**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

Kafe adalah tempat untuk makan dan minum sajian cepat saji dan menyuguhkan suasana santai atau tidak resmi, selain itu juga merupakan suatu tipe dari restoran yang biasanya menyediakan tempat duduk didalam dan diluar restoran. Kebanyakan kafe tidak menyajikan makanan berat namun lebih berfokus pada menu makanan ringan seperti kue, roti, sup, dan minuman (Marsum, 2005). Kafe biasanya dijadikan sebagai tempat pertemuan atau sekedar tempat kumpul untuk berdiskusi yang didominasi oleh kalangan remaja maupun mahasiswa. Namun maraknya kafe yang muncul membuat pengunjung kafe juga ikut meningkat, sebagaimana pada saat bulan ramadhan kafe-kafe banyak digunakan oleh masyarakat kota medan untuk melakukan kegiatan buka puasa bersama. Salah satu kesulitan untuk menentukan kafe dalam kegiatan buka puasa bersama adalah pada saat penentuan dan pencarian kafe yang ingin dikunjungi. Pada saat melakukan reservasi tempat pihak kafe tersebut mengharuskan setiap pelanggan untuk menentukan jumlah tempat yang tersedia beserta sejumlah menu yang dipesan. Pelanggan diharuskan datang langsung atau telepon untuk mengetahui menu yang tersedia pada kafe, maupun kapasitas meja yang tersedia pada kafe tersebut hal ini akan sangat sulit dilakukan jika hanya melalui telepon. Meski saat ini telah hadir aplikasi Zomato (aplikasi mencari resto dan tempat makan di Jakarta) aplikasi ini tidak memungkinkan pelanggan melakukan pemesanan secara online dan terfokus pada informasi kafe serta testimonial dari para pelanggan dan hanya terbatas di Jakarta.

Dari masalah tersebut dibutuhkan sebuah sistem yang dapat digunakan untuk melakukan pencarian kafe dan reservasi secara online, selain itu user juga dapat memesan makanan dan minuman sekaligus dan dapat melakukan pencarian kafe yang difilter berdasarkan nama dan lokasi kafe. Sistem sekaligus dapat memberikan kemudahan bagi pelanggan untuk melihat detail kafe maupun informasi detail menu berupa harga maupun kategori makanan (halal, non halal, *vegetarian*, dsb). Pada API google maps memungkinkan user menemukan kafe terdekat menggunakan *routing get direction* yang tersedia pade GPS smartphone pengguna. Sistem pembayaran pada pemesanan kafe menggunakan sistem rekening bersama yang dilakukan via transfer antar rekening bank.

Berdasarkan uraian di atas, dibangunlah aplikasi yang dapat memenuhi kebutuhan pencarian dan reservasi kafe . Oleh sebab itu, Tugas Akhir ini diberi judul “**Aplikasi Sistem Informasi Pencarian Dan Reservasi Kafe Dengan API Maps Berbasis Mobile Dan Web**”.

## Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dijabarkan sebagai berikut :

1. Sulitnya mencari dan menentukan kafe yang sesuai untuk melakukan pertemuan, karena aplikasi saat ini tidak memberikan informasi jumlah tempat yang tersedia.
2. Aplikasi sistem informasi kafe saat ini hanya sebatas informasi menu makanan dan fasilitas yang tersedia dari kafe tersebut, dan tidak terdapat mekanisme reservasi yang memudahkan user untuk langsung melakukan pemesanan tempat.

## Ruang Lingkup

Tujuan tugas akhir ini adalah mengembangkan sebuah aplikasi yang memungkinkan pelanggan melakukan pencarian lokasi kafe terdekat dengan menggunakan API google maps serta pemesanan kafe secara online, aplikasi mengetahui informasi detail kafe dan informasi detail menu yang terdapat pada kafe

Manfaat tugas akhir ini adalah sebagai berikut:

1. Menyediakan wadah kepada pemilik-pemilik kafe mempromosikan kafenya.
2. Mempermudah pencarian lokasi kafe terdekat dengan menggunakan *routing get direction* yang tersedia pada GPS *smartphone* User
3. Menyediakan fitur Reservasi kafe secara online, sehinggan pelanggan tidak perlu datang ke kafe untuk melakukan reservasi dan mengetahui berapa meja yang tersedia pada kafe.

Mempermudah pelanggan mendapatkan informasi detail kafe serta promo-promo potongan harga sebelum berkunjung ke kafe.

## Tujuan dan Manfaat

Batasan masalah pada tugas akhir ini adalah:

* + - 1. Fitur-fitur yang terbagi sesuai pada kategori pengguna:

1. Administrator
2. Mengelola data pengguna dan data kafe;
3. Menverifikasi data kafe;
4. Mengelola pembayaran reservasi kafe*;*
5. Administrator Kafe
6. Membuat akun untuk dapat mengakses fitur administrator kafe pada sistem, fitur ini tersedia pada *platform web;*
7. Mengelola daftar menu pada Kafe, fitur ini tersedia pada *platform web;*
8. Memperbaharui informasi meja yang tersedia pada kafe;
9. Mengelola promo pada menu, fitur ini tersedia pada *platform web;*
10. Pengguna
11. Membuat akun untuk dapat mengakses semua pengguna;
12. Melakukan reservasi tempat pada kafe, fitur ini tersedia pada platform web dan mobile;
13. Mengelola profil, komentar serta memberikan rating pada kafe, fitur ini tersedia pada platform web dan mobile;
14. Melakukan pencarian kafe terdekat menggukana *routing get direction* yang tersedia pade GPS smartphone pengguna, fitur ini hanya tersedia pada platform mobile;
15. Guest
16. Melihat informasi detail kafe dan menu;
17. Melakukan pencarian kafe terdekat menggunakan routing get direction yang tersedia pade GPS smartphone pengguna, fitur ini hanya tersedia pada platform;
    * + 1. *Mobile Application* sebagai produk dari tugas akhir ini hanya dapat di-*install /* pasang pada *smartphone*  yang menggunakan sistem operasi Android dan memiliki keterbatasan fitur dari sisi *administrator panel* yang hanya dapat diakses di *web application.*
        2. *Web application*  sebagai produk dari tugas akhir ini memiliki keterbatasan fitur dari sisi pencarian lokasi kafe terdekat.
        3. Sistem pembayaran pada pemesanan kafe menggunakan sistem rekening bersama yang pengimplementasiannya bersifat simulasi, simulasi yang dimaksud adalah pembayaran dilakukan via transfer antar rekening bank, dengan pengelolaan rekening pihak ketiga dilakukan secara manual oleh administrator sistem.

## Metodologi dan Pengembangan

Untuk menyelesaikan tugas akhir ini, diimplementasikan *waterfall* sebagai metodologi pengembangan sistem. Berikut tahapan proses yang harus dilaksanakan:

1. *Requirement*

Tahapan ini bertujuan mengumpulkan data yang diperlukan dalam proses bisnis. Beberapa proses yang terjadi dalam tahapan ini:

1. Menentukan pengguna yang terlibat dalam aplikasi beserta fungsi kontrol masing-masing pengguna.
2. Melakukan kajian terhadap proses bisnis yang terjadi dalam proses pemesanan menu yang terdapat pada kafe.
3. *Analysis*

Analisis terhadap sistem yang sedang dibangun dilakukan dengan manfaat UML *(Unified Model Languange).* Beberapa jenis UML yang digunakan dalam menganalisa sistem:

* 1. *Usecase* digunakan untuk memperlihatkan interaksi yang terjadi antara *user* dan sistem.
  2. *Data Flow Diagram* digunakan untuk memperlihatkan aliran data yang terjadi antara *user,* sistem dan *datastore.*

1. *Design*

Desain yang dihasilkan dari tahapan ini berupa *mockup* atau rancangan tampilan dari sistem yang sedang dibangun.

1. *Implementation*

Tahapan ini dibagi dalam 2 (dua) proses utama:

1. C*oding*

Mengimplementasikan analisis dan desain yang telah dirancang pada tahapan sebelumnya untuk membangun sistem.

1. *Testing*
   1. Bertujuan untuk mengevaluasi sistem yang telah dibangun.

# 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. 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.* |