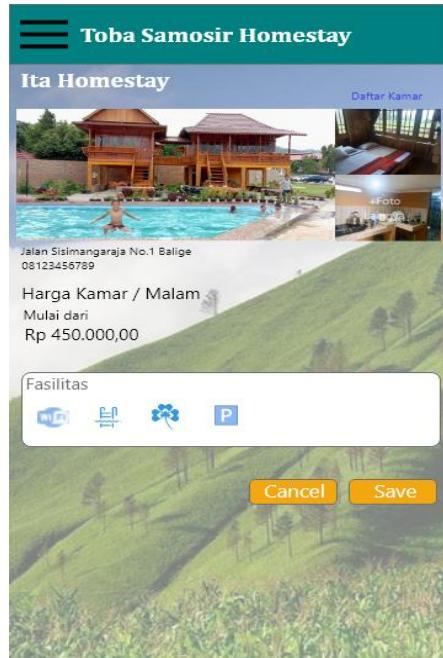
Berikut *Design interface* Register Homestay



Gambar 79 Design Interface Register Homestay

2. Design interface Edit Homestay

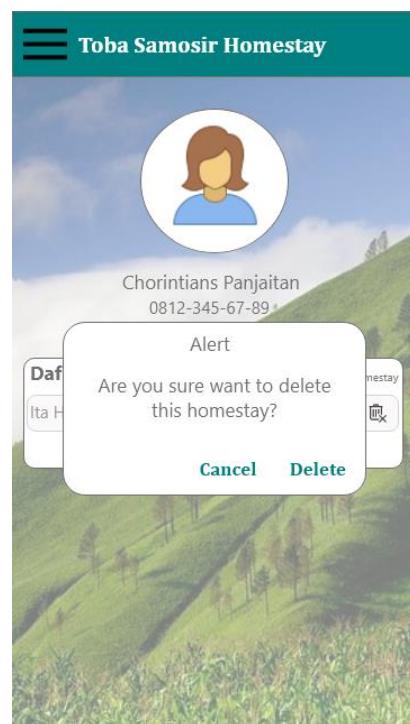
Berikut Design interface Edit Homestay



Gambar 80 Design interface Edit Homestay

3. Design interface Hapus Homestay

Berikut Design interface Hapus Homestay



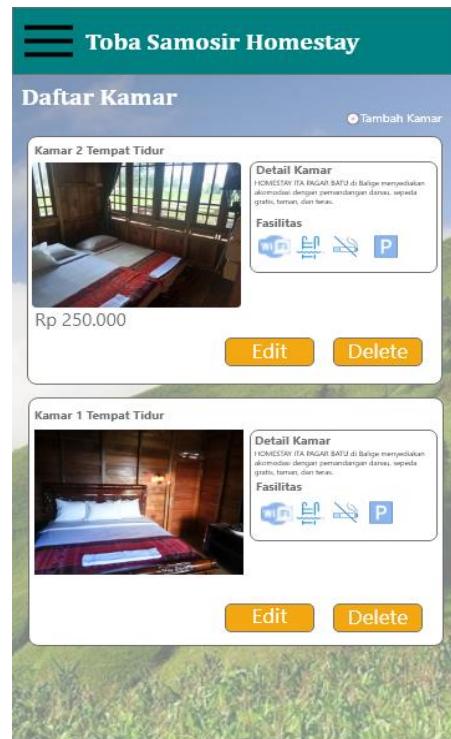
Gambar 81 Design interface Hapus Homestay

4. *Design interface* Tambah Kamar
 Berikut *Design interface* Tambah Kamar



Gambar 82 *Design interface* Tambah Kamar

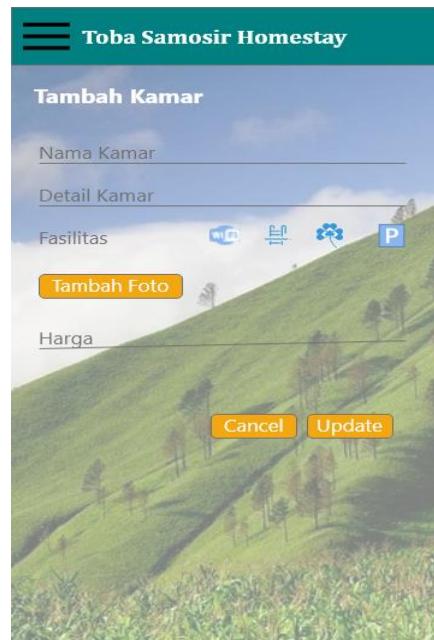
5. *Design interface* Daftar Kamar
 Berikut *Design interface* Daftar Kamar



Gambar 83 *Design interface* Daftar Kamar

6. Design interface Edit Kamar

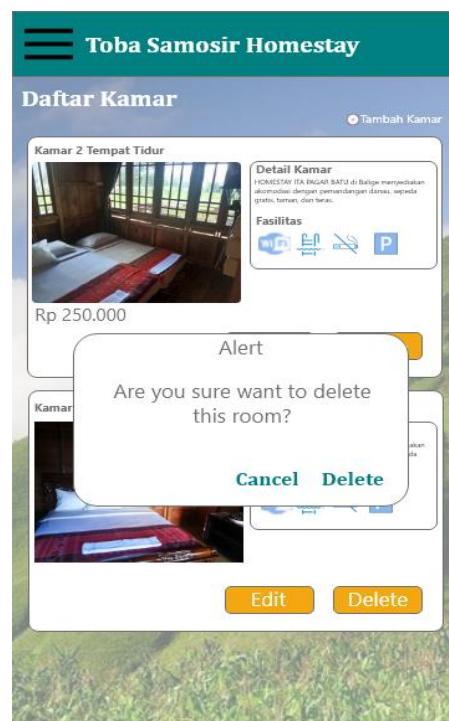
Berikut Design interface Edit Kamar



Gambar 84 Design interface Edit Kamar

7. Design interface Hapus Kamar

Berikut Design interface Hapus Kamar



Gambar 85 Design interface Delete Kamar

8. *Design interface* Notifikasi Pemesanan
Berikut *Design interface* Notifikasi Pemesanan



Gambar 86 *Design interface* Notifikasi

4.3 Evaluasi Prototype

Pada tahapan evaluasi *prototype*, *developer* sudah melakukan evaluasi *prototype* yang sudah dibangun kepada *user* dan pemilik *homestay*.

Hal yang dilakukan pada saat melakukan evaluasi prototype adalah dengan mengunjungi pemilik *homestay* yang berada di sekitaran Kota Balige dan kemudian menunjukkan prototype yang telah dibangun. Setelah itu, pemilik *homestay* menguji prototype yang telah dibangun.

Dari beberapa pemilik *homestay* yang telah melakukan evaluasi terhadap *prototype* yang telah dibangun hasil yang didapatkan adalah bahwa aplikasi yang akan dibangun sudah sesuai dengan kebutuhan pemilik *homestay* dan mudah digunakan dan harapan mereka bahwa aplikasi tersebut dapat digunakan segera dan menampilkan lokasi terdekat dengan *homestay* tersebut.

BAB V IMPLEMENTASI

Pada bab ini dijabarkan hasil penggeraan aplikasi dan proses pengujian model serta aplikasi, serta pembahasan mengenai hasil rancangan pada Bab III. Hal yang dibahas adalah implementasi untuk pembuatan aplikasi dan tahapan melakukan pengujian terhadap aplikasi.

5.1 Implementasi

Pada Bab implementasi dijelaskan mengenai implementasi dari aplikasi yang dibuat. Pada Bab ini juga akan memuat penjelasan lingkungan implementasi, *tools*, dan batasan implementasi.

5.1.1 Tools

Pada sub bab ini akan dipaparkan kebutuhan *tools* yang digunakan dalam pembangunan aplikasi. Dibutuhkan beberapa *tools hardware* maupun *software* dalam mendukung dan mempermudah proses pembangunan aplikasi, seperti penggeraan *code program*. Adapun spesifikasi *tools* yang digunakan dalam Rancang Bangun Aplikasi *Homestay* Daerah Toba Samosir Berbasis Android dapat dilihat pada tabel berikut.

Tabel 23. Tools Developing Environment

| | | |
|-----------------|---|---|
| Sistem Operasi | : | Windows 10 |
| <i>Software</i> | : | 1. Android Studio 3.5 2. Sublime Text 3 3. Enterprise Architect 4. IntelliJ Idea |
| <i>Hardware</i> | : | 1. Intel(R) Core(TM) i5 8GB RAM 2. Intel(R) Core(TM) i5 8GB RAM 3. Android Smartphone |
| <i>Library</i> | : | SDK, Retrofit |

5.1.2 Data

Data yang digunakan pada pembangunan aplikasi *Homestay* diperoleh dari berbagai sumber. Data koordinat yang digunakan sebagai data spasial diperoleh menggunakan GPS. Data foto-foto *homestay* diperoleh dari survei yang telah dilakukan dan dari berbagai website dan dari Dinas Kepariwisataan Toba Samosir.

5.1.3 Persiapan Pembangunan Aplikasi

Pada subbab ini dijelaskan persiapan sebelum pelaksanaan implementasi aplikasi *homestay*. Berikut tahapannya:

1. Instalasi *tools/software*

Instalasi tools dan software dilakukan pada laptop yang akan digunakan sebagai media untuk menjalankan program dengan spesifikasi seperti pada subbab 4.1.1.

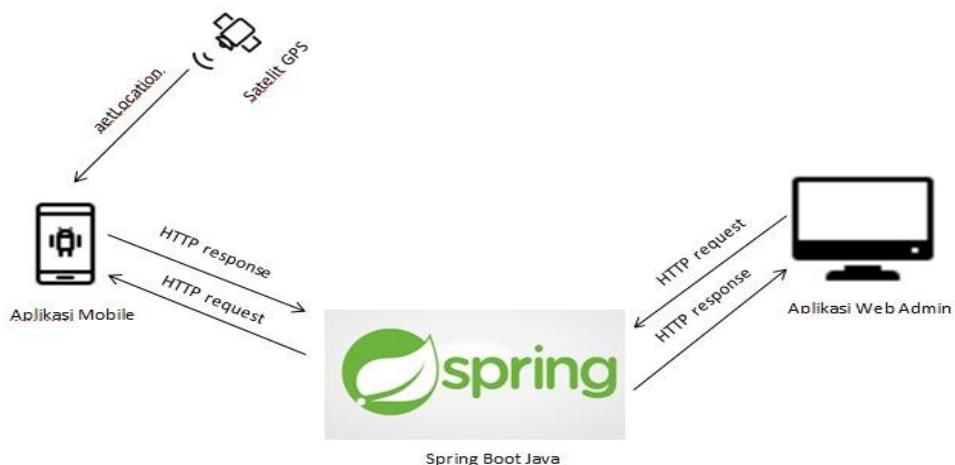
2. Setelah tools dan software diinstalasi maka selanjutnya akan dilakukan

Penulisan source code pada editor yang sudah diinstalasi. Setelah source code dituliskan, maka akan dilakukan proses pembangunan aplikasi *homestay*.

Tahapan dalam implementasi yang akan dilaksanakan oleh tim pengembang adalah sebagai berikut:

1. Tahap pertama adalah mengumpulkan data *homestay*. Data tersebut akan dikumpulkan dari berbagai sumber dan telah dipilih berdasarkan kebutuhan.
2. Tahap kedua adalah melakukan identifikasi *point of interest* yang terdapat pada beberapa *homestay*. Tahap ini dilakukan secara manual menggunakan Google Maps.
3. Tahap ketiga adalah melakukan implementasi *code program* untuk role *admin* dengan menggunakan bahasa pemrograman PHP pada tools Sublime Text 3. Role admin akan dibuat dengan model website.
4. Tahap keempat adalah implementasi untuk role pengunjung dan pemilik *homestay*. Dalam tahap ini tim pengembang akan melakukan implementasi untuk membuat aplikasi yang bisa dijalankan pada platform Android. Tahap ini akan menghasilkan aplikasi Android yang memiliki fungsi untuk menampilkan informasi detail *homestay* daerah Toba Samosir dan dapat melakukan pemesanan kamar *homestay*.

5.1.4 Perancangan Sistem



Gambar 87 Rancangan Sistem

Pada gambar diatas ditunjukkan gambaran umum arsitektur sistem yang menggunakan Spring boot Java sebagai back-end service.

Identifikasi Sistem Dari gambaran umum di atas, dapat ditemukan 3 aktor utama pada sistem aplikasi, yaitu pengguna mobile (pengunjung dan pemilik *homestay*) dan admin sebagai pengguna web.

Berikut merupakan tabel identifikasi aktor yang ditunjukkan pada tabel berikut.

Tabel 24 Identifikasi Aktor

| Identifikasi Aktor | Deskripsi |
|-------------------------|--|
| Pengunjung | Merupakan salah satu aktor pada aplikasi android, Pegunjung dapat memperoleh informasi data <i>homestay</i> , juga dapat menambah <i>homestay</i> baru. |
| Pemilik <i>Homestay</i> | Merupakan aktor pada aplikasi android, memiliki hak akses mengelola data <i>homestay</i> yang dimiliki, konfirmasi pemesanan, dan mendaftar <i>homestay</i> |
| Admin | Merupakan aktor pada aplikasi web, memiliki hak akses untuk mengelola daftar <i>homestay</i> dan konfirmasi pendaftaran <i>homestay</i> , untuk mengelola laporan pemesanan, |

5.1.5 Implementasi Fungsi

Pada bagian ini dijelaskan mengenai implementasi fungsi yang diimplementasikan pada aplikasi toba *homestay* serta input dan output dari setiap fungsi. Implementasi fungsi pada pembangunan aplikasi toba *homestay* dapat dilihat pada Tabel 24, Tabel 25, dan Tabel 26

Tabel 25 Implementasi Fungsi Web (Admin)

| No | Nama Use Case | Nama Fungsi | Keterangan |
|----|--------------------------------|-------------|--|
| 1 | Mengelola Daftar Homestay | Create | Menambah homestay baru |
| | | Detail | Melihat detail data homestay |
| | | Update | Mengubah data homestay |
| | | Delete | Menghapus data homestay |
| | | Cetak | Mencetak data <i>homestay</i> |
| 2 | Mengelola pengguna | Delete | Menghapus data pengguna |
| 3 | Mengelola pemesanan | Konfirmasi | Konfirmasi pemesanan |
| | | Cetak | Cetak pemesanan |
| 4 | Mengelola Pendaftaran Homestay | Konfirmasi | Konfirmasi pendaftaran <i>homestay</i> |
| | | Delete | Menghapus pendaftaran <i>homestay</i> |

Tabel 26 Implementasi Fungsi Pengunjung

| No | Nama Use Case | Nama Fungsi | Keterangan |
|----|------------------|-------------|---|
| 1 | Registrasi Akun | Register | Mendaftar akun pengunjung |
| 2 | Login | Login | Melakukan login untuk mengakses aplikasi |
| 3 | Melihat Homestay | View | Melihat list homestay yang dibuat per kecamatan |
| | | Detail | Melihat detail homestay yang dipilih pengunjung |

| | | | |
|---|------------------------------|------|-------------------------------|
| 4 | Melakukan Pemesanan Homestay | Add | Melakukan pemesanan homestay |
| 5 | Melakukan Pembayaran | Add | Melakukan pembayaran homestay |
| 6 | Melihat Lokasi Homestay | View | Melihat lokasi homestay |

Tabel 27 Implementasi Fungsi Pemilik Homestay

| No | Nama Use Case | Nama Fungsi | Keterangan |
|----|----------------------|----------------------|----------------------|
| 1. | Login | Login | Login |
| 2. | Registrasi Akun | Registrasi Akun | Registrasi Akun |
| 3. | Mendaftar Homestay | Mendaftar Homestay | Mendaftar Homestay |
| 4. | Mengelola Data Kamar | Mengelola Data Kamar | Mengelola Data Kamar |
| | | Detail | |
| | | Update | |
| | | Delete | |

5.1.6 Implementasi Code

Kode program yang diimplementasikan untuk membangun aplikasi Toba *Homestay* adalah sebagai berikut:

5.1.6.1 Implementasi Code Mengelola Homestay

Pada fungsi mengelola data *homestay* terdapat empat fungsi yaitu *create*, *update*, *delete*, dan cetak yang dapat dilakukan oleh *administrator*. Kode programnya dapat dilihat pada gambar dibawah ini

Tabel 28 Implementasi code create homestay

```
<?php
include 'config.php';
$nama = $_POST['nama'];
/*$harga = $_POST['harga'];*/
$jumlah = $_POST['qty'];
$deskripsi = $_POST['keterangan'];
$fasilitas = $_POST['fasilitas'];
$alamat = $_POST['alamat'];
$poi = $_POST['poi'];
$nama_kec = $_POST['kecamatan'];
```

```

$kecamatan = mysqli_query($koneksi, "SELECT * FROM kecamatan WHERE nama_kec= '$nama_kec'");
$kat = mysqli_fetch_array($kecamatan);
$id_kec = $kat['id_kec'];
$gambar = $_FILES['gambar']['name'];
move_uploaded_file($_FILES['gambar']['tmp_name'], 'gambar/'.$gambar);
$sql = mysqli_query($koneksi, "INSERT INTO homestay VALUES('$nama', '$jumlah', '$deskripsi', '$gambar', '$id_kec', '$fasilitas', '$alamat', '$poi')");
if ($sql) {
echo "<script>alert('Data $nama berhasil disimpan!'); window.location ='data_homestay.php'</script>";
} else {
echo "<script>alert('Data $nama gagal disimpan!'); window.location ='data_homestay.php'</script>";
}
?>

```

Tabel 29 Implementasi code fungsi update

```

<?php
$id_brg=mysqli_real_escape_string($koneksi, $_GET['id']);
$det=mysqli_query($koneksi, "select * from homestay where id_homestay='$id_brg'")or die(mysql_error());
$d=mysqli_fetch_array($det);
?>
<form action="update.php" method="post" enctype="multipart/form-data">
<table class="table">
<tr>
<td></td>
<td><input type="hidden" name="id" value="<?php echo $d['id_homestay'] ?>"></td>
</tr>
<tr>
<td>Nama</td>
<td><input type="text" class="form-control" name="nama" value="<?php echo $d['nama'] ?>"></td>
</tr>
<!-- <tr>
<td>Harga</td>
<td><input type="text" class="form-control" name="harga" value="<?php echo $d['harga'] ?>"></td>
</tr> -->
<tr>
<td>Kecamatan</td>
<td>
<select class="form-control" name="kecamatan">
<?php
$query = mysqli_query($koneksi, "SELECT * FROM kecamatan");
while($q = mysqli_fetch_array($query)){
echo '<option>'. $q['nama_kec'] .</option>';
}
?>
</select>
</td>
</tr>
<tr>
<td>Jumlah Kamar</td>

```

```

        <td><input type="text" class="form-control" name="qty" value="<?php echo
$d['qty'] ?>"></td>
</tr>
<tr>
    <td>Deskripsi</td>
    <td><textarea class="form-control" name="deskripsi"><?php echo $d['keterangan']
?></textarea>
</tr>
<tr>
    <td>Gambar</td>
    <td>
        <br/>
        <input type="file" class="form-control" name="gambar" value="<?php echo
$d['gambar'] ?>">
    </td>
</tr>
<tr>
    <td>Fasilitas</td>
    <td><textarea class="form-control" name="fasilitas"><?php echo $d['fasilitas']
?></textarea>
</tr>
<tr>
    <td>Alamat</td>
    <td><textarea class="form-control" name="alamat"><?php echo $d['alamat']
?></textarea>
</tr>
<tr>
    <td>POI</td>
    <td><textarea class="form-control" name="poi"><?php echo $d['poi'] ?></textarea>
</tr>
<tr>
    <td></td>
    <td><input type="submit" class="btn btn-info" value="Simpan"></td>
</tr>
</table>
</form>

```

Tabel 30 Implementasi code fungsi view

```

<?php
$id_brg=mysqli_real_escape_string($koneksi, $_GET['id']);

$det=mysqli_query($koneksi, "select * from homestay where id_homestay='$id_brg'")or
die(mysql_error());
while($d=mysqli_fetch_array($det)){
?
<table class="table">
<tr>
    <td>ID Homestay</td>
    <td><?php echo $d['id_homestay'] ?></td>
</tr>
<tr>
    <td>Nama Homestay</td>
    <td><?php echo $d['nama'] ?></td>
</tr>
<!-- <tr>
    <td>Harga</td>
    <td>Rp.<?php echo number_format($d['harga']) ?>,-</td>

```

```

</tr> -->
<tr>
    <td>Jumlah Kamar</td>
    <td><?php echo number_format($d['qty']) ?></td>
</tr>
<tr>
    <td>Deskripsi</td>
    <td><?php echo $d['keterangan']; ?></td>
</tr>
<tr>
    <td>Kecamatan</td>
    <td><?php echo $d['id_kec']; ?></td>
</tr>
<tr>
    <td>Gambar</td>
    <td></td>
</tr>
<tr>
    <td>Fasilitas</td>
    <td><?php echo $d['fasilitas']; ?></td>
</tr>
<tr>
    <td>Alamat</td>
    <td><?php echo $d['alamat']; ?></td>
</tr>
<tr>
    <td>POI</td>
    <td><?php echo $d['poi']; ?></td>
</tr>
</table>
<?php
}
?>

```

Tabel 31 Implementasi Code Fungsi Cetak Data

```

<table>
    <tr class="tableheader">
        <th rowspan="1">ID Homestay</th>
        <th>Nama Homestay</th>
        <th>Jumlah</th>
        <th>Keterangan</th>
        <th>Alamat</th>
        <th>Poi</th>
    </tr>
    <?php
    if (isset($_GET['homestay'])) {
        $cari = mysqli_real_escape_string($koneksi, $_GET['homestay']);
        $data = mysqli_query($koneksi, "SELECT * FROM homestay
WHERE nama = '$cari' order by nama");
    } else{
        $data = mysqli_query($koneksi, "SELECT * FROM homestay");
    }
    while ($hasil = mysqli_fetch_array($data)) { ?>
        <tr id="rowHover">
            <td width="10%" align="center"><?php echo
$hasil['id_homestay']?></td>
            <td width="25%" align="center"><?php echo
$hasil['nama']?></td>

```

```

$hasil['qty']?></td>
<td width="15%" align="center"><?php echo
$hasil['keterangan']?></td>
<td width="25%" align="center"><?php echo
$hasil['alamat']?></td>
<td width="25%" align="center"><?php echo
$hasil['poi']?></td> </tr>
<?php
}
?>
<tr>
</tr>

</table>
<script>
window.load = print_data();
function print_data(){
    window.print();
}

```

5.1.6.2 Implementasi Code Mengelola Data Pengguna

Pada Fungsi mengelola data pengguna, *admin* dapat menghapus data pengguna yang sudah pernah *login*. Berikut kode program hapus data pengguna.

Tabel 32 Implementasi Code Fungsi Hapus Pengguna

```

<?php
    global $conn;
    $conn = mysqli_connect("localhost", "root", "", "homestay");

    // $conn = mysqli_connect("localhost", "p1d4ti08", "6YsE4o1p", "p1d4ti08_pu");

    if(!$conn){
        die("Masalah Pada Database");
    }

    $db_use = mysqli_select_db($conn, "homestay") or die("Select Database Problem !!");

    $id = $_GET['id_user'];
    $query = "DELETE FROM users WHERE id_user=$id";
    if($conn->query($query)==true){
        echo "<script>alert('Hapus Pengguna Berhasil');</script>";
    }
    header('Refresh:0 url=data_user.php');
?>

```

5.1.6.3 Implementasi Code Mengelola Pendaftaran Homestay

Pada fungsi mengelola pendaftaran *homestay*, *admin* dapat melakukan konfirmasi pendaftaran *homestay* dan menghapusnya. Berikut kode program fungsi konfirmasi dan hapus.

Tabel 33 Implementasi Code Fungsi Konfirmasi Pendaftaran

```
<?php } ?>
<tr>
    <th>Persetujuan</th>
    <td>
        <form
action="beri_notifikasi_pendaftaran.php?id_pendaftaran=<?=$id_pendaftaran;?>"
method="post">
            <select class="form-control" name="ajukan">
                <option>-----</option>

                <option>Di terima</option>
                <option style="color: red;">Di tolak</option>
            </select>
        </td>
    </tr>

</table>
<button class="btn btn-primary" style="float: right;" name="beriPersetujuan">Ber
persetujuan</button>
</form>
</div>
</div>
</div>
```

Tabel 34 Implementasi Code Fungsi Hapus Pendaftaran

```
<?php
    global $conn;
    $conn = mysqli_connect("localhost", "root", "", "homestay");

    // $conn = mysqli_connect("localhost", "p1d4ti08", "6YsE4o1p", "p1d4ti08_pub");
    if(!$conn){
        die("Masalah Pada Database");
    }

    $db_use = mysqli_select_db($conn, "homestay") or die("Select Database Problem !!");

    $id = $_GET['id_pendaftaran'];
    $query = "DELETE FROM pendaftaran WHERE id_pendaftaran=$id";
    if($conn->query($query)==true){
        echo "<script>alert('Hapus Lamaran Berhasil');</script>";
    }
    header('Refresh:0 url=data_pendaftaran_homestay.php');
?>
```

5.1.6.4 Implementasi Code Mengelola Pemesanan Homestay

Pada fungsi mengelola pemesanan homestay terdapat dua fungsi yaitu konfirmasi pemesanan *homestay* dan cetak pemesanan

Tabel 35 Implementasi Code Fungsi Konfirmasi pemesanan

```
<form action="beri_konfirmasi_transaksi_process.php?id_toba=<?=$idToba?>"
method="post">
    <tr>
        <th>Status Transaksi</th>
        <td>
            <select name="statusPemesanan" class="form-control">
```

```

<option value=<?=$tobawisa['status_transaksi']?>>--->
<?=$tobawisa['status_transaksi']?>----</option>
    <option>Di terima</option>
    <option>Lunas</option>
</select>
<?php } ?>
</td>
</tr>
</table>
<button name="beriKonfirmasi" style="float: right" class="btn btn-primary">Kirim Status</button>
</form>
</div>
</div>
</div>

```

5.1.6.5 Implementasi Code Registrasi Akun Pengunjung

Tabel 36 Implementasi Code Registrasi Akun Pengunjung

```

class RegisterActivity : BaseActivity(), RegisterContract.View {

    @Inject lateinit var presenter: RegisterPresenter
    private var request = RegisterRequest()

    companion object {
        private val TAG = "RegisterActivity"

        @JvmStatic fun newIntent(context: Context): Intent {
            return Intent(context, RegisterActivity::class.java)
        }
    }
}

```

5.1.6.6 Implementasi Code Login Pengunjung

```

1  class LoginActivity : BaseActivity(), LoginContract.View {
2
3      @Inject lateinit var presenter: LoginPresenter
4
5      companion object {
6          private val TAG = "LoginActivity"
7
8          @JvmStatic fun newIntent(context: Context): Intent
9          {
10              return Intent(context,
11                  LoginActivity::class.java)
12          }
13      }
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
297
298
299
299
300
301
302
303
304
305
305
306
307
308
309
309
310
311
312
313
313
314
315
315
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
561
561
562
562
563
563
564
564
565
565
566
566
567
567
568
568
569
569
570
570
571
571
572
572
573
573
574
574
575
575
576
576
577
577
578
578
579
579
580
580
581
581
582
582
583
583
584
584
585
585
586
586
587
587
588
588
589
589
590
590
591
591
592
592
593
593
594
594
595
595
596
596
597
597
598
598
599
599
600
600
601
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
13
```

5.1.6.7 Implementasi Code Melihat Homestay

Tabel 38 Implementasi Code Melihat Homestay

```
1 class HomestayDetailActivity : BaseActivity(),
2 HomestayDetailContract.View {
3
4     @Inject lateinit var presenter: HomestayDetailPresenter
5     private var oneBedRoom = ""
6     private var twoBedRoom = ""
7     private var fasilitas = ""
8
9     companion object {
10         private val TAG = "LoginActivity"
11
12         @JvmStatic fun newIntent(context: Context, id: String): Intent {
13             val i = Intent(context,
14             HomestayDetailActivity::class.java)
15             i.putExtra(IntentKey.HOMESTAYID, id)
16
17             return i
18         }
19     }
20 }
```

5.1.6.8 Implementasi Code Melakukan Pemesanan Homestay

Tabel 39 Implementasi Code Melakukan Pemesanan Homestay

```
1 class PemesananHomestayActivity : BaseActivity(),
2 PemesananHomestayContract.View {
3
4     @Inject lateinit var presenter:
5     PemesananHomestayPresenter
6     @Inject lateinit var localPreferences:
7     LocalPreferences
8
9     private var price : Int = 0
10    private var id = ""
11    private var request = PemesananRequest()
12
13    companion object {
14        private val TAG = "PemesananHomestayActivity"
15
16        @JvmStatic fun newIntent(context: Context, id: String, fasilitas: String): Intent {
17            val i = Intent(context,
18            PemesananHomestayActivity::class.java)
19            i.putExtra(IntentKey.KAMARID, id)
20            i.putExtra(IntentKey.FASILITAS, fasilitas)
21
22            return i
23        }
24    }
25 }
```

5.1.6.9 Implementasi Code Melakukan Pembayaran Homestay

Tabel 40 Implementasi Code Melakukan Pembayaran Homestay

```

1 class PembayaranActivity : BaseActivity(),
    PembayaranContract.View {
2
3     @Inject lateinit var presenter: PembayaranPresenter
4
5     companion object {
6         private val TAG = "PembayaranActivity"
7
8         @JvmStatic fun newIntent(context: Context): Intent
9         {
10             return Intent(context,
11                 PembayaranActivity::class.java)
12         }
13     }
14 }
```

5.1.6.10 Implementasi Code Melihat Lokasi Homestay

Tabel 41 Implementasi Code Melihat Lokasi Homestay

```

class MainActivity : BaseActivity(), MainContract.View,
HomestayAdapter.HomestayAdapterListener{

    @Inject lateinit var presenter: MainPresenter
    private var request = -1
    private var adapter = HomestayAdapter(this)

    companion object {
        private val TAG = "MainActivity"

        @JvmStatic fun newIntent(context: Context): Intent {
            return Intent(context, MainActivity::class.java)
        }
    }
}
```

5.1.6.11 Implementasi Code Mengupdate Data Kamar

Tabel 42 Implementasi Code Mengupdate Data Kamar

```

class EditKamarActivity : BaseActivity(), EditKamarContract.View {
    @Inject
    lateinit var presenter: EditKamarPresenter
    private var request = EditKamarRequest()
    private var no_kamar = ""
    companion object {
        private val TAG = "KamarEditActivity"
        @JvmStatic fun newIntent(context: Context, id: String): Intent {
            val i = Intent(context, EditKamarActivity::class.java)
            i.putExtra(IntentKey.KAMARID, id)
            return i
        }
    }
    override fun onCreate(savedInstanceState: Bundle?) {
        AndroidInjection.inject(this)
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_edit_kamar)
        getData()
    }
}
```

```
initListener()
}
```

5.1.6.12 Implementasi Code Melihat Detail Kamar

Tabel 43 Implementasi Code Melihat Detail Kamar

```
1 abstract class DetailKamarActivity : BaseActivity(), DetailKamarContract.View {
2
3     @Inject
4     lateinit var presenter: DetailKamarPresenter
5     private var id = ""
6
7     companion object {
8         private val TAG = "LoginActivity"
9
10        @JvmStatic fun newIntent(context: Context, id: String): Intent {
11            val i = Intent(context, DetailKamarActivity::class.java)
12            i.putExtra(IntentKey.KAMARID, id)
13
14            return i
15        }
16    }
```

5.1.6.13 Implementasi Code Menambah Kamar

Tabel 44 Implementasi Code Menambah Kamar

```
1 class CreateKamarActivity : BaseActivity(), CreateKamarContract.View {
2
3     @Inject
4     lateinit var presenter: CreateKamarPresenter
5     private var request = CreateKamarRequest()
6
7     companion object {
8         private val TAG = "CreateKamarActivity"
9
10        @JvmStatic fun newIntent(context: Context): Intent {
11            return Intent(context, CreateKamarActivity::class.java)
12        }
13    }
14
15    override fun onCreate(savedInstanceState: Bundle?) {
16        AndroidInjection.inject(this)
17        super.onCreate(savedInstanceState)
18        setContentView(R.layout.activity_creat_kamar)
```

5.1.6.14 Implementasi Code Melihat Daftar Kamar

Tabel 45 Implementasi Code Melihat Daftar Kamar

```
1 class MainActivity : BaseActivity(), MainContract.View,
2     KamarAdapter.KamarAdapterListener{
3     @Inject lateinit var presenter: MainPresenter
4     private var request = -1
5     private var adapter = KamarAdapter(this)
6     companion object {
7         private val TAG = "MainActivity"
8
9         @JvmStatic fun newIntent(context: Context): Intent {
```

```

9         return Intent(context, MainActivity::class.java)
10    }
11 }
```

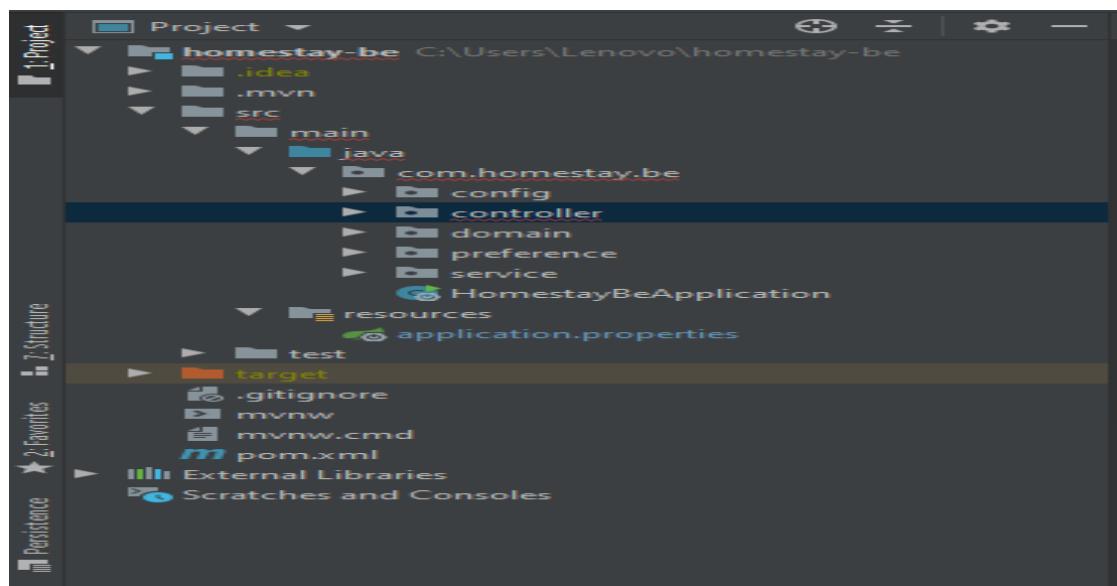
5.1.7 Implementasi Web service dengan Springboot java

Pada perancangan *service* ini menggunakan android kotlin dengan *Spring boot* java dengan configurasi *maven*. *Spring* adalah kerangka kerja lengkap dengan modular yang digunakan untuk pengembangan aplikasi berbasis *java*, sehingga dapat fokus pada logika bisnis tingkat aplikasi. *Spring boot* adalah titik awal untuk membangun semua aplikasi berbasis *spring*. *Spring boot* dirancang untuk membuat lebih cepat. *Service* yang telah diset ke port 7078 akan mengakses lokasi port menggunakan ipv4 *network* yang terhubung. Untuk dokumentasi API servicenya menggunakan *swagger* dengan mengakses <http://localhost:7078/swagger-ui.html#/>

Berikut contoh bagian kode dengan configurasi maven yang telah dibuat

1. Struktur Maven

Pada IntelijIdea, berikut struktur maven yang telah dibuat



Gambar 88 Struktur Maven

2. Setup application properties

Tabel 46 Setup application properties

```
spring.datasource.url = jdbc:mysql://localhost:3306/homestay?useSSL=false
spring.datasource.username =root
spring.datasource.password =
spring.jpa.properties.hibernate.dialect =
org.hibernate.dialect.MySQL5Dialect
spring.jpa.hibernate.ddl-auto = update
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
spring.main.allow-bean-definition-overriding=true
```

3. Model

Model digunakan untuk *entity* data yang disesuaikan dengan database contohnya adalah *homestay.java*

Tabel 47 Model entity

```
import javax.persistence.*;
import java.util.Date;

@Data
@Builder
@Entity
@Table(name = "homestay")
@DynamicUpdate
@NoArgsConstructor
@AllArgsConstructor
public class Homestay {

    @Id @Column(name = "id_homestay", length = 11)
    @GeneratedValue(strategy = GenerationType.SEQUENCE)
    private int id;

    @Column(name = "nama", nullable = false, length = 100)
    private String nama;

    @Column(name = "qty", length = 9)
    private int qty;

    @Column(name = "keterangan", length = 500)
    private String keterangan;

    //    @Column(name = "gambar", length = 225)
    //    private String gambar;

    @Column(name = "fasilitas", length = 100)
    private String fasilitas;

    @Column(name = "alamat", length = 100)
    private String alamat;

    @Column(name = "poi", length = 500)
    private String poi;

    @Column(name = "id_kec", length = 11)
```

```

private int kecamatan;

public Homestay(CreateHomestayRequest request) {
    this.nama = request.getNama();
    this.qty = request.getQty();
    this.keterangan = request.getKeterangan();
    //      this.gambar = request.getGambar();
    this.fasilitas = request.getFasilitas();
    this.kecamatan = request.getKecamatan();
    this.alamat = request.getAlamat();
    this.poi = request.getPoi();
}
}

```

4. Repository

Setelah membuat entity kemudian membuat interface *homestayrepository* untuk mengakses JPA

Tabel 48 Repository

```

@Repository
public interface HomestayRepository extends
JpaRepository<Homestay, String> {

    Homestay findById(int id);

    List<Homestay> findByKecamatan(int kecamatan);
}

```

5. Service

Setelah itu membuat interface *homestayservice.java* untuk menentukan service yang harus diimplementasikan. Setelah itu membuat implementasi dari *homestayservice.java* dengan membuat class implementasi *homestayimpl.java*.

Tabel 49 Implementasi service

```

@Service
public class HomeImpl implements HomeService {

    @Autowired
    HomeRepository homeRepository;

    @Override
    public List<Home> findAll() {
        return homeRepository.findAll();
    }

    @Override
    public Home findById(int id) {
        return homeRepository.findById(id);
    }
}

```

```

}

@Override
public Home update(int id, Home home) {
    home.getId();
    return homeRepository.save(home);
}

@Override
public Home create(Home home) {
    return homeRepository.save(home);
}

@Override
public void delete(int id) {
    homeRepository.deleteById(String.valueOf(id));
}
}

```

6. Response

Kemudian untuk responsenya dibuat class BaseRespon.java

Tabel 50 Respons

```

package com.homestay.be.domain.response;

import lombok.Getter;
import lombok.Setter;

@Getter
@Setter
public class BaseResponse<T> {
    private int code;
    private String message;
    private T data;
}

```

7. Controller

Setelah itu membuat file *HomestayController.java* yang digunakan untuk akses RestAPI. Controller untuk melayani permintaan *Get*, *Post*, *Put*, dan *Delete* dan untuk mengembalikan JSON yang mewakili instance pengguna. Adapun format JSON yang telah dibuat adalah

@GetMapping: digunakan untuk mengambil informasi dari server yang diberikan menggunakan URI yang diberikan. Permintaan menggunakan GET

seharusnya hanya mengambil data dan seharusnya tidak memiliki efek lain pada data.

@PostMapping: digunakan untuk mengirim data ke server

@PutMapping: digunakan untuk mengganti data saat ini

@DeleteMapping: digunakan untuk menghapus data

Tabel 51 Controller

```

@RestController
@RequestMapping(value = "home")
public class HomeController {

    @Autowired
    HomeService homeService;

    @PostMapping
    ResponseEntity<BaseResponse> create (@RequestBody @Validated
    Home home) {
        BaseResponse response = new BaseResponse();
        response.setCode(200);
        response.setMessage("success");
        response.setData(homeService.create(home));

        return ResponseEntity
            .status(HttpStatus.OK)
            .contentType(MediaType.APPLICATION_JSON)
            .body(response);
    }
}

```

8. Swagger Config API

Tambahkan konfigurasi swagger untuk menggenerate dokumentasi Rest Api secara otomatis

Tabel 52 Swagger Config

```

package com.homestay.be.config;

import com.google.common.base.Predicates;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import springfox.documentation.builders.PathSelectors;
import springfox.documentation.builders.RequestHandlerSelectors;
import springfox.documentation.service.ApiInfo;
import springfox.documentation.service.Contact;
import springfox.documentation.spi.DocumentationType;
import springfox.documentation.spring.web.plugins.Docket;
import springfox.documentation.swagger2.annotations.EnableSwagger2;

```

```

import java.util.Collections;

@Configuration
@EnableSwagger2
public class SwaggerConfig {

    @Bean
    public Docket docket() {
        return new Docket(DocumentationType.SWAGGER_2)
            .select()
            .apis(RequestHandlerSelectors.any())
            .paths(PathSelectors.any())
            .paths(Predicates.not(PathSelectors.regex("/error/*")))
            .paths(Predicates.not(PathSelectors.regex("/actuator")))
            .build()
            .apiInfo(apiInfo());
    }

    private ApiInfo apiInfo() {
        Contact contact = new Contact(
            "", "",
            "", "");
        return new ApiInfo(
            "Homestay",
            "Homestay",
            "Version 1.0.0",
            "",
            contact,
            "",
            "",
            Collections.emptyList());
    }
}

```

Setelah membuat service tersebut kita akan menjalankan proyek *homestay* tersebut dengan mengklik button run pada IntelliJIdea

Setelah itu akses <http://localhost:7078/swagger-ui.html#/> dan halaman dokumentasi API dari proyek yang telah dibuat
Berikut dokumentasi dari webservice yang telah dibuat.

Swagger

Select a spec: default

Homestay [Version 1.0.0]

[Base URL: localhost:7078 /]
<http://localhost:7078/v2/api-docs>

Homestay

homestay-controller Homestay Controller

POST /homestay/create create

GET /homestay/detail/{id} getDetail

GET /homestay/search/kecamatan/{id} searchByKecamatan

kamar-controller Kamar Controller

GET /kamar/detail/{id} getDetail

The screenshot shows a browser window with several tabs open. The active tab is 'Swagger UI' displaying the API documentation for a service. The documentation is organized into sections for different controllers:

- kamar-controller**: Kamar Controller
 - GET /homestay/search/kecamatan/{id} searchByKecamatan
- pemesanan-controller**: Pemesanan Controller
 - POST /pemesanan/create create
 - GET /pemesanan/detail/{id} detail
- upload-controller**: Upload Controller
 - GET /file/download/{fileName} downloadFileFromLocal
 - POST /file/upload upload
- user-controller**: User Controller
 - POST /user/pemilik/login loginPemilik
 - POST /user/pengunjung/login loginPengunjung

At the bottom right of the Swagger UI interface, there is a watermark: 'Activate Windows. Go to Settings to activate Windows.'

Gambar dibawah ini merupakan hasil URI yang telah dibuat dengan menggunakan dokumentasi swagger

The screenshot shows a Swagger UI interface with a sidebar labeled "Models". Below the sidebar, there are five code snippets representing different API models:

```
Models
```

```
BaseResponse ✕ {
  code      integer($int32)
  data      > {...}
  message   string
}
```

```
CreateHomestayRequest ✕ {
  alamat    string
  fasilitas string
  gambar    string
  kecamatan integer($int32)
  keterangan string
  nama     string
  poi      string
  user     integer($int32)
}
```

```
CreatePemesananRequest ✕ {
  homestay  integer($int32)
  in       string
  jumlahKamar integer($int32)
  jumlahTamu integer($int32)
  kamar    integer($int32)
  out      string
  total    integer($int32)
  user     integer($int32)
}
```

```
LoginUserRequest ✕ {
  password string
  username string
}
```

```
RegisterUserRequest ✕ {
  alamat    string
  email    string
  nama    string
  nohp    string
  password string
  role    integer($int32)
  username string
}
```

BAB VI TESTING

6.1 Testing

Pada subbab testing akan diuraikan pelaksanaan pengujian pada aplikasi yang telah dibangun yang terdiri dari tujuan pengujian dan pelaksanaanya, skenario pengujian serta hasil pengujian. Pengujian aplikasi ini dilakukan dengan menggunakan metode *black box testing*.

6.1.1 Tujuan Pengujian

Tujuan dari proses pengujian yang dilakukan pada sistem adalah sebagai berikut:

1. Pengujian dilakukan untuk memastikan apakah sistem/ aplikasi yang dibangun dapat berjalan dengan baik.
2. Pengujian dilakukan untuk memastikan sistem tidak memiliki error
3. Pengujian dilakukan untuk memastikan apakah semua fungsi yang dibangun sudah sesuai dengan kebutuhan pengguna dan dapat berjalan dengan baik.

6.1.2 Skenario Pengujian

Berikut dijelaskan pengujian pada fungsi yang terdapat pada sistem berdasarkan *use case scenario*

6.1.2.1 Skenario Pengujian Login

Pengujian *Login* dilakukan untuk memastikan apakah dapat *login*.

Hasil pengujian dapat dilihat pada tabel berikut:

Tabel 53 Skenario Pengujian Login

| | |
|-----------------------|---|
| Nama Kasus Uji | <i>Login</i> akun untuk admin, pengunjung dan pemilik <i>homestay</i> |
| Tujuan | Menguji apakah admin, pengunjung dan pemilik <i>homestay</i> dapat <i>login</i> |
| Deskripsi | Fungsi ini untuk admin, pengunjung dan pemilik <i>homestay</i> dapat <i>login</i> |
| Kondisi Awal | admin, pengunjung dan pemilik <i>homestay</i> telah terotentikasi dan berada pada tampilan awal untuk admin, pengunjung dan pemilik <i>homestay</i> . |
| Skenario Uji | |

| <ol style="list-style-type: none"> 1. Admin, pengunjung dan pemilik <i>homestay</i> membuka aplikasi 2. Admin, pengunjung dan pemilik <i>homestay</i> memasukan username dan password 3. Sistem menampilkan halaman utama aplikasi | | | |
|---|---|---|------------|
| Kriteria Evaluasi Hasil | | | |
| Admin, pengunjung dan pemilik <i>homestay</i> dapat <i>login</i> | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Admin, pengunjung dan pemilik <i>homestay</i> memasukkan username dan password | Sistem menampilkan halaman utama aplikasi | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Salah satu kolom username tidak diisi | Sistem menampilkan pesan error “Please fill out this field” | Sistem menampilkan pesan error “Please fill out this field”  | Diterima |
| Catatan | | | |
| Admin, pengunjung dan pemilik <i>homestay</i> berhasil <i>login</i> | | | |

6.1.2.2 Skenario Pengujian Registrasi Akun

Pengujian Registrasi akun dilakukan untuk memastikan apakah dapat melakukan registrasi akun.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 54 Skenario Pengujian Registrasi Akun

| Nama Kasus Uji | Registrasi akun untuk pengunjung dan pemilik <i>homestay</i> | | |
|---|---|---|-------------------|
| Tujuan | Menguji apakah pengunjung dan pemilik <i>homestay</i> dapat registrasi akun | | |
| Deskripsi | Fungsi ini untuk pengunjung dan pemilik <i>homestay</i> dapat registrasi akun | | |
| Kondisi Awal | Pengunjung dan pemilik <i>homestay</i> telah terotentikasi dan berada pada tampilan awal untuk pengunjung dan pemilik <i>homestay</i> . | | |
| Skenario Uji | <ol style="list-style-type: none"> 1. Pengunjung dan pemilik <i>homestay</i> membuka aplikasi 2. Pengunjung dan pemilik <i>homestay</i> memilih tombol registrasi 3. Pengunjung dan pemilik <i>homestay</i> mengisi <i>form</i> registrasi akun 4. Sistem menampilkan form <i>login</i> dan registrasi berhasil | | |
| Kriteria Evaluasi Hasil | Pengunjung dan pemilik <i>homestay</i> dapat registrasi akun | | |
| Kasus dan Hasil Uji (Data Normal) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Pengunjung dan pemilik <i>homestay</i> memasukkan data diri | Sistem menampilkan halaman <i>login</i> | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Salah satu kolom username tidak diisi | Sistem menampilkan pesan error “Please fill out this field” | Sistem menampilkan pesan error “Please fill out this field” | Diterima |

| | | | |
|---|--|--|--|
| | | | |
| Catatan | | | |
| Pengunjung dan pemilik <i>homestay</i> berhasil registrasi akun | | | |

6.1.2.3 Skenario Pengujian Pendaftaran Homestay

Pengujian Registrasi *homestay* dilakukan untuk memastikan apakah dapat melakukan registrasi *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 55 Skenario Pengujian Pendaftaran Homestay

| | |
|---|---|
| Nama Kasus Uji | Registrasi <i>homestay</i> untuk pemilik <i>homestay</i> |
| Tujuan | Menguji apakah pemilik <i>homestay</i> dapat registrasi <i>homestay</i> |
| Deskripsi | Fungsi ini untuk pemilik <i>homestay</i> dapat registrasi <i>homestay</i> |
| Kondisi Awal | Pemilik <i>homestay</i> telah terotentikasi dan berada pada tampilan awal untuk pemilik <i>homestay</i> . |
| Skenario Uji | |
| <ol style="list-style-type: none"> 1. Pemilik <i>homestay</i> membuka aplikasi 2. Pemilik <i>homestay</i> memilih tombol registrasi 3. Pemilik <i>homestay</i> mengisi <i>form</i> registrasi <i>homestay</i> 4. Sistem menampilkan halaman <i>homestay</i> | |
| Kriteria Evaluasi Hasil | |
| Pemilik <i>homestay</i> dapat registrasi <i>homestay</i> | |
| Kasus dan Hasil Uji (Data Normal) | |

| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
|---|--|-------------------------------|------------|
| Pemilik <i>homestay</i> memasukkan data <i>homestay</i> | Sistem menampilkan halaman <i>homestay</i> | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| - | - | - | - |
| Catatan | | | |
| Pemilik <i>homestay</i> berhasil registrasi <i>homestay</i> | | | |

6.1.2.4 Skenario Pengujian Melihat Homestay

Pengujian melihat data *homestay* dilakukan untuk memastikan apakah dapat melakukan melihat data *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 56 Skenario Pengujian Melihat Homestay

| | |
|---|---|
| Nama Kasus Uji | Melihat daftar <i>homestay</i> untuk pengunjung |
| Tujuan | Menguji apakah pengunjung dapat melihat data <i>homestay</i> |
| Deskripsi | Fungsi ini untuk pengunjung dapat melihat data <i>homestay</i> |
| Kondisi Awal | Pengunjung telah terotentikasi dan berada pada tampilan awal untuk pengunjung |
| Skenario Uji | |
| <ol style="list-style-type: none"> 1. Pengunjung membuka aplikasi 2. Pengunjung mencari <i>homestay</i> melalui tombol pencarian <i>homestay</i> 3. Pengunjung mengisi <i>form</i> registrasi <i>homestay</i> 4. Sistem menampilkan halaman <i>homestay</i> | |
| Kriteria Evaluasi Hasil | |
| Pengunjung dapat melihat data <i>homestay</i> | |

| Kasus dan Hasil Uji (Data Normal) | | | |
|--|------------------------------------|-------------------------------|-------------------|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Pengunjung memasukkan kata kunci kecamatan <i>homestay</i> | Sistem menampilkan <i>homestay</i> | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Pengunjung berhasil melihat <i>homestay</i> | | | |

6.1.2.5 Scenario Pengujian Melihat Data Homestay

Pengujian melihat data *homestay* dilakukan untuk memastikan apakah dapat melakukan melihat data *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 57 Scenario Pengujian Melihat Data Homestay

| | |
|--------------------------------|---|
| Nama Kasus Uji | Melihat data <i>homestay</i> untuk pengunjung |
| Tujuan | Menguji apakah pengunjung dapat melihat data <i>homestay</i> |
| Deskripsi | Fungsi ini untuk pengunjung dapat melihat data <i>homestay</i> |
| Kondisi Awal | Pengunjung telah terotentikasi dan berada pada tampilan awal untuk pengunjung |
| Skenario Uji | <ol style="list-style-type: none"> 1. Pengunjung membuka aplikasi 2. Pengunjung mencari <i>homestay</i> melalui tombol pencarian <i>homestay</i> 3. Pengunjung mengisi <i>form</i> registrasi <i>homestay</i> 4. Sistem menampilkan halaman <i>homestay</i> |
| Kriteria Evaluasi Hasil | Pengunjung dapat melihat data <i>homestay</i> |

| Kasus dan Hasil Uji (Data Normal) | | | |
|--|------------------------------------|-------------------------------|-------------------|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Pengunjung memasukkan kata kunci kecamatan <i>homestay</i> | Sistem menampilkan <i>homestay</i> | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Pengunjung berhasil melihat <i>homestay</i> | | | |

6.1.2.6 Scenario Pengujian Mengelola Data Homestay

Pengujian mengelola data *homestay* dilakukan untuk memastikan apakah dapat melakukan melihat data *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 58 Scenario Pengujian Mengelola Data Homestay

| | |
|--------------------------------|--|
| Nama Kasus Uji | Mengelola data <i>homestay</i> untuk admin |
| Tujuan | Menguji apakah admin dapat mengelola data <i>homestay</i> |
| Deskripsi | Fungsi ini untuk admin dapat mengelola data <i>homestay</i> |
| Kondisi Awal | Admin telah terotentikasi dan berada pada tampilan awal untuk admin |
| Skenario Uji | <ol style="list-style-type: none"> 1. Admin membuka aplikasi 2. Admin memilih tombol tambah <i>homestay</i> 3. Admin mengisi form tambah <i>homestay</i> 4. Admin memilih tombol edit <i>homestay</i> 5. Admin mengedit <i>homestay</i> 6. Admin memilih tombol hapus <i>homestay</i> 7. Admin mencetak data <i>homestay</i> 8. Sistem menampilkan halaman <i>homestay</i> |
| Kriteria Evaluasi Hasil | |
| | Admin dapat mengelola data <i>homestay</i> |

| Kasus dan Hasil Uji (Data Normal) | | | |
|---|---|---|-------------------|
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Admin mengisi form untuk tambah <i>homestay</i> | Sistem berhasil menambah data <i>homestay</i> | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Tidak memasukkan data yang diperlukan | Sistem menampilkan pesan error | Sistem menampilkan pesan error localhost:8080 says Data Blessing Homestay gagal disimpan! OK | - |
| Catatan | | | |
| Admin berhasil mengelola data <i>homestay</i> | | | |

6.1.2.7 Scenario Pengujian Melakukan Pemesanan Homestay

Pengujian melakukan pemesanan *homestay* dilakukan untuk memastikan apakah dapat melakukan pemesanan *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 59 Scenario Pengujian Melakukan Pemesanan Homestay

| | |
|-----------------------|---|
| Nama Kasus Uji | Melakukan pemesanan <i>homestay</i> |
| Tujuan | Menguji apakah pengunjung dapat Melakukan pemesanan <i>homestay</i> |
| Deskripsi | Fungsi ini untuk pengunjung dapat Melakukan pemesanan <i>homestay</i> |
| Kondisi Awal | Pengunjung telah terotentikasi dan berada pada tampilan awal untuk pengunjung |
| Skenario Uji | |

| 1. Pengunjung membuka aplikasi 2. Pengunjung mencari homestay yang ingin dipesan 3. Pengunjung melihat detail homestay 4. Pengunjung mengklik button pesan 5. Pengunjung mengisi form pemesanan | | | |
|---|--|--------------------------------|------------|
| Kriteria Evaluasi Hasil | | | |
| Pengunjung dapat melakukan pemesanan <i>homestay</i> | | | |
| Kasus dan Hasil Uji (Data Normal) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Pengunjung mengisi data form pemesanan | Sistem berhasil menyimpan data pemesanan | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Tidak memasukkan data yang diperlukan | Sistem menampilkan pesan error | Sistem menampilkan pesan error | - |
| Catatan | | | |
| Pengunjung berhasil melakukan pemesanan <i>homestay</i> | | | |

6.1.2.8 Scenario Pengujian Konfirmasi Pendaftaran Homestay

Pengujian konfirmasi pendaftaran *homestay* dilakukan untuk memastikan apakah dapat konfirmasi pendaftaran *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 60 Scenario Pengujian Konfirmasi Pendaftaran Homestay

| | |
|-----------------------|---|
| Nama Kasus Uji | <i>konfirmasi</i> pendaftaran homestay oleh admin |
| Tujuan | Menguji apakah admin dapat <i>konfirmasi</i> pendaftaran homestay |
| Deskripsi | Fungsi ini untuk admin dapat <i>konfirmasi</i> pendaftaran homestay |

| Kondisi Awal | admin telah terotentikasi dan berada pada tampilan awal untuk admin | | |
|---|---|--|------------|
| Skenario Uji | | | |
| | <ol style="list-style-type: none"> 1. Admin membuka aplikasi website 2. Admin memilih menu daftar homestay 3. Admin memilih homestay yang terdaftar 4. Admin memberikan persetujuan terhadap pendaftaran <i>homestay</i> 5. Admin mengklik tombol beri persetujuan | | |
| Kriteria Evaluasi Hasil | | | |
| | Admin dapat melakukan konfirmasi pendaftaran <i>homestay</i> | | |
| Kasus dan Hasil Uji (Data Normal) | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
| Admin memilih status persetujuan “diterima” | Sistem berhasil menyimpan data pemesanan | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Admin memilih status persetujuan “ditolak” | Sistem menampilkan pesan error | Sistem menampilkan pesan error Status Persetujuan ● Di tolak | - |
| Catatan | | | |
| | Admin berhasil mengkonfirmasi pendaftaran homestay | | |

6.1.2.9 Scenario Pengujian Konfirmasi Pemesanan Homestay

Pengujian konfirmasi pemesanan *homestay* dilakukan untuk memastikan apakah dapat konfirmasi pemesanan *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 61 Scenario Pengujian Konfirmasi Pemesanan Homestay

| Nama Kasus Uji | <i>konfirmasi</i> pemesanan homestay oleh admin | | | | | |
|--|--|---|-------------------|--|--|--|
| Tujuan | Menguji apakah admin dapat <i>konfirmasi</i> pemesanan homestay | | | | | |
| Deskripsi | Fungsi ini untuk admin dapat <i>konfirmasi</i> pemesanan homestay | | | | | |
| Kondisi Awal | admin telah terotentikasi dan berada pada tampilan awal untuk pengunjung | | | | | |
| Skenario Uji | | | | | | |
| <ol style="list-style-type: none"> 1. Admin membuka aplikasi website 2. Admin memilih menu daftar pemesanan 3. Admin pemesanan homestay 4. Admin memberikan persetujuan terhadap pemesanan <i>homestay</i> 5. Admin mengklik tombol beri konfirmasi | | | | | | |
| Kriteria Evaluasi Hasil | | | | | | |
| Admin dapat melakukan konfirmasi pemesanan <i>homestay</i> | | | | | | |
| Kasus dan Hasil Uji (Data Normal) | | | | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan | | | |
| Admin memilih status transaksi “diterima” | Sistem berhasil menyimpan data pemesanan | Sesuai dengan yang diharapkan | Diterima | | | |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | | | | |
| Admin memilih status persetujuan “ditolak” | Sistem menampilkan pesan error | Sistem menampilkan pesan error • Di tolak | - | | | |

| Catatan |
|--|
| Admin berhasil mengkonfirmasi pemesanan homestay |

6.1.2.10 Scenario Pengujian Melihat Lokasi Homestay

Pengujian melihat lokasi *homestay* dilakukan untuk memastikan apakah dapat melakukan melihat lokasi *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 62 Scenario Pengujian Melihat Lokasi Homestay

| | |
|--------------------------------|--|
| Nama Kasus Uji | Melihat lokasi <i>homestay</i> untuk pengunjung |
| Tujuan | Menguji apakah pengunjung dapat melihat lokasi <i>homestay</i> |
| Deskripsi | Fungsi ini untuk pengunjung dapat melihat lokasi <i>homestay</i> |
| Kondisi Awal | Pengunjung telah terotentikasi dan berada pada tampilan awal untuk pengunjung |
| Skenario Uji | <ol style="list-style-type: none"> 1. Pengunjung membuka aplikasi 2. Pengunjung mencari <i>homestay</i> melalui tombol pencarian <i>homestay</i> 3. Pengunjung memilih salah satu <i>homestay</i> 4. Sistem menampilkan detail <i>homestay</i> 5. Pengunjung klik tombol lihat lokasi |
| Kriteria Evaluasi Hasil | |
| | Pengunjung dapat melihat lokasi <i>homestay</i> |
| | Pengunjung berhasil melihat lokasi <i>homestay</i> |

6.1.2.11 Scenario Pengujian Mencari Homestay

Pengujian mencari *homestay* dilakukan untuk memastikan apakah dapat melakukan mencari *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 63 Scenario Pengujian Mencari Homestay

| Nama Kasus Uji | Mencari <i>homestay</i> untuk pengunjung | | | | | |
|---|---|-------------------------------|-------------------|--|--|--|
| Tujuan | Menguji apakah pengunjung dapat mencari <i>homestay</i> | | | | | |
| Deskripsi | Fungsi ini untuk pengunjung dapat mencari <i>homestay</i> | | | | | |
| Kondisi Awal | Pengunjung telah terotentikasi dan berada pada tampilan awal untuk pengunjung | | | | | |
| Skenario Uji | | | | | | |
| <ol style="list-style-type: none"> 1. Pengunjung membuka aplikasi 2. Pengunjung mencari <i>homestay</i> melalui tombol pencarian <i>homestay</i> 3. Pengunjung memilih <i>homestay</i> berdasarkan kecamatan 4. Sistem menampilkan <i>homestay</i> yang dipilih | | | | | | |
| Kriteria Evaluasi Hasil | | | | | | |
| Pengunjung dapat melihat <i>homestay</i> yang dicari | | | | | | |
| Kasus dan Hasil Uji (Data Normal) | | | | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan | | | |
| Pengunjung memasukkan kata kunci kecamatan <i>homestay</i> | Sistem menampilkan <i>homestay</i> | Sesuai dengan yang diharapkan | Diterima | | | |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | | | | |
| Pengunjung berhasil melihat <i>homestay</i> | | | | | | |

6.1.2.12 Scenario Pengujian Mengelola Kamar

Pengujian mengelola data kamar dilakukan untuk memastikan apakah dapat mengelola data kamar.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 64 Scenario Pengujian Mengelola Data Kamar

| Nama Kasus Uji | mengelola data kamar untuk pemilik <i>homestay</i> | | | | | |
|--|---|--------------------------------|------------|--|--|--|
| Tujuan | Menguji apakah pemilik <i>homestay</i> dapat mengelola data kamar | | | | | |
| Deskripsi | Fungsi ini untuk pemilik homestay dapat mengelola data kamar | | | | | |
| Kondisi Awal | Pemilik homestay telah terotentikasi dan berada pada tampilan awal untuk Pemilik homestay | | | | | |
| Skenario Uji | | | | | | |
| <ol style="list-style-type: none"> 1. Pemilik homestay membuka aplikasi 2. Pemilik homestay memilih kamar yang ingin di update 3. Pemilik homestay mengisi form 4. Pemilik homestay mengklik button save | | | | | | |
| Kriteria Evaluasi Hasil | | | | | | |
| Pemilik homestay dapat mengelola data kamar | | | | | | |
| Kasus dan Hasil Uji (Data Normal) | | | | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan | | | |
| Pemilik homestay mengisi form untuk tambah <i>kamar</i> | Sistem berhasil menambah data <i>kamar</i> | Sesuai dengan yang diharapkan | Diterima | | | |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | | | | |
| Tidak mengisi form yang dibutuhkan | Sistem menampilkan pesan error | Sistem menampilkan pesan error | - | | | |
| Catatan | | | | | | |

| |
|--|
| Pemilik homestay berhasil mengelola data <i>homestay</i> |
|--|

6.1.2.13 Scenario Pengujian Mengelola Laporan Pemesanan Homestay

Pengujian mengelola laporan pemesanan dilakukan untuk memastikan apakah dapat laporan pemesanan

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 65 Scenario Pengujian Mengelola Laporan Pemesanan Homestay

| Nama Kasus Uji | mengelola <i>laporan pemesanan</i> untuk admin | | | | | |
|---|---|-------------------------------|-------------------|--|--|--|
| Tujuan | Menguji apakah admin dapat mengelola <i>laporan pemesanan</i> | | | | | |
| Deskripsi | Fungsi ini untuk admin dapat mengelola <i>laporan pemesanan</i> | | | | | |
| Kondisi Awal | admin telah terotentikasi dan berada pada tampilan awal untuk admin | | | | | |
| Skenario Uji | | | | | | |
| <ol style="list-style-type: none"> 1. admin membuka aplikasi 2. admin memilih menu pemesanan 3. admin mengklik button cetak 4. sistem menampilkan laporan pemesanan | | | | | | |
| Kriteria Evaluasi Hasil | | | | | | |
| Admin dapat mencetak laporan pemesanan | | | | | | |
| Kasus dan Hasil Uji (Data Normal) | | | | | | |
| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan | | | |
| Data pemesanan | Sistem berhasil menampilkan data pemesanan <i>homestay</i> | Sesuai dengan yang diharapkan | Diterima | | | |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | | | | |

| | | | |
|---|--------------------------------|--------------------------------|---|
| Tidak ada pemesanan homestay | Sistem menampilkan pesan error | Sistem menampilkan pesan error | - |
| Catatan | | | |
| Admin berhasil melihat dan mencetak laporan pemesanan | | | |

6.1.2.14 Scenario Pengujian Melakukan Pembayaran

Pengujian melakukan pembayaran *homestay* dilakukan untuk memastikan apakah dapat melakukan pembayaran *homestay*.

Hasil pengujian dapat dilihat pada tabel berikut.

Tabel 66 Scenario Pengujian Melakukan Pembayaran

| | |
|---|---|
| Nama Kasus Uji | Melakukan pembayaran <i>homestay</i> |
| Tujuan | Menguji apakah pengunjung dapat Melakukan pembayaran <i>homestay</i> |
| Deskripsi | Fungsi ini untuk pengunjung dapat Melakukan pembayaran <i>homestay</i> |
| Kondisi Awal | Pengunjung telah terotentikasi dan berada pada tampilan awal untuk pengunjung |
| Skenario Uji | |
| <ol style="list-style-type: none"> 1. Pengunjung membuka aplikasi 2. Pengunjung mencari homestay yang ingin dipesan 3. Pengunjung melihat detail homestay 4. Pengunjung mengklik button pesan 5. Pengunjung mengisi form pemesanan 6. Sistem menampilkan halaman selanjutnya yaitu pembayaran 7. Pengunjung memilih type pembayaran 8. Pengunjung mengisi form dan kirim bukti pembayaran | |
| Kriteria Evaluasi Hasil | |
| Pengunjung dapat melakukan pembayaran <i>homestay</i> | |
| Kasus dan Hasil Uji (Data Normal) | |

| Data Masukan | Yang diharapkan | Pengamatan | Kesimpulan |
|--|--|--------------------------------|------------|
| Pengunjung mengisi data form pembayaran | Sistem berhasil menyimpan data pemesanan | Sesuai dengan yang diharapkan | Diterima |
| Kasus dan Hasil uji coba (Data Tidak Normal) | | | |
| Tidak memasukkan data yang diperlukan | Sistem menampilkan pesan error | Sistem menampilkan pesan error | - |
| Catatan | | | |
| Pengunjung berhasil melakukan pembayaran <i>homestay</i> | | | |

BAB VII

HASIL DAN PEMBAHASAN

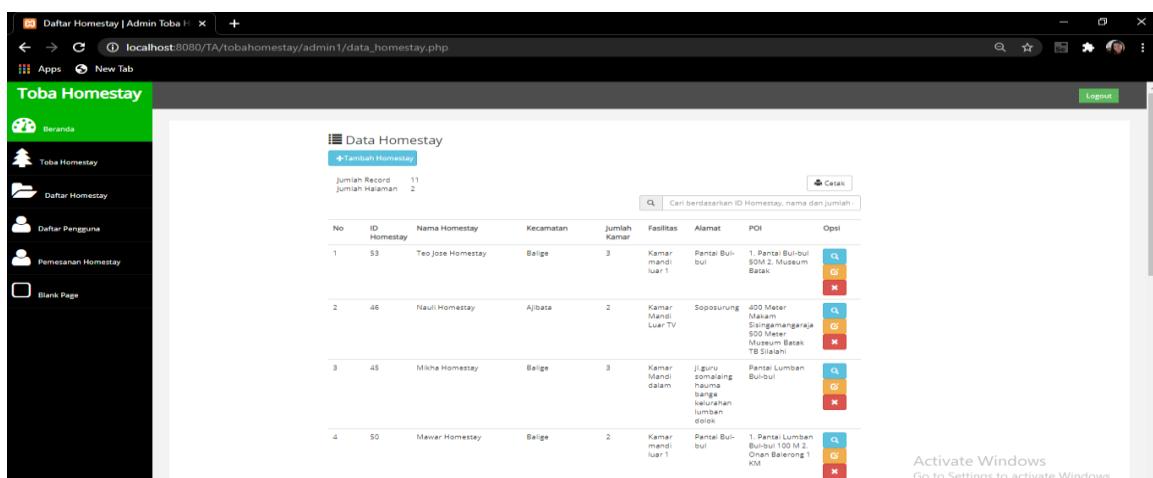
Pada bab ini dijelaskan pembahasan dari pembangunan Apliasi Homestay Toba Samosir Berbasis Android. Pembangunan aplikasi ini menggunakan *library Spring-boot, Maven, Firebase, dan Kotlin*. Hasil dari pembangunan aplikasi ini adalah sebuah aplikasi homestay untuk Toba Samosir yang menyediakan layanan pemesanan.

7.1 Hasil Admin

Pada subbab ini dijelaskan mengenai hasil untuk admin yang dihasilkan dengan berbasis web. Dimana admin dapat mengelola data *homestay*, konfirmasi pendaftaran *homestay*, konfirmasi pemesanan *homestay*, dan menghapus daftar pengguna.

7.1.1 Hasil Mengelola Data Homestay

Pada mengelola data homestay tersebut admin dapat melakukan *create, update, delete*.



The screenshot shows a web-based administration interface for 'Toba Homestay'. The left sidebar has a green header 'Beranda' and a black background with white icons for 'Toba Homestay', 'Daftar Homestay', 'Daftar Pengguna', 'Pemesanan Homestay', and 'Blank Page'. The main content area has a white header 'Data Homestay' with a 'Tambah Homestay' button. Below it is a search bar with placeholder text 'Caril berdasarkan ID Homestay, nama dan jumlah' and a 'Cetak' button. A table lists four homestays:

| No | ID Homestay | Nama Homestay | Kecamatan | Jumlah Kamar | Fasilitas | Alamat | POI | Opsi |
|----|-------------|-------------------|-----------|--------------|----------------------|---|---|---|
| 1 | 93 | Teo Jose Homestay | Balige | 3 | Kamar mandi luar I | Pantai Bulbul | 1. Pantai Bulbul 2. Museum Batak |   |
| 2 | 46 | Nauli Homestay | Ajibata | 2 | Kamar Mandi, Luar TV | Soposurung | 400 Meter Sekitar Giamanggaraja 500 Meter Kota Batak TB Silalahi |   |
| 3 | 45 | Milka Homestay | Balige | 3 | Kamar Mandi dalam | Jl. guru somaling hausen Balige kelurahan tumbaran dorok | Pantai Lumban Bulbul |   |
| 4 | 50 | Maver Homestay | Balige | 2 | Kamar mandi luar I | Pantai Bulbul | 1. Pantai Lumban Bulbul 100 M 2. Ohan Balerong 1 KM |   |

Gambar 89 Hasil Mengeola Data Homestay

Berikut hasil tambah homestay

The screenshot shows a web-based administration interface for 'Toba Homestay'. On the left is a sidebar with icons for Beranda, Toba Homestay, Daftar Homestay, Daftar Pengguna, Pemesanan Homestay, and Blank Page. The main content area has a title '+ Tambah Homestay'. It contains several input fields: 'Nama Homestay' (Name), 'Jumlah' (Quantity), 'Deskripsi' (Description), 'Kecamatan' (Subdistrict), 'District' (District), 'Gambar' (Image) with a file selection button, 'Fasilitas' (Facilities), 'Alamat' (Address), and 'POI' (Point of Interest). At the bottom are 'Simpan' (Save) and 'Batal' (Cancel) buttons.

Gambar 90 Hasil Tambah Homestay

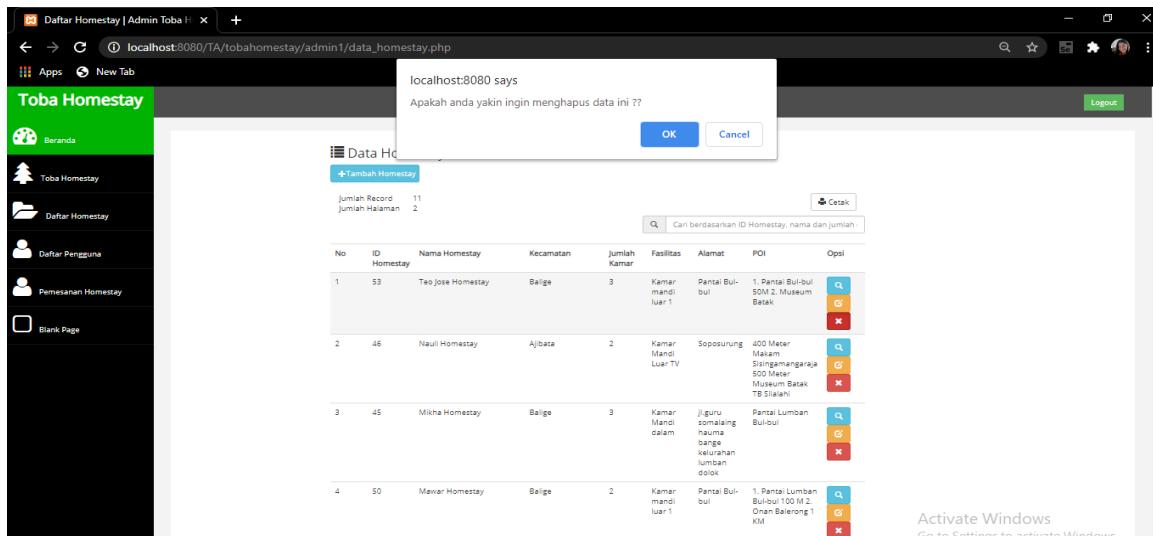
Apabila admin ingin melihat detail homestay, admin dapat mengklik icon *view* maka aplikasi akan menampilkan seperti gambar dibawah ini

This screenshot shows the 'Detail Homestay' page. The sidebar on the left is identical to the previous one. The main content displays the details of a homestay with ID 53, named 'Teo Jose Homestay', located in 'Aijibata' with 3 rooms. The description is 'Aman, nyaman dan lingkungan asri'. It shows a thumbnail image, facilities like 'Kamar mandi luar 1', address 'Pantai Bul-bul', and POI '1. Pantai Bul-bul 50M 2. Museum Batak'.

Gambar 91 Hasil Lihat Detail Homestay

This screenshot shows the 'Edit Homestay' page. The sidebar is the same. The form is identical to the 'Tambah Homestay' one, with fields for Name, Subdistrict, Room Quantity, Description, Image, Facilities, Address, and POI. The 'Nama' field is populated with 'Teo Jose Homestay', 'Kecamatan' is 'Aijibata', 'Jumlah Kamar' is '3', 'Deskripsi' is 'Aman, nyaman dan lingkungan asri', and the 'POI' field contains '1. Pantai Bul-bul 50M 2. Museum Batak'.

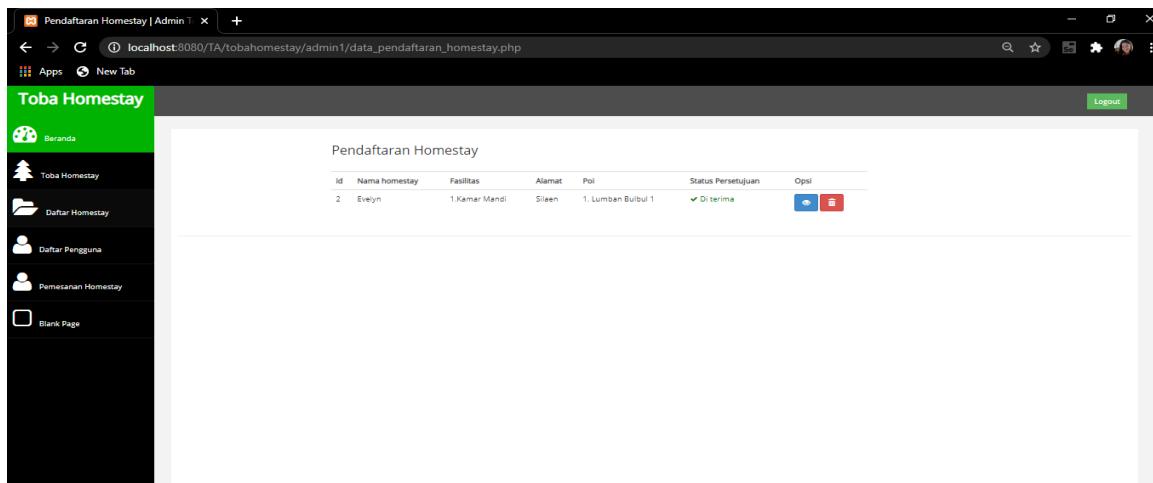
Gambar 92 Hasil Edit Homestay



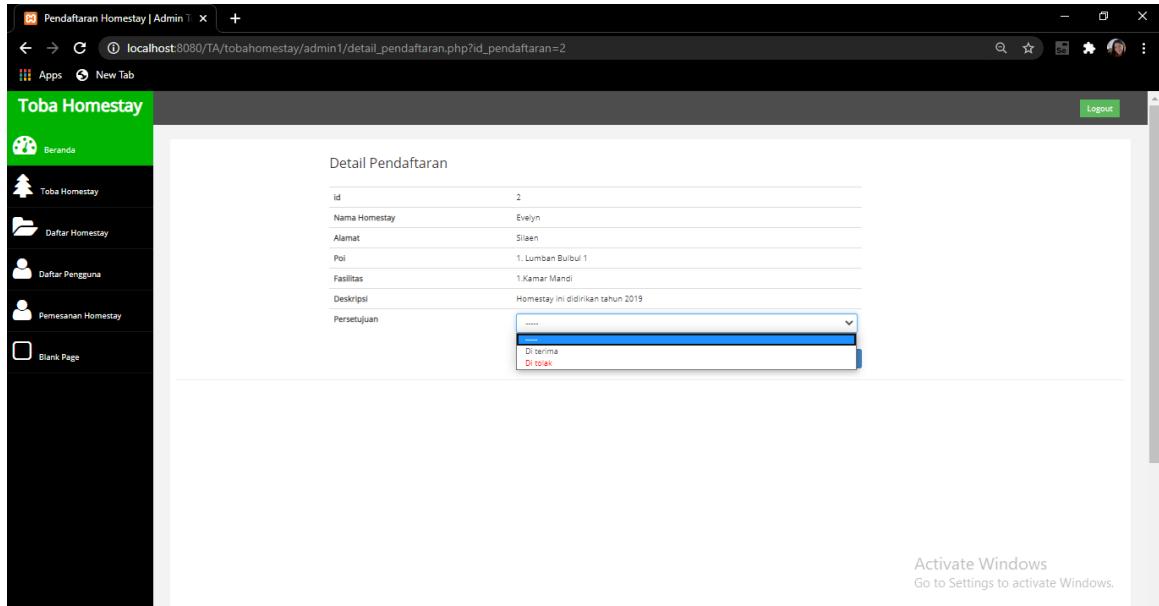
Gambar 93 Hasil Hapus Homestay

7.1.2 Hasil Pendaftaran Homestay

Pada menu pendaftaran homestay admin dapat memberikan persetujuan terhadap pendaftaran homestay



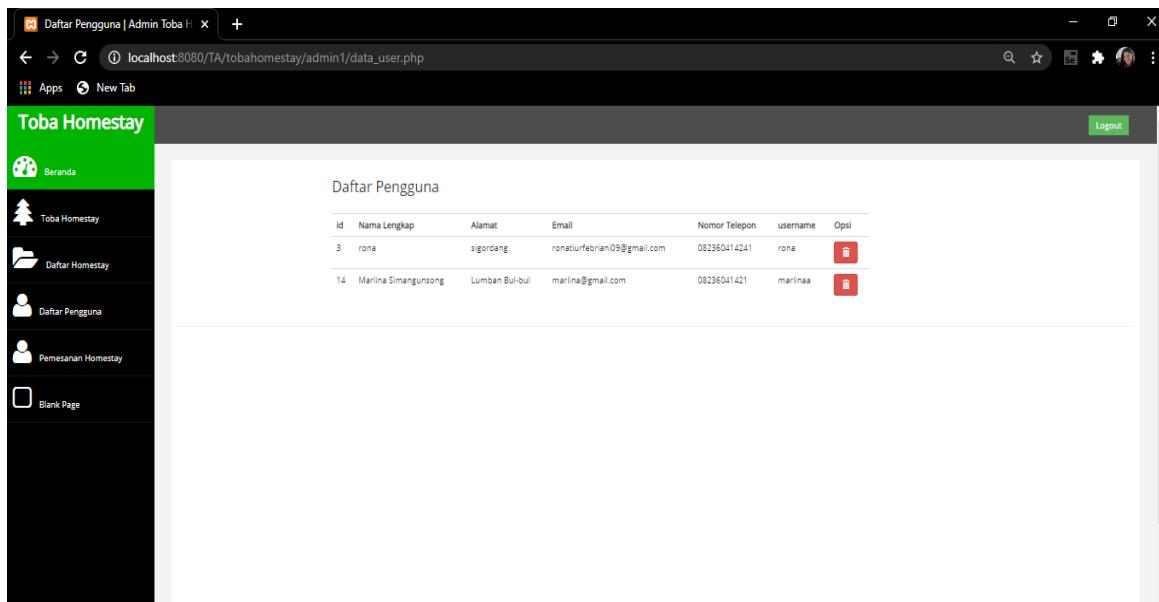
Gambar 94 Hasil Pendaftaran Homestay



Gambar 95 Hasil Memberikan Persetujuan

7.1.3 Hasil Daftar Pengguna

Pada menu ini admin dapat menghapus daftar pengguna aplikasi *homestay*



Gambar 96 Hasil Daftar Pengguna

7.1.4 Hasil Pemesanan Homestay

Pada menu ini admin dapat melakukan konfirmasi transaksi pemesanan homestay yang dilakukan oleh pengunjung dan dapat mencetak laporan pemesanan

| Id | Nama Pemesan | Room Type | Jumlah Tamu | Metode Pembayaran | Total Harga | Status | Bukti Pembayaran | Opsi |
|----|-----------------------------|-----------|-------------|-------------------|-------------|-------------|------------------|---------------------------------------|
| 2 | Marilina Simangunsong | Exclusive | 4 | Bayar DP | 2500000 | Belum lunas | JPG | <button>✓ Berikan Konfirmasi</button> |
| 1 | Ronasur Febriani Lumbanggao | ttt | 4 | ss | 2500000 | Lunas | sss | <button>✓ Transaksi Selesai</button> |

Gambar 97 Hasil Pemesanan Homestay

Berikut hasil konfirmasi pemesanan *homestay*

Konfirmasi Transaksi

| | |
|-------------------|---|
| Nama | Marilina Simangunsong |
| Tujuan | Exclusive |
| Jumlah Tamu | 4 orang |
| Total Harga | Rp. 2.500.000 |
| Keterangan Tamu | sekit |
| Metode Pembayaran | Bayar DP |
| Status Transaksi | <input type="button" value="Belum lunas ---"/> <input type="button" value="Belum lunas ---"/> <input type="button" value="Di terima"/> <input type="button" value="Lunas"/> |

Gambar 98 Hasil Konfirmasi Pemesanan

Berikut Hasil Cetak Laporan pemesanan

| ID Pemesanan | Nama Pemesan | Room Type | Jumlah Tamu | Metode Pembayaran | Total Harga | Status Pembayaran | Bukti Pembayaran |
|--------------|----------------------------------|-----------|-------------|-------------------|-------------|-------------------|------------------|
| 1 | Romarie Febriani Lumbangcasih | B | 4 | ss | 2500000 | Lunas | sss |
| 2 | Melina Simangunsong | Exclusive | 4 | Bayar DP | 2500000 | Belum Lunas | JPG |

Gambar 99 Hasil Cetak Laporan pemesanan

7.2 Hasil Pengunjung

7.2.1 Splashscreen Pengunjung

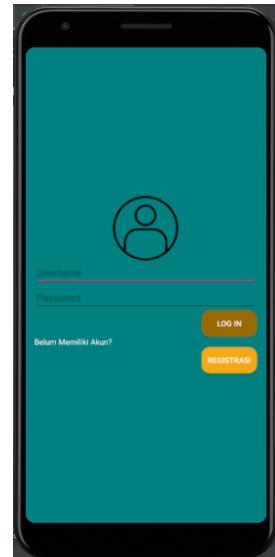
Pada menu ini pengunjung pertama kali melihat splashscreen aplikasi



Gambar 100 Splashscreen Pengunjung

7.2.2 Login Pengunjung

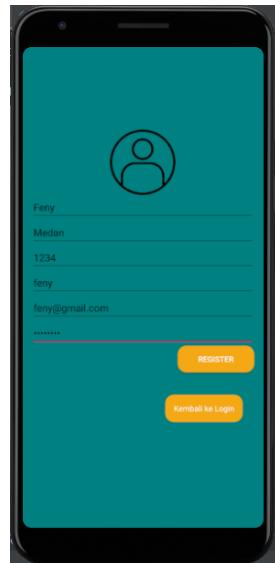
Pada menu ini pengunjung harus melakukan login terlebih dahulu



Gambar 101 Login Pengunjung

7.2.3 Register Pengunjung

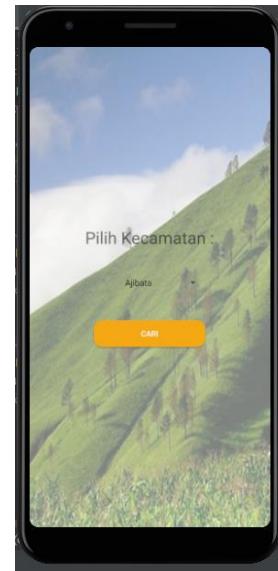
Pada menu ini pengunjung yang belum memiliki akun harus registrasi akun



Gambar 102 Register Pengunjung

7.2.4 Home Pengunjung

Pada menu ini pengunjung akan melihat halaman awal



Gambar 103 Home Pengunjung

7.2.5 List Homestay

Pada menu ini pengunjung akan melihat list homestay



Gambar 104 List Homestay

7.2.6 Detail Homestay

Pada menu ini pengunjung akan melihat detail homestay



Gambar 105 Detail Homestay

7.2.7 Tampilan Lokasi

Pada menu ini pengunjung akan melihat lokasi homestay



Gambar 106 Tampilan Lokasi

7.2.8 Pemesanan

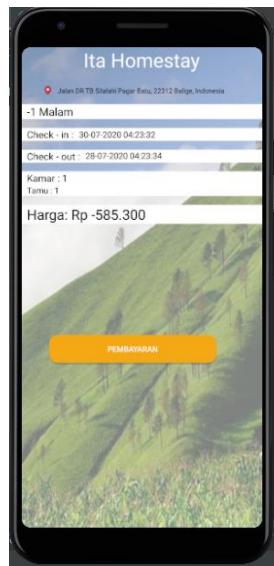
Pada menu ini pengunjung akan melakukan pemesanan homestay



Gambar 107 Pemesanan

7.2.9 Rincian Akhir

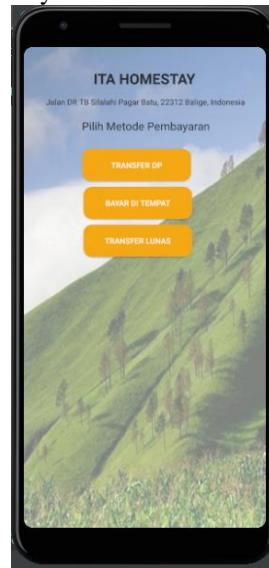
Pada menu ini pengunjung akan melihat rincian akhir pemesanan



Gambar 108 Rincian Akhir

7.2.10 Pembayaran

Pada menu ini pengunjung akan melakukan proses pembayaran



Gambar 109 Pembayaran

7.3 Hasil Pemilik Homestay

7.3.1 Hasil Splashscreen

Pada menu ini pemilik homestay akan melihat splashscreen



Gambar 110 Hasil Tampilan Splash Screen

7.3.2 Hasil Melakukan Login

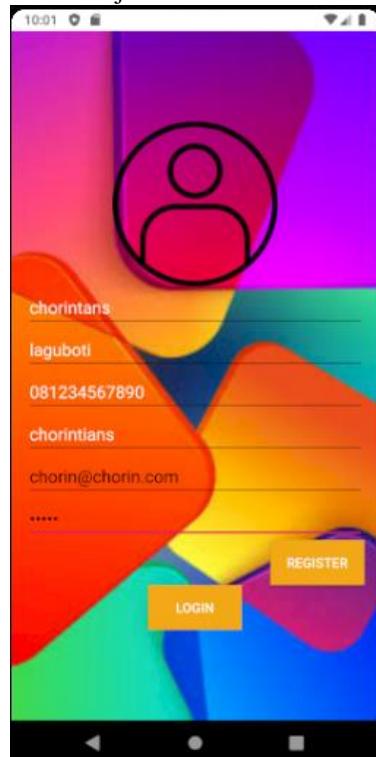
Pada menu ini pemilik homestay akan login



Gambar 111 Hasil Tampilan Login

7.3.3 Hasil Melakukan Registrasi

Pada menu ini pemilik homestay harus login terlebih dahulu jika tidak memiliki akun



Gambar 112 Hasil Tampilan Register

7.3.4 Hasil Mengelola Kamar

Pada menu ini pemilik homestay mengelola kamar

| Daftar Kamar | |
|--------------|---------|
| Harga | 585300 |
| Jumlah Bed | 1 |
| Harga | 1174600 |
| Jumlah Bed | 2 |
| Harga | 20000 |
| Jumlah Bed | 1 |
| Harga | 200000 |
| Jumlah Bed | 2 |
| Harga | 200000 |
| Jumlah Bed | 1 |
| Harga | 200000 |
| Jumlah Bed | 2 |
| Harga | 200000 |
| Jumlah Bed | 1 |
| Harga | 200000 |
| Jumlah Bed | 2 |

Gambar 113 Hasil Daftar Kamar

BAB VIII

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dan saran selama proses pengerjaan Tugas Akhir ini yaitu pembuatan aplikasi android *homestay* di Kabupaten Toba Samosir.

8.1 Kesimpulan

Berdasarkan proses pembangunan dan pengujian yang telah dilakukan, berikut merupakan kesimpulan yang dapat diperoleh oleh penulis dari pengerjaan Tugas Akhir :

1. Aplikasi Homestay Toba Samosir adalah aplikasi yang digunakan sebagai alat untuk mencari homestay di daerah Toba Samosir. Aplikasi ini memungkinkan pengunjung wisata untuk memesan homestay sesuai kebutuhan.
2. Pembayaran pemesanan homestay dapat dilakukan secara langsung, atau dapat transfer ke ATM terdekat dan mengirim bukti pembayaran.

5.2 Saran

Saran yang dapat dipertimbangkan untuk Tugas Akhir berikutnya adalah aplikasi Homestay yang telah dibangun memiliki kekurangan terutama pada antarmuka pengguna.

Bagi pengembang aplikasi ini dapat membuat Point of Interest dalam bentuk peta dan dapat diklik untuk melihat POI. Untuk pengembangan selanjutnya supaya untuk Maps bisa dilakukan tanpa penghitungan secara manual.

Daftar Pustaka

- [1] R. A. Siregar, H. W. Wiranegara, and H. Hermantoro, “Pengembangan Kawasan Pariwisata Danau Toba, Kabupaten Toba Samosir,” *Tataloka*, vol. 20, no. 2, p. 100, 2018.
- [2] Ismayanti, *Pengantar Pariwisata*. Jakarta: Grasindo, 2011.
- [3] S. F. Balica *et al.*, “POTENSI DAN KENDALA PENGEMBANGAN PARIWISATA BERBASIS KEKAYAAN ALAM DENGAN PENDEKATAN MARKETING PLACES (STUDI KASUS PENGEMBANGAN PARIWISATA DI KABUPATEN BOJONEGORO),” *運輸と経済*, vol. 2014, no. June, pp. 1–2, 2014.
- [4] B. Derviş, “PERANCANGAN HOMESTAY DI DESA WISATA NGLINGGO DENGAN PENDEKATAN ARSITEKTUR TROPIS DAN PENERAPAN KONSEP EDUKATIF DAN KOMUNIKATIF Design,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.
- [5] D. H. Bondan, “Sistem Informasi Obyek Wisata (Tour Guide) Secara Real Time Menggunakan GPS Di Bogor Via Mobile Berbasis Android,” *Fakt. Exacta*, vol. 7, no. 1, pp. 59–71, 2014.
- [6] K. Indrawan, E. Putra, N. Santoso, and E. Santoso, “Pengembangan Aplikasi Pelelangan Ternak Burung Lovebird berbasis Android,” vol. 3, no. 7, pp. 6887–6895, 2019.
- [7] D. Putri, P. Rais, E. B. Setiawan, and J. D. Bandung, “TEMPAT WISATA ALAM DI PULAU TIMOR BERBASIS ANDROID Program Studi Teknik Informatika , Universitas Komputer Indonesia.”
- [8] B. W. Niko Sumanda Sibarani¹, Ghifari Munawar², “Analisis Performa Aplikasi Android Pada Bahasa Pemrograman Java dan Kotlin,” *J. Inform.*
- [9] E. Sutanta and K. Mustofa, “Identifying The Needs of Web Service to Data Synchronization Between Information Systems as E-Government Ecosystem at Bantul-Yogyakarta,” *Tek. Inform. - STMIK Bandung*, vol. 2, no. 3, pp. 20–26, 2012.
- [10] andi Juansyah, “Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System (A-GPS) Dengan Platform Android,” *J. Ilm. Komput. dan Inform.*, vol. 1, no. 1, pp. 1–8, 2015.

- [11] A. C. Baramuli, A. Nugroho, and A. A. Setiyanti, “Implementasi Model View Presenter dan Object Relational Mapping NHibernate pada Aplikasi eStop Card berbasis Web (Studi Kasus: PT. XYZ Jakarta),” *J. Inform.*, vol. 8, no. 2, 2013.
- [12] B. Adi Pranata, A. Hijriani, and A. Junaidi, “Perancangan Application Programming Interface (Api) Berbasis Web Menggunakan Gaya Arsitektur Representational State Transfer (Rest) Untuk Pengembangan Sistem Informasi Administrasi Pasien Klinik Perawatan Kulit,” *J. Komputasi*, vol. 6, no. 1, pp. 33–42, 2018.
- [13] W. Nugraha and M. Syarif, “Penerapan Metode Prototype Dalam Perancangan Sistem Informasi Penghitungan Volume Dan Cost Penjualan Minuman Berbasis Website,” *JUSIM (Jurnal Sist. Inf. Musirawas)*, vol. 3, no. 2, pp. 94–101, 2018.
- [14] A. A. Pradipta, Y. A. Prasetyo, and N. Ambarsari, “Pengembangan Web E-Commerce Bojana Sari Menggunakan Metode Prototype,” *e-Proceeding Eng.*, vol. 2, no. 1, pp. 1042–1056, 2015.
- [15] M. Khosrow-Pour, *Encyclopedia of Information Science and Technology (5 Volumes)*. United States of America: IGI Global, 2005.
- [16] M. K. Canggih Ajika Pamungkas, “Aplikasi Penghitung Jarak Koordinat Berdasarkan Latitude Dan Longitude Dengan Metode Euclidean Distance Dan Metode Haversine,” *J. Inf. Politek. Indonusa Surakarta*, vol. 5, pp. 8–13, 2019.

LAMPIRAN

1. Lampiran Hasil Wawancara

Berikut dilampirkan hasil wawancara dengan beberapa pemilik *homestay* di Pantai Bul-bul Balige.

SCRIPT WAWANCARA DENGAN PEMILIK HOMESTAY BERINGIN

Bagaimana pengaruh homestay terhadap ekonomi?

“Menjanjikan kok. Kita kan kebetulan ada kamar kamar kosong. Paling 2x seminggu kalo hari biasa, tapi kalau hari besar atau hari merah gitu kan rame. Kadang juga mau sampe di buat di depan ini karna rame. Disini ada 3 kamar. Ada kamar yg untuk 4 orang yg lainnya bisa 5. Berkeluarga, suami istri, pernah kemarin bule. “

Bagaimana pengaruh homestay ini terhadap wisatawan ?

“Jelaslah. Karna kan kita punya fasilitas air panas, artinya kalo mereka habis main air, mandi-mandi gitu kan ya gak bayar lagi artinya kan mereka bisa langsung mandi kesini ga bayar lagi. Kalau disini kan kalau mau mandi bayar, buang air kecil bayar, di homestay ini gak lagi. Itu dia daya tariknya. Lumayan terkenal homestay ini”

Fasilitas apa saja yang disediakan?

“fasilitasnya ya tempat tidurlah, air panas, kamar mandi diluar”.

Apakah homestay ini menyediakan peminjaman seperti sepeda motor atau fasilitas lainnya?

“ada. Siapa yg mau pakai, boleh. Tapi belom ada yang minjem gitu, tapi pernah bule itu minta sepeda motor dp Rp500.000 saya nggak mau ngasih, karna dia orang asing kan. Biasa kami nyewa perhari Rp50.000. itu wajib itu, ada aja yang merental sampai satu bulan, 2 minggu, 1 hari, gitu. Mobil pun ada kita, harganya

Rp250.000. Untuk makan pengunjung bisa membeli dari kami. Tapi itu uangnya beda sama uang kamar ya“

Efylin Homestay

1. Kira – kira sudah berapa lama *homestay* ini berdiri pak?

Homestay ini sudah berdiri selama 2 tahun.

2. Ada berapa jumlah kamar yang disediakan *homestay* ini?

Homestay ini memiliki 3 kamar. 1 kamar bisa menampung 5 orang.

3. Berapa harga per kamar di *homestay* ini pak ?

1 kamar Rp. 200.000 selama 24 jam

4. Fasilitas apa saja yang disediakan *homestay* ini pak ?

Fasilitas yang disediakan adalah kamar mandi, CCTV untuk memantau keamanan kendaraan pengunjung, mesin cuci, kompor dan gas.

5. Apa *homestay* ini menyediakan penyewaan kendaraan pak ?

Homestay ini tidak menyediakan penyewaan kendaraan, tetapi pengunjung dapat menggunakan 2 motor yang sudah disediakan di *homestay* ini. Jika pengunjung ingin memakai motor tersebut, tidak ada biaya tambahan yang dipungut. *Homestay* ini menyediakan penyewaan ATV bagi pengunjung yang ingin berekreasi di Pantai Lumban Bul-bul.

6. Berapa harga penyewaan ATV nya pak ?

Rp. 50.000 /jam. Kami menyediakan 3 ATV untuk disewakan.

7. Bagaimana proses pemesanan dan transaksi di *homestay* ini pak ?

Pemesanan dilakukan melalui nomor telepon yang disebar di *facebook*. Transaksi dilakukan secara langsung di tempat.

8. Bagaimana perkembangan *homestay* di daerah Pantai Lumban Bul-bul ini pak ?

Perkembangan *homestay* di daerah ini cukup baik semenjak Pantai Bul-bul terkenal. *Homestay* disini sangat ramai pengunjung bahkan sampai kurang jika sedang hari libur atau hari besar seperti Imlek, Natal, ataupun hari raya Idul Fitri. Jika hari biasa, tetap ada pengunjung misalnya rombongan gereja, jika sudah rombongan gereja atau sekolah yang berkunjung biasanya memesan 1 rumah langsung. Harga nya disesuaikan dengan

kesepakatan bersama dengan pengunjung. Jika 1 rumah, maka pengunjung akan disediakan tikar, selimut dan juga bantal.

Blessing Homestay

1. *Homestay Blessing* ini sudah berapa lama berdiri kak ?

Homestay ini sudah berdiri selama 3 tahun

2. Ada berapa jumlah kamar yang disediakan *homestay* ini kak ?

Homestay ini menyediakan 2 kamar

3. Berapa harga sewa per kamarnya kak ?

Rp. 150.000 /kamar. 1 kamar bisa menampung 3 hingga 4 orang

4. Fasilitas apa saja yang disediakan *homestay* ini kak ?

Fasilitas yang disediakan kamar mandi, pengunjung juga bisa memakai dapur. Jika

Pengunjung ingin disediakan makanan, maka akan disepakati secara langsung

menu apa yang diinginkan, untuk porsi berapa orang, dan harga yang dipatok juga

disepakati bersama dengan pengunjung.

5. Apa *homestay* ini menyediakan penyewaan kendaraan kak ?

Homestay ini tidak menyediakan penyewaan kendaraan karena biasanya pengunjung datang membawa kendaraan pribadi.

6. Bagaimana proses pemesanan dan transaksi pembayaran di *homestay* ini kak?

Pemesanan dilakukan melalui nomor telepon yang disebar di *facebook*. Transaksi

Pembayaran dapat dilakukan secara langsung di tempat, atau dengan mentranfer

Uang muka. Sisa pembayaran bisa dibayar ketika pengunjung sudah ingin pulang.

7. Bagaimana perkembangan *homestay* di daerah Pantai Lumban Bul-bul ini kak ?

Sejak Pantai Lumban Bul-bul ini terkenal, banyak pengunjung yang datang hingga

bisnis *homestay* di daerah ini pun ikut berkembang pesat. Jika sedang hari libur atau hari – hari besar, *homestay* sangat penuh hingga kekurangan kamar terkadang. Jika hari – hari biasa tetap ada pengunjung tetapi tidak seramai hari libur. Setiap bulannya pasti ada pengunjung. Terkadang ada pengunjung yang datang dengan rombongan dan menyewa 1 rumah. Jika sudah menyewa 1 rumah harga sewanya di sepakati bersama dengan pengunjung.

2. Lampiran Data Yang Diperoleh

Tabel 67 Data Homestay Tobasa

| No | Nama Homestay | Fasilitas | Jumlah Kamar | Alamat |
|----|-----------------------------------|-------------------------------------|--------------|-----------------------------------|
| 1 | <i>Homestay Jos N Mo</i> | Kamar Mandi Dalam | 3 | Jl. Sutomo |
| 2 | <i>Brussel Homestay</i> | Kamar Mandi Luar | 3 | Napitupulu Bagasan |
| 3 | <i>Pardest Homestay</i> | Kamar Mandi Dalam, TV , Kipas Angin | 3 | Jl. Sutomo |
| 4 | <i>Mawar Homestay</i> | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 5 | <i>Aster Homestay</i> | Kamar Mandi Luar | 1 | Pantai Bulbul |
| 6 | <i>Eflin Homestay</i> | Kamar Mandi Luar | 1 | Pantai Bulbul |
| 7 | <i>Blessing Homestay</i> | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 8 | <i>Melati Homestay</i> | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 9 | <i>Homestay Nael Jaya</i> | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 10 | <i>Ita Pagarbatu</i> | Kamar Mandi Dalam | 5 | Pagar Batu |
| 11 | <i>Uli Homestay</i> | Kamar Mandi Luar, TV | 3 | Soposurung |
| 12 | <i>Berkat Homestay</i> | Kamar Mandi Luar | 1 | Soposurung |
| 13 | <i>Green's Homestay</i> | Kamar Mandi Luar | 1 | Lumban Silintong |
| 14 | <i>Bahagia Homestay</i> | Kamar Mandi Luar | 1 | Soposurung |
| 15 | <i>Mery Homestay</i> | Kamar Mandi Luar | 2 | Jln. Raya-Tarutung |
| 16 | <i>Mikha Homestay</i> | Kamar Mandi Luar | 2 | Hauma Bange |
| 17 | <i>Homestay Ici</i> | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 18 | <i>3c Homestay</i> | Kamar Mandi Luar | 2 | Jl. Perluasan No. 10 |
| 19 | <i>Pagar Homestay</i> | Kamar Mandi Luar | 1 | Jl. Guru Somalaing, Lbn. Dolok |
| 20 | <i>Yiska Homestay</i> | Kamar Mandi Luar | 2 | Desa Sibolahotang |
| 21 | <i>Ony Homestay</i> | Kamar Mandi Dalam | 4 | Desa Sibolahotang |
| 22 | <i>Nauli Homestay</i> | Kamar Mandi Luar | 2 | Pantai Lumban Binanga |
| 23 | <i>Nauli Homestay</i> | Kamar Mandi Dalam | 3 | Jl. Raja Paindoan No.20A |
| 24 | <i>Leo Tampu Homestay</i> | Kamar Mandi Luar | 1 | Janji Maria, Sibulele |
| 25 | <i>Polado Homestay</i> | Kamar Mandi Dalam | 3 | Aruan |
| 26 | <i>Homestay Lambas</i> | Kamar Mandi Luar | 1 | Lumban Bulbul |
| 27 | <i>Jaya Ajibata Homestay</i> | Kamar Mandi Luar, TV | 3 | Desa Sigapiton |
| 28 | <i>Situmeang Ajibata Homestay</i> | Kamar Mandi Luar, TV | 1 | Desa Sigapiton |
| 29 | <i>Baragas Homestay</i> | Kamar Mandi Luar | 2 | Jln. Pematang Siantar, Tampubolon |
| 30 | <i>J&J Homestay</i> | Kamar Mandi Dalam | 2 | Jln. Napitupulu |
| 31 | <i>Toni Homestay Meat</i> | Kamar Mandi Luar | 2 | Desa Meat |
| 32 | <i>Rita Homestay Meat</i> | Kamar Mandi Luar | 2 | Desa Meat |
| 33 | <i>Guntur Homestay Meat</i> | Kamar Mandi Luar | 2 | Desa Meat |
| 34 | <i>Homestay Edelweis</i> | Kamar Mandi Luar | 2 | Desa Sigapiton |
| 35 | <i>Homestay Ecopod</i> | Kamar Mandi Dalam | 2 | Desa Sigapiton |
| 36 | <i>Homestay Jabu Nature</i> | Kamar Mandi Luar | 3 | Desa Sigapiton |

| | | | | |
|----|--|-------------------------------------|---|--------------------------------------|
| 37 | Bulbul Beach Homestay | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 38 | <i>Homestay Unedo</i> | Kamar Mandi Luar | 2 | Pantai Bulbul |
| 39 | Sopar Simanjuntak <i>Homestay</i> | Kamar Mandi Luar | 2 | Dusun I, Desa Meat |
| 40 | Janter Lodewyck Homestay | Kamar Mandi Luar | 2 | Dusun I, Desa Meat |
| 41 | Bagasta Homestay | Kamar Mandi Luar | 2 | Dusun III, Desa Meat |
| 42 | Togap Tampubolon <i>Homestay</i> | Kamar Mandi Luar | 2 | Dusun II, Desa Meat |
| 43 | R Jahara Tampubolon <i>Homestay</i> | Kamar Mandi Luar | 2 | Dusun II, Desa Meat |
| 44 | Karina Homestay | Kamar Mandi Luar | 2 | Kelurahan Napitupulu |
| 45 | Basado Homestay | Kamar Mandi Luar | 2 | Dusun III, Desa Meat |
| 46 | Felix Homestay | Kamar Mandi Dalam, TV , Kipas Angin | 2 | Desa Janji Maria |
| 47 | <i>Homestay Dion</i> | Kamar Mandi Luar | 2 | Lumban Bulbul |
| 48 | Sri Bunga Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 49 | Tupan Homestay | Kamar Mandi Luar | 2 | Dusun I Batu Mamak, Meranti Utara |
| 50 | Fatia Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 51 | Darma Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 52 | Carles Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 53 | Hendra Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 54 | Parunungan Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 55 | Johan Homestay | Kamar Mandi Luar | 3 | Dusun IV Meranti Utara |
| 56 | Dinar Homestay | Kamar Mandi Luar | 2 | Dusun IV Meranti Utara |
| 57 | Edi Homestay | Kamar Mandi Luar | 2 | Dusun I Batu Mamak, Meranti Utara |
| 58 | Opung Ella Homestay | Kamar Mandi Luar | 2 | Lumban Binanga, Desa Jangga Dolok |
| 59 | Opung Marhutala Homestay | Kamar Mandi Luar | 2 | Lapo Lintong, Desa Jangga Dolok |
| 60 | Opung Maranatha Homestay | Kamar Mandi Luar | 2 | Lumban Binanga, Desa Jangga Dolok |
| 61 | Managara Homestay | Kamar Mandi Luar | 2 | Gereja Buttu Raja, Desa Jangga Dolok |
| 62 | Nahason Homestay | Kamar Mandi Luar | 2 | Hutabagasan, Desa Jangga Dolok |
| 63 | Renika Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 64 | Mangantar Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 65 | Rumia Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 66 | Rumia Gultom Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 67 | Delpi Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 68 | Sahat Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 69 | Serbi Homestay | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |

| | | | | |
|----|--------------------------------|-----------------------|---|----------------------------------|
| 70 | Parulian <i>Homestay</i> | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 71 | Elferida <i>Homestay</i> | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 72 | Binsar <i>Homestay</i> | Kamar Mandi Luar | 2 | Sigapiton Kecamatan Ajibata |
| 73 | Hio Baja <i>Homestay</i> | Kamar Mandi Dalam, TV | 5 | Sigumpar |
| 74 | Taman Eden 100 <i>Homestay</i> | Kamar Mandi Dalam | 3 | Desa Sionggang, Kec. Lumban Julu |