**APLIKASI SISTEM INFORMASI PENCARIAN DAN RESERVASI KAFE DI MEDAN DENGAN API MAPS BERBASIS MOBILE DAN WEB**

**TUGAS AKHIR**

Oleh:

**HANDOKO**

NIM. 131112080

**RANDY MUHROJI**

NIM. 131111661



**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**MIKROSKIL**

**MEDAN**

**2017**

# Bab I Pendahuluan

## Latar Belakang

## Rumusan Masalah

## Ruang Lingkup

## Tujuan dan Manfaat

## Metodologi dan Pengembangan

# Bab II Tinjauan Pustaka

## lorem

## ipsum

# Bab III Analisis dan Perancangan

## Analisis

### Analisis Sistem Berjalan

Analisis perancangan sistem berjalan pada tugas akhir ini dilakukan terhadap Zomato(aplikasi pencarian restoran yang menyediakan informasi seputar restoran). Berikut adalah *usecase diagram* dari Zomato:



Gambar 3.1 Usecase diagram zomato

Berikut adalah narasi *usecase* untuk menjelaskan setiap *usecase* yang ada pada Gambar 3.1 *Usecase* diagram zomato:

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Login** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara user untuk masuk ke dalam sistem | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman login |  |
|  | 1. Menampilkan halaman regiser |
| 1. Mengisi data pada form register 2. Mengetuk tombol daftar |  |
|  | 1. Mengirim link verifikasi ke email user |
| 1. User melakukan verikasisi yang dikirimkan melalui email |  |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Register** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan pengunjung mendaftarkan diri menjadi pengguna/*user* dari sistem | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman login |  |
|  | 1. Menampilkan halaman login |
| 1. Mengisi data pada form login 2. Mengetuk tombol login |  |
| 1. User melakukan verikasisi yang dikirimkan melalui email | 1. Masuk ke halaman homepage dengan status login sebagai user |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Search** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara user untuk melakukan pencarian terhadap kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Pada halaman homepage terdapat form untuk pencarian kafe 2. User dapat mengetik keyword dari nama kafe |  |
|  | 1. Menampilkan hasil pencarian |
| 1. User dapat melakukan pengurutan seperti berdasarkan sory by popularity, rating, ataupun jarak terdekat. |  |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Add Review** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan bagaimana cara melakukan review terhadap suatu kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman kafe yang ingin di review |  |
|  | 1. Menampilkan halaman kafe |
| 1. Mengetuk tombol review |  |
|  | 1. Menampilkan form review |
| 1. Mengisi review, lalu mengetuk tombol publish review |  |
|  |  | 1. Menampilkan hasil review pada halaman kafe |

### Analisis Kebutuhan

#### Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional dilakukan dengan menggunakan *usecase diagram* untuk memperlihatkan interaksi antara *user* dan sistem. Berikut *usecase* dari Aplikasi Sistem Informasi Pencarian Dan Reservasi Kafe Di Medan Berbasis *mobile* dan *web:*



Berikut adalah narasi usecase untuk menjelaskan setiap usecase yang terdapat pada usecase diagram:

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Register** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara pengunjung untuk register sebagai user/pengguna | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman login |  |
|  | 1. Menampilkan halaman regiser |
| 1. Mengisi data pada form register 2. Mengetuk tombol daftar |  |
|  | 1. Mengirim link verifikasi ke email user |
| 1. User melakukan verikasisi yang dikirimkan melalui email |  |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Login** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara user untuk masuk ke dalam sistem | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman login |  |
|  | 1. Menampilkan halaman login |
| 1. Mengisi data pada form login 2. Mengetuk tombol login |  |
|  | 1. Menampilkan user ke halaman homepage dengan status telah login |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Information** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara user untuk menemukan informasi secara mendetail dari kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman homepage |  |
|  | 1. Menampilkan halaman homepage |
| 1. Mengetuk kafe yang ingin dilihat informasinya |  |
|  | 1. Menampilkan halaman informasi kafe |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Search** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara user untuk melakukan pencarian terhadap kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman homepage |  |
|  | 1. Menampilkan halaman homepage |
| 1. Memasukkan keyword dari kafe yang ingin dicari |  |
|  | 1. Menampilkan hasil pencarian |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Bookmark** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara pengunjung untuk melihat kafe yang telah di bookmark | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman profile |  |
|  | 1. Menampilkan halaman profile |
| 1. Mengetuk tombol bookmark |  |
|  | 1. Menampilkan daftar kafe yang telah di bookmark |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Edit Profile** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara pengunjung untuk mengedit user profile | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman profile |  |
|  | 1. Menampilkan halaman profile |
| 1. Mengetuk tombol account 2. Lalu pilih edit profile |  |
|  | 1. Menampilkan form edit profile |
| 1. Melakukan edit profile, pada form yang tersedia 2. Mengetuk tombol simpan |  |
|  |  | 1. Menampilkan pesan profile berhasil di edit |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Make Review** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara pengunjung untuk register sebagai user/pengguna | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman informasi kafe |  |
|  | 1. Menampilkan halaman informasi kafe |
| 1. Mengetuk tombol review 2. Mengetuk tombol add review |  |
|  | 1. Menampil form add review |
| 1. Isi review dan berikan rating terhadap kafe tersebut |  |
|  |  | 1. Menampilkan pesan sukses, lalu menampilkan hasil review |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Sent Message** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara melakukan sent message kepada administrator | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman profile |  |
|  | 1. Menampilkan halaman profile |
| 1. Mengetuk tombol sent message |  |
|  | 1. Menampilkan form untuk sent message |
| 1. Mengisi form, dengan cara memasukkan pesan |  |
|  |  | 1. Menampilkan pesan sukses, pesan terkirim |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Reservasi** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan melakukan reservasi kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman informasi kafe |  |
|  | 1. Menampilkan halaman informasi kafe |
| 1. Ketuk tombol reservasi |  |
|  | 1. Menampilkan halaman reservasi |
| 1. Isi form reservasi seperti jumlah orang, menu makanan, dan waktu kapan reservasi 2. Mengetuk tombol simpan |  |
|  |  | 1. Menampilkan form pembayaran |
|  | 1. Mengisi form pembayaran 2. Melakukan pembayaran dengan cara transfer ke rekening yang dipilih sebelumnya 3. Mengetuk tombol Confirm Payment, lalu memfoto bukti pembayaran |  |
|  |  | 1. Menampilkan pesan sukses reservasi |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Refund Reservation** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara melakukan refund reservation terhadap reservasi yang telah dilakukan | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman informasi kafe |  |
|  | 1. Menampilkan halaman informasi kafe |
| 1. Mengetuk tombol sent reservation |  |
|  | 1. Menampilkan daftar kafe yang telah di reservasi |
| 1. Mengetuk kafe yang ingin di refund 2. Ketuk refund reservation dengan menuliskan alasan pembatalan |  |
|  |  | 1. Melakukan proses verifikasi 2. Menampilkan pesan suksen refund reservation, lalu sistem akan mengembalikan dana yang telah dikirim user |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Delievery** | |
| Aktor | ***User*** | |
| Deskripsi | Usecase ini menjelaskan cara melakukan delievery | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman informasi kafe |  |
|  | 1. Menampilkan halaman informasi kafe |
| 1. Mengetuk tombol Delievery |  |
|  | 1. Menampilkan form delievery |
| 1. User mengisi form 2. Mengetuk tombol submit |  |
|  |  | 1. Menampilkan pesan “Menunggu paling lama 10 menit” 2. Mengeluarkan hasil status delievery, dan total biaya delievery |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Edit info kafe** | |
| Aktor | ***Café owner*** | |
| Deskripsi | Usecase ini menjelaskan cara melakukan Edit informasi kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman homepage café owner |  |
|  | 1. Menampilkan halaman homepage café owner |
| 1. Mengetuk tombol edit café |  |
|  | 1. Menampilkan form untuk edit cafe |
| 1. Mengisi form 2. Mengetuk tombol submit |  |
|  |  | 1. Menampilkan pesan berhasil |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Sent Message** | |
| Aktor | ***Café owner*** | |
| Deskripsi | Usecase ini menjelaskan cara melakukan Sent Message kepada administrator | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman homepage café owner |  |
|  | 1. Menampilkan halaman homepage café owner |
| 1. Mengetuk tombol sent message |  |
|  | 1. Menampilkan form untuk sent message |
| 1. Mengisi form 2. Mengetuk tombol submit |  |
|  |  | 1. Menampilkan status “pesan berhasil terkirim” |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Status Reservasi** | |
| Aktor | ***Café owner*** | |
| Deskripsi | Usecase ini menjelaskan cara menampilkan status reservasi | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman homepage café owner |  |
|  | 1. Menampilkan halaman homepage café owner |
| 1. Mengetuk tombol status reservasi |  |
|  | 1. Menampilkan halaman status reservasi |

|  |  |  |
| --- | --- | --- |
| Nama usecase | **Edit info kafe** | |
| Aktor | ***Café owner*** | |
| Deskripsi | Usecase ini menjelaskan cara melakukan Edit informasi kafe | |
|  | Aksi Aktor | Respon Sistem |
| *Normal Flow Event* | 1. Masuk ke halaman homepage café owner |  |
|  | 1. Menampilkan halaman homepage café owner |
| 1. Mengetuk tombol edit café |  |
|  | 1. Menampilkan form untuk edit cafe |
| 1. Mengisi form 2. Mengetuk tombol submit |  |
|  |  | 1. Menampilkan pesan berhasil |

#### Analisis Kebutuhan Non Fungsional

Berikut tabel PIECES dari Aplikasi Sistem Informasi Pencarian Dan Reservasi Kafe:

|  |  |
| --- | --- |
| **Parameter** | **Penjelasan** |
| *Performance* | 1. Mempercepat proses pencarian kafe berdasarkan jarak terdekat. |
| *Information* | 1. Mempermudah *user* untuk mengetahu informasi secara detail terhadap kafe. 2. Mempermudah pemilik kafe untuk mempromosikan kafenya dengan cara menawarkan berbagai promo menarik*.* |
| *Economic* | 1. Mempermudah proses pembayaran *user* kepada pemiliki kafe atas reservasi yang dilakukan user*.* |
| *Control* | 1. Mempermudah penyimpanan history reservasi yang dilakukan oleh user. 2. Memudahkan *administrator* melaksanakan *user control* dan *review control.* |
| *Efficiency* | 1. Mempermudah *user* melakukan proses reservasi 2. Mempermudah *payment confirmation.* |
| *Service* | 1. Sistem menerapkan GUI yang *user friendly.* |

## Perancangan

### Perancangan Sistem Usulan

Perancangan sistem usulan dari sistem ini dimodelkan dengan *data flow diagram* versi Gane dan Sarson. *Data flow diagram* dipilih dengan alasan dapat memperlihatkan dengan jelas aliran data yang terjadi pada sistem sehingga memudahkan dalam proses penyesuaian desain dan API dari sistem.

### Perancangan Tampilan

#### Web Platform

##### *User Side*

###### Homepage

###### Login dan Register

###### Detail Kafe

###### User Profile

###### Bookmarks

###### Reservasi

###### Account

##### *Owner* Kafe *Side*

###### Profil Kafe

###### Daftar Menu

###### Galeri

###### Reservasi

###### Account

##### Admin Side

###### Home

###### Sponsor

###### Daftar Kafe

#### Mobile Platform

##### *Homepage* dan Menu (*guest*)

*Homepage* merupakan halaman yang pertama kali muncul pada saat aplikasi dijalankan, *guest* merupakan sebutan untuk user yang menggunakan aplikasi namun belum *login*. Pada aplikasi ini *guest* hanya bisa menggunakn fitur pencarian dan melihat informasi tentang kafe.

##### *Login* dan *Register*

##### Menu(user) dan *View* *Profile*

##### *Searchpage*

##### Detail Kafe

##### *Booking* dan *Payment*

##### *Feedpage*

##### Bookmarks

##### Reserv

##### Account

##### *Edit Profile* dan *Review Drafts*

### Perancangan Basis Data

Perancangan basis data sangat penting dalam pengembangan sebuah sistem karena basis data merupakan salah satu bagian terpenting yang sangat menentukan tampilan dan penyesuaian API untuk *mobile apps.* Berikut adalah basis data yang telah melalui proses normalisasi tahap 3 dan akan diimplementasikan dalam sistem ini:

# Daftar Pustaka