**PROPOSAL TUGAS AKHIR**

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

Oleh

**HANDOKO**

NIM. 131112080

**RANDY MUHROJI**

NIM. 131111661

Disetujui Oleh:

Dosen Pembimbing I, Dosen Pembimbing II,

Sunario Megawan, S.Kom., M.Kom. Felix Pandi, S.Kom., M.Kom.

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**MIKROSKIL**

**MEDAN**

**2016**

# Latar Belakang

Kafe adalah tempat untuk makan dan minum sajian cepat saji dan menyuguhkan suasanan 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. Aplikasi yang telah ada saat ini belum memiliki fitur untuk mengetahui berapa jumlah meja yang tersedia pada kafe tersebut, sebagaimana user masih harus direpotkan untuk menanyakan perihal informasi terkait jumlah meja yang tersedia maupun fitur untuk mengetahui informasi detail kafe, informasi detail menu berupa harga maupun kategori makanan (halal, non halal, *vegetarian*, dsb). Aplikasi yang telah ada saat ini belum memenuhi kebutuhan pengguna seperti pencarian lokasi kafe terdekat dan fitur reservasi online secara bersamaan. 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 resto dan tempat makan serta 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. Sistem juga dapat melakukan pencarian kafe yang difilter berdasarkan nama dan lokasi kafe. Sistem sekaligus dapat memberikan kemudahan bagi pemilik kafe untuk mempromosikan kafenya, dan dalam hal ini pemilik kafe dapat melakukan potongan harga tertentu terhadap menunya agar dapat menarik perhatian konsumen. Pada API google maps memungkinkan user menemukan kafe terdekat menggunakan *routing get direction* yang tersedia pade GPS smartphone pengguna.

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. Perlunya aplikasi pencarian kafe terdekat dan reservasi kafe secara online, yang memungkinkan user tidak perlu lagi datang ke kafe untuk melakukan reservasi.
2. Perlunya sistem informasi pada kafe yang memungkinkan user dapat melihat detail kafe berupa tempat, lokasi, detail menu berupa harga maupun kategori makanan (halal, non halal, vegetarian, dsb).
3. Perlunya sistem pembayaran yang aman untuk di implementasikan pada sistem.

# Tujuan dan Manfaat

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

# Batasan Masalah

Batasan masalah pada tugas akhir ini adalah :

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

1. Adminitrator

Mengelola data pengguna dan data kafe;

Menverifikasi data cafe;

Mengelola pembayaran reservasi kafe*;*

1. Administrator Kafe

Membuat akun untuk dapat mengakses fitur administrator kafe pada sistem, fitur ini tersedia pada *platform web;*

1. Mengelola daftar menu pada Kafe, fitur ini tersedia pada *platform web;*
2. Mengelola promo pada menu, fitur ini tersedia pada *platform web;*
3. Pengguna
4. Membuat akun untuk dapat mengakses semua pengguna;
5. Melakukan reservasi tempat pada kafe, fitur ini tersedia pada platform web dan mobile;
6. Mengelola profil, komentar serta memberikan rating pada kafe, fitur ini tersedia pada platform web dan mobile;
7. Melakukan pencarian kafe terdekat menggukana *routing get direction* yang tersedia pade GPS smartphone pengguna, fitur ini hanya tersedia pada platform mobile;
8. Guest
9. I .Melihat informasi detail kafe dan menu;
10. Melakukan pencarian kafe terdekat menggukana routing get direction yang tersedia pade GPS smartphone pengguna, fitur ini hanya tersedia pada platform;
    * + 1. *Mobile Aplication* sebagai produk dari tugas akhir ini hanya dapat di-*install /* pasang pada *smartphone*  yang menggunakan system operasi Andoroid 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 sister.

# Metodologi Pengembangan Sistem

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. Mengumpulkan data tentang *stakeholder* yang terlibat sistem yang sedang dibangun.
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* digunakana 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*.

Bertujuan untuk mengevaluasi sistem yang telah dibangun.

# Tinjauan Pustaka

* + - 1. Cafe

Pengertian Kafe Pada dasarnya kafe itu sendiri mempunyai arti harafiah yaitu Restoran kecil yang melayani atau menjual makanan ringan dan minuman, kafe biasanya digunakan orang untuk rileks (Sumber: Dictionary of English Language and Culture, Longman) Restoran murah yang menyediakan makanan yang mudah dimasak/dihidangkan kembali (Sumber: The New Dictionary and Theosaurus) Jam buka kafe secara umum dimulai dari pagi sampai malam (10.00-22.00) atau ada pula kafe yang dibuka mulai dari sore hari (18.00-02.00). Tempat yang biasanya dipakai untuk membuka kafe adalah Rumah yang didekorasi dan ditata dengan baik, dan terkadang *shopping* *mall* atau sebuah pusat perbelanjaan. Tempat seperti ini biasanya lebih ramai dibanding dengan restoran karena banyak tamu yang keluar masuk.

Sistem Penyajian Kafe Cara penyajian makanan dalam kafe terdapat beberapa cara, yaitu :

1. Self Service

Dimana pengunjung melakukan pelayanan bagi dirinya sendiri. Pengunjung datang kemudian mengambil makanan dan minuman yang mereka inginkan kemudian menuju ke kasir dan membayar makanan mereka lalu duduk di tempat yang telah disediakan. Cara ini terkesan familiar dan bersahabat.

1. Waiter of Waitress Service to Table

Pengunjung datang lalu duduk pada kursi yang telah disediakan, kemudian pramusaji akan melayani mereka, mengantar menu dan makanan hingga membayar ke kasir, sehingga orang tidak perlu beranjak dari kursinya. Cara ini terkesan formal.

1. Counter Service

Dimana terdapat area khusus yang terdapat display makanan yang ada, biasanya digunakan untuk pelayanan yang cepat dan service tidak formal.

1. Automatic Vending

Menggunakan mesin otomatis. Pengunjung memasukkan koin lalu dari mesin keluar makanan yang dipilihnya.

Menurut Building Planning & Desain Standart secara umum kafe merupakan tempat yang menyediakan makanan dan minuman yang mendekati restaurant dalam sistem pelayanan yang di dalamnya terdapat hiburan alunan musik, sehingga kafe dapat digunakan sebagai tempat yang santai dan untuk berbincang-bincang.

Dan yang membedakan kafe dengan tempat makan yang lain adalah kafe yang lebih mengutamakan hiburan dan kenyamanan pengunjung. Hiburan tersebut dapat membuat pengunjung kafe tidak cepat bosan.

Makanan dan minuman yang terdapat dalam kafe antara lain:

- *Hot Drink* seperti kopi

- *Cold Drink* seperti *ice coffe, ice tea, soft drink.*

- *Hot Food* seperti *sandwich*, *burger*

- *Cold Food* seperti *salad*, *baker*, dan *snack*

* + - 1. Android Mobile Platform

Android merupakan sebuah sistem operasi pada perangkat *mobile* seperti *smartphone* dan *PC Tablet* dan merupakan salah satu versi modifikasi dari Linux *Operating System.* Pada awalnya, Android dikembangkan oleh perusahaan *startup* yang bernama Android Inc. pada tahun 2005. Namun seiring perkembangan Android yang sangat pesat dan mampu bersaing dengan *mobile operating system* lainnya*,* Google Inc. mengakuisisi Android dan menjadikan Android *opensource operating system.*  Pengembang dapat membuat aplikasi untuk flatform android dengan menggunakan android SDK dengan menggunakan bahasa pemrograman java dan berjalan dalam Dalvik dengan misin virtual yang telah tertanam dan dan berjalan di atas kernel linux. Hal ini menguntungkan bagi *vendor* (penyedia *hardware*) yang diberikan hak untuk menambahkan ekstensi yang mereka inginkan di dalam Android, seperti Oppo yang telah melakukan modifikasi pada Android yang digunakan pada *smartphone* mereka dan menamakannya ColorOS. Android juga merilis sebagian besar *code* di bawah Apache License Open-Source sehingga *mobile developer* dapat mengunduh *source code* Android secara keseluruhan. (Lee, W., M., 2011).

* 1. Fitur-fitur Android

Beberapa fitur yang tersedia pada Android mobile platform (Lee, W., M., 2011) :

Mendukung relational database.

Koneksi. Secara umum Android telah mendukung GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, WiFi, LTE dan WiMAX. Namun penggunaan fitur koneksi ditentukan oleh vendor (penyedia hardware) yang menggunakan Android platform pada mobile device yang mereka pasarkan.

Pengolah pesan – MMS dan SMS.

Internet browser – Google Chrome.

Media support. Android telah mendukung H.263, H.264 (dalam kategori 3GP dan MP4), MPEG-4 SP, AMR, AMR-WB (dalam kategori 3GP), AAC, HE-AAC (dalam kategori 3GP dan MP4), MP3, MIDI, Oog Vorbis, WAV, JPEG, PNG, GIF dan BMP.

Hardware support. Beberapa hardware yang dapat dikombinasikan dengan software yang dipasang pada Android adalah sensor Accelerometer, kamera, kompas digital, sensor jarak dan GPS (Global Positioning System).

Multi touch dan multi tasking.

Flashing.

Tethering (pembagian jaringan internet sebagai hotspot nirkabel maupun dengan memanfaatkan kabel).

2.2. Arsitektur Android

Arsitektur perangkat lunak adalah struktur-struktur yang menjadi landasan untuk menentukan keberadaan komponen-komponen perangkat lunak, cara komponen perangkat lunak berinteraksi dengan perangkat keras dan organisasi antar perangkat. Arsitektur perangkat lunak diklasifikasikan mejadi 5 (lima) bentuk (Hafid, A., M., 2015):

*Monolhitic system*.

Merupakan stuktur sistem operasi sederhana yang dilengkapi dengan operasi “dual” pelayanan *(system call)* yang diberikan oleh sistem operasi. Modul *system call* dikerjakan dengan cara mengambil sejumlah parameter pada tempat yang telah ditentukan sebelumnya, seperti *register* atau *stack* dan kemudian mengeksekusi setiap instruksi *trap* pada *monitor mode*.

Gambar 1. Monolitic System

Seperti yang digambarkan pada Gambar 1., setiap layanan dieksekusi pada ruang memori yang sama sehingga mempercepat proses eksekusi. Di saat yang bersamaan penggunaan satu alamat memori membuat rumitnya penanganan masalah karena sulit untuk dipisahkan dan dilokasikan. Kelemahan lain dari arsitektur adalah:

a. Sulit menyediakan fasilitas pengamanan;

b. Kurang efisien dalam penggunaan memori, karena setiap perangkat harus menjalankan seluruh kernel yang ada, sementara tidak semua kernel diperlukan untuk menjalan suatu perangkat’;

c. Tidak berfungsinya sebuah kernel menyebabkan tidak berfungsinya sebuah perangkat secara total.

*Layered System*

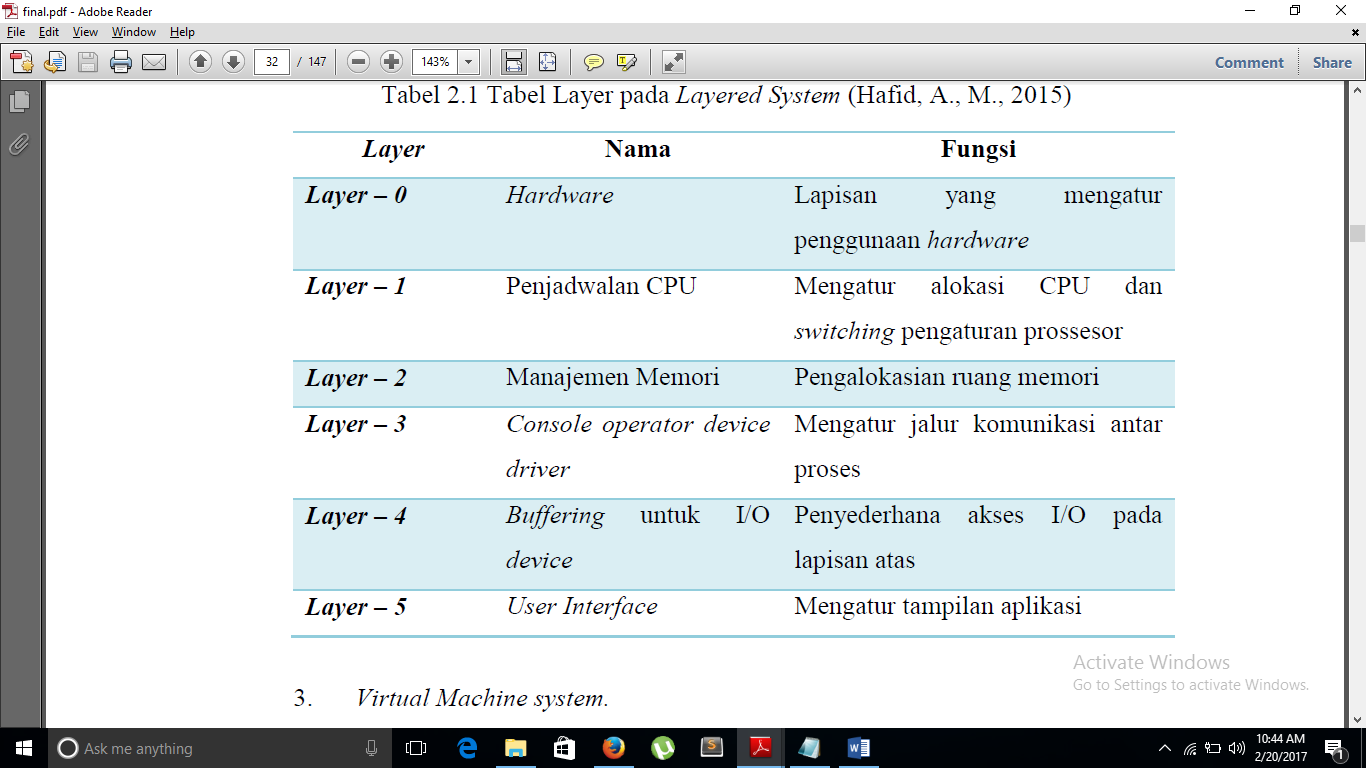
Pendekatan yang digunakan pada sistem ini adalah modularisasi, yaitu membagi sistem operasi menjadi beberapa lapis (tingkat). Lapisan terendah (layer 0) adalah perangkat keras dan lapisan teratas (layer N) adalah user interface. Beberapa sistem operasi yang telah menggunakan pendekatan ini adalah THE, UNIX termodifikasi, Venus dan OS/2.

Beberapa kelebihan layered system:

a. Semua keuntungan dari pendekatan modularisasi.

b. Setiap modul dirancang, dibangun dan diuji secara independen. Sehingga memudahkan developer untuk melakukan penanganan kesalahan ketika terjadi kesalahan pada salah satu modul.

Sementara kelemahan layered system adalah adanya kemungkinan seluruh lapisan harus diprogram ulang apabila terjadi logic error pada salah satu fungsi utama pada sistem operasi. Hal ini dikarenakan layered system mengharuskan fungsi utama ditanam pada setiap lapisan. Berikut adalah tabel 2.1. yang memperlihatkan daftar lapisan pada layered system:



Gambar 2. Layered System

*Virtual Machine System*

Pendekatan yang digunakan oleh arsitektur ini hampir sama dengan layered system, dengan tambahan berupa user interface yang menghubungkan hardware dengan kernel untuk tiap-tiap proses. Virtual machine menyediakan user interface yang identik untuk hardware yang ada. Arsitektur ini membuat ilusi atau virtual untuk beberapa proses dan mengeksekusi processor dan memori (virtual) masing-masing.

Walau konsep ini cukup baik, namun sangat kompleks untuk dapat diimplementasikan karena system berjalan menggunakan metode dual mode, sementara virtual machine hanya dapat berjalan pada monitor mode atau user mode. Dampaknya adalah virtual machine harus dijalankan melalui physical user mode yang menyebabkan adanya transfer dari user mode ke monitor mode pada virtual machine. Namun arsitektur ini menghadirkan banyak fitur yang memanjakan penggunanya, misalnya penjadwalan CPU dapat membuat tampilan seolah-olah pengguna mempunyai processor sendiri atau spooling dan system file dapat menyediakan card reader virtual dan line printer virtual. Beberapa sistem operasi yang pernah menggunakan arsitektur ini adalah IBM S/370 dan IBM VM/370.



Gambar 3 Perbandingan sitem operasi dengan dan tanpa virtual machine

Pendekatan ini berkembang seiring perkembangan teknologi menjadi emulator operating system sehingga sistem operasi dapat menjalankan aplikasi-aplikasi untuk sistem operasi lain. Misalnya MS-Windows NT dapat menjalankan aplikasi untuk MS-DOS atau OS/3 text mode dan aplikasi Win16 dengan memanfaatkan Win32 API (system call di MS-Windows NT) untuk melakukan proses eksekusi Seperri yang digambarkan pada Gambar 2.2., salah satu kelebihan dari pendekatan ini adalah setiap proses bekerja secara independen karena dijalankan oleh virtual machine yang berbeda, sehingga kerusakan/kegagalan eksekusi pada salah satu proses tidak berdampak pada seluruh sistem yang ada. Kelebihan dan kekurangan lain dari pendekatan ini secara bersamaan adalah:

a. Virtual machine menyediakan perlindungan yang lengkap bagi sumber daya sistem sehingga masing-masing virtual machine dari virtual machine yang lain. Isolasi ini berdampak pada tidak terjadinya pembagian sumber daya secara langsung.

b. Arsitektur ini sangat cocok jika ingin melakukan riset dan pengembangan operating system. Pengembangan dikerjakan pada virtual machine sehingga tidak mengganggu main operating system.

c. Konsep ini sangat sulit untuk mengimplementasikan kebutuhan dan duplikasi yang tepat pada mesin yang sebenarnya.

*Client-Server System*

Arsitektur client-server membagi setiap proses yang dikerjakan dalam beberapa modul server process yang akan mengeksekusi setiap permintaan dari user/client server untuk meminimalisir jumlah kernel yang mengeksekusi setiap proses yang ada. Karena arsitektur ini bekerja pada user mode process dan bukan monitor mode, maka server tidak dapat mengakses hardware secara langsung, yang berarti apabila terjadi kerusahakan pada server file, maka kerusakan tidak menyebabkan gangguan bagian lain dari sistem, misalnya server terminal.



Gambar 4. Client-Server System

Seperti yang terlihat pada Gambar 3., kelemahan pendekatan ini adalah lambatnya pertukaran informasi yang terjadi antara user/client dan server process menyebabkan lambatnya eksekusi masing-masing perintah. Sementara kelebihan dari arsitektur ini adalah:

a. Dapat diadaptasikan pada sistem terdistribusi;

b. Server tidak perlu tahu sumber (local atau by network) pesan yang dikirimkan oleh client ketika sedang menjalin komunikasi dengan server;

c. Pengembangan dilakukan secara modular;

d. Kesalahan pada suatu subsistem tidak akan menggangu subsistem lain ataupun sistem secara utuh

Object Oriented System

Setiap layanan akan menjadi object dengan proses, direktori, berkas sebagai properti dari objek tersebut. Beberapa sistem operasi yang menggunakan arsitektur ini adalah Eden, Choices, X-Kernel, Medusa, Clunds, Amoeba, Muse dan Sistem Operasi MS-Windows NT meski tidak secara keseluruhan.

Sebagai sebuah platform, Android juga dibangun di atas platform yang dapat membatu penanganan setiap layanan. Berikut adalah gambar arsitektur Android:



Gambar 5. Arsitektur Android mobile platform

Berikut adalah penjelasan dari setiap lapisan pada arsitektur Android seperti Gambar 2.4. (Lee, W., M., 2011):

a. Linux Kernel.

Lapisan ini merupakan kernel dasar dari Android yang berisi semua driver tingkat rendah untuk berbagai komponen hardware yang dipasangkan pada perangkat keras seperti smartphone. Kernel ini akan menjadi penerjemah bagi hardware untuk dapat melaksanakan fungsinya secara baik.

b. Libraries.

Mengandung seluruh code yang menyediakan fitur utama dari sistem operasi Android. Sebagai contoh, library SQLite yang menyediakan dukungan database yang dapat digunakan pula oleh aplikasi lain yang dipasang oleh pengguna di sistem operasi Android.

c. Android runtime.

Lapisan ini juga menyediakan kode yang disimpan sebagai library, namun memiliki fungsi dan tujuan yang lebih spesifik, yaitu library untuk mengembangkan aplikasi dengan menggunakan bahasa pemrograman java.

d. Application Framework.

Menjadi lapisan yang dapat digunakan oleh mobile apps developer ketika mengembangkan aplikasi yang dapat dipasang pada mobile device dengan android platform.

e. Applications.

Pada lapisan teratas ini, ditanamkan mobile applications yang sudah dipasangkan secara otomatis oleh pihak Android ataupun mobile applications yang dipasangkan oleh pengguna melalui Android market.

* + - 1. Website

Website atau situs web adalah sekumpulan halaman-halaman web yang saling berhubungan, termasuk didalamnya *homepage*, dan umumnya berada pada server yang sama, dan disiapkan dan dipelihara sebagai kumpulan informasi oleh seseorang, kelompok atau organisasi. *World Wide Web* (*WWW*) adalah suatu metode untuk menampilkan informasi di internet, baik berupa text, gambar, suara maupun video yang interaktif dan mempunyai kelebihan untuk menghubungkan (*link)* satu dokumen dengan dokumen yang lainnya (*hypertext*) yang dapat diakses melalui *browser*(Yuhefizar, 2008).Salah satu keuntungan membangun aplikasi berbasis *website* adalah karena *website* tidak tergantung pada *platform* atau operating system perangkat yang digunakan, karena *website* dijalankan menggunakan *internet* *browser* yang dipasang pada operating system perangkat. Selain itu, informasi pada *website* terpusat pada *server*, sehingga ketika web *developer* melakukan perubahan atau modifikasi terhadap *website*, pengguna tidak perlu melakukan *upgrade* terhadap aplikasi web, karena pengguna mengakses server tempat *website* diletakkan.

Klasifikasi Website

Secara fisik, website adalah kumpulan komputer pribadi, web browser, koneksi ISP, komputer server, router dan switch yang digunakan untuk mengalirkan informasi yang menjadi wahana pertama bagi pihak terkait. Berdasarkan fungsinya, web diklasifikasikan menjadi 4 (empat) jenis, yaitu (Yuhefizar, 2008) :

1. Search engine website. Jenis ini merupakan website yang memiliki kemampuan untuk melakukan pencarian berdasarkan kata kunci (keyword). Pencarian dilakukan pada server dengan tujuan menyuguhkan informasi yang lebih fariatif bagi pengguna. Website yang ditampilkan oleh search engine didasarkan pada SEO (Search Engine Optimization) yang dimiliki oleh setiap website. Menurut Matt Cutts (Head of Webmaster Google Spam), SEO dapat sangat berdampak pada index website anda ketika seseorang melakukan pencarian pada search engine jika dikombinasikan dengan optimasi lain dan berdampak pada pengalaman pengguna dan dapat mendongkrak kinerja website
2. Portal website. Jenis website ini berisi kumpulan link dan search engine terhadap seluruh halaman utama dari webiste tersebut. Salah satu contoh portal website yang ada saat ini adalah http://it-ebooks.info. Website ini berisi kumpulan ebooks tentang dunia IT yang dapat diunduh secara gratis dan menyediakan fitur search engine yang mengarahkan kita pada halaman Google yang khusus mencari tentang ebooks.
3. Content website. Digunakan oleh seseorang, sekelompok orang, organisasi, pemerintah bahkan negara untuk membagikan informasi 1 (satu) arah tentang pemilik webiste kepada pengguna. Salah satu contoh dari content website adalah http://www.mikroskil.ac.id yang berisi informasi tentang STMIK – STIE Mikroskil.
4. Application website. Website ini lebih kompleks dibandingkan 3 (tiga) jenis website yang lain karena application website memberikan wadah bagi pengguna untuk berinteraksi dengan pengguna untuk menjalankan fungsi utama dari website. Salah satu contoh dari website ini adalah https://www.lazada.com yang dapat digunakan oleh pengguna untuk membeli barang secara online.

Segmentasi untuk tampilan website dari perangkat yang heterogen

Secara ideal, sebuah *website* harus kompatibel degan perangkat berbasis *desktop*, *tablet*, dan *smartphone*. Seorang pengembang web harus memperhatikan beragamnya perangkat *device* yang digunakan pengguna, karena hal ini berkaitan dengan informasi yang akan ditampilkan sesuai perangkat yang digunakan untuk mengakses halaman *website.* Akses informasi tidak dibatasi oleh waktu maupun peralatan yang digunakan untuk mengaksesnya. Perubahan prilaku orang dalam mencari dan menemukan informasi sudah berubah secara signifikan. Pada tahun 2002, google melakukan studi dan menemukan bahwa 38% aktifitas setiap hari yang kita lakukan melalui *.* Sedangkan hasil analisa yang diterbitkan Februari 2014 oleh ComScore ([http://comscore.com](http://comscore.com/)) menunjukkan bahwa mayoritas konsumen melakukan multiScreening untuk mengakses Internet melalui ponsel maupun *deskto.* Bahkan *Data Corporation* (http://idc.com*)*  memprediksi perubahan cara akses internet melalui *desktop*, *tablet*, dan *smartphone*  dari tahun 2002-2017. Dan Pada tahun 2015, diprediksi akses internet sebagian besar melalui *smartphone.*

Untuk dapat menghasilkan tampilan website yang dapat diakses melalui perangkat mobile, dihasilkan 2 (dua) pendekatan yang dapat digunakan: 1). lowest-common-denominator dan membuat website yang sederhana sehingga dapat diakses dengan lebih cepat; dan 2). menciptakan versi perangkat yang khusus bagi smartphone. (Kulkarni, C., 2011). Namun perkembangan smartphone juga pada akhirnya mematahkan kedua ide inovasi ini. Hingga pada akhirnya dikemukakanlah beberapa metode yang dapat digunakan untuk menghasilkan desain yang responsive, seperti: fixed, fluid, adaptive dan responsive.

Agar sebuah dapat kompetibel dengan berbagai peralatan yang heterogen . Dengan memanfaatkan bebrapa CSS Framework yang sudah ada saat ini, akan semakin mudah dalam menguraikan blok segmentasi berdasarkan elemen DIV kemudian ditambahkan Class pada setiap DIV sesuai blok segmentasi. Setiap kelas memiliki nilai properti dalam CSS yang akan menentukan tampilan sebuah *website*. Dan bergantung pada perangkat yang digunakan. Tujuan utama segmentasi adalah memudahkan *designer front-end website* untuk menentukan konten mana yang diatur tata letaknya ketika *website dibuka dengan resolusi lain.*

* + - 1. *API*

*Application program interface (API)* adalah sekumpulan perintah, fungsi, serta protokol yang digunakan oleh programmer untuk membangun sebuah aplikasi, API menentukan bagaimana komponen perangakat lunak harus berinteraksi. Selain itu, API digunakan ketika promrograman komponen pada Atar muka User (Grafical User Interface). Sebuah API yang baik akan membuatnya lebih mudah untuk mengenbangkan program dengan menyediakan semua komponen, dan programmer kemudian menempat semua komponen secara bersama-sama (Beal, V., 2016). API bekerja seperti middleware pada sebuah aplikasi dengan tujuan yang spesifik dan tidak bekerja pada operating system/platform tempat aplikasi dibangun. beberapa keuntungan menggunakan API :

1. Portabilitas. Aplikasi yang dibangun dengan memanfaatkan API dapat dijalankan pada platform/operating system apapun. Hal ini dikarenakan cara kerja dari API adalah dengan menggunakan system call interface yang dibuat sendiri oleh vendor yang berfungsi sebagai penghubung antara API dan system call operating system. Sehingga programmer tidak perlu memikirkan system call operating system yang akan digunakan pada platform atau interface tertentu.
2. Lebih mudah dimengerti karena menggunakan bahasa yang lebih terstruktur dan mudah dimengerti dibanding system call operating system. Hal ini membuat programmer tidak perlu mempelajari system call dari setiap platform yang akan digunakan.

# Daftar Pustaka

Beal, V., 2016, API – *application programming interface,* tersedia pada: <http://www.webopedia.com/term/a/api.html>, tanggal akses: 8 Februari 2017

website,-The Free Dictionary by Farlex, tersedia pada: <http://www.thefreedictionary.com/Website>, tanggal akses 8 Februari 201

<https://www.academia.edu/22961059/Segmentasi_dan_Klasifikasi_untuk_Tampilan_Website_dari_Perangkat_yang_Heterogen>, tanggal akses 9 Februari 2017

<https://www.openhub.net/p/android>. Tanggal akses 20 Februari 2017

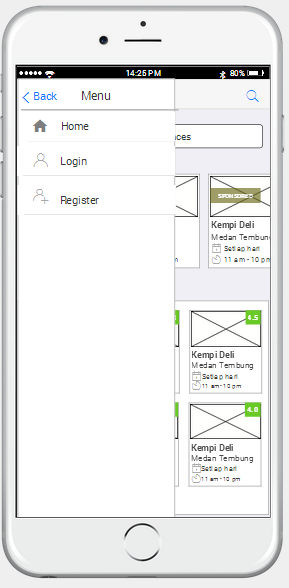
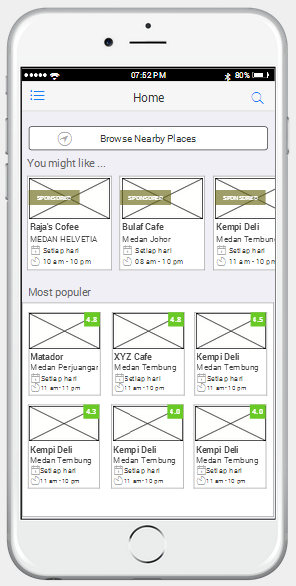
Hafid, A., M., 2015, Arsitektur Sistem Operasi dan Penjelasan Lengkap, tersedia

pada:<http://www.blog-hafid25.com/2015/09/arsitektur-sistem-operasi-dan.html>, tanggal akses 20 Fevruari 2017

# Lampiran

1. ***Mobile Version***
   1. *Homepage & menu (guest)*

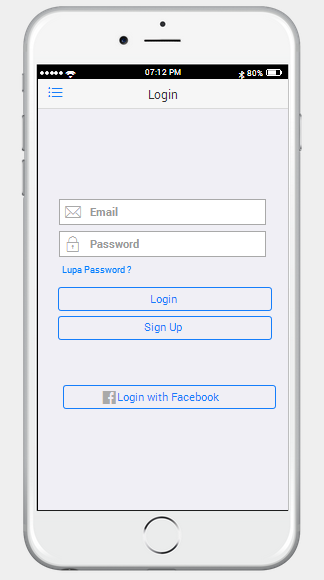
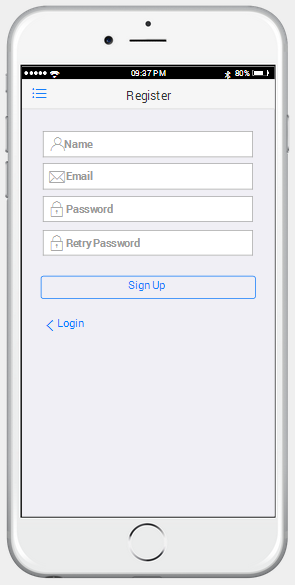
*Homepage* merupakan halaman yang pertama kali muncul pada saat aplikasi dijalankan, *guest* merupakan sebutan untuk user yang menggunakan aplikasi namun belum *login*. Pada aplikasi ini *guest* hanya bisa menggunakn fitur pencarian dan melihat informasi tentang kafe.



Gambar 6. Homepage & menu (guest)

Keterangan :

1. Merupakan *button* untuk *show-hide* menu.
2. *Browse nearby places* untuk menampilkan kafe terdekat dari lokasi pengguna
3. Kafe yang di rekomenadasi oleh sistem, sebagai wadah sponsor untuk pemilik kafe.
4. Tampilan kafe terpopuler bedasarkan rating, apabila kafe tersebut di klik maka akan menampilkan informasi tentang kafe tersebut.
   1. *Login page & Register page*



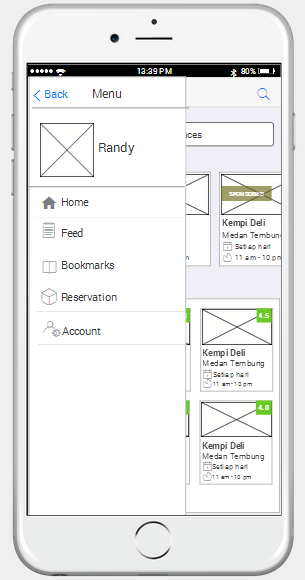
Keterangan :

1. Apabila data yang dimasukkan telah benar, sistem akan otomatis membuat akun dengan *email* dan *password* tersebut untuk dapat *login* ke sistem.
2. Apabila *email* dan *password* telah benar, maka user akan diarahkan ke *homepage*.
3. User dapat login menggunakan akun facebook.

Gambar 7. Register Page & Login Page

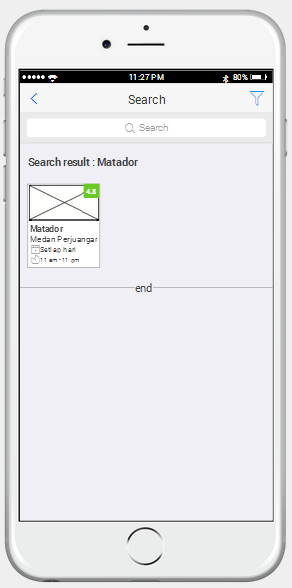
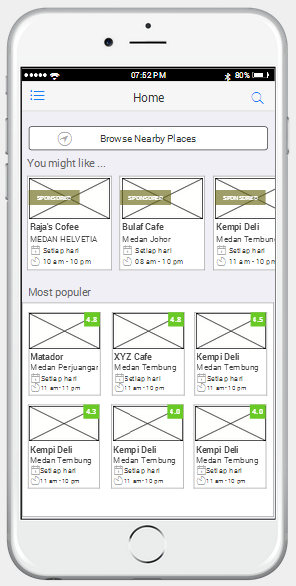
* 1. *Menu (user) & View profil*

Gambar 8. Menu (user) & profil Page



Keterangan :

1. Apabila di klik maka sistem akan mengarahkan aplikasi ke halaman *view profile*.
   1. *Searchpage*

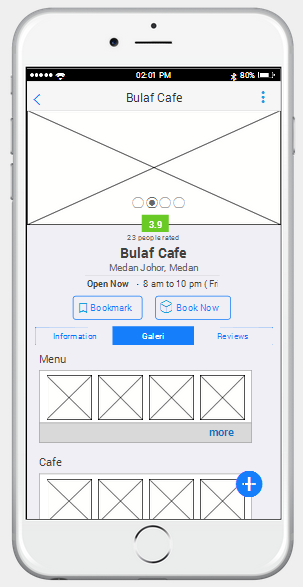
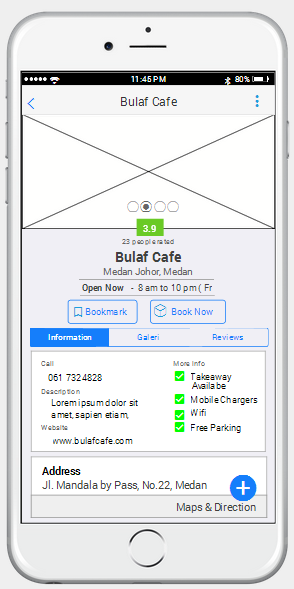
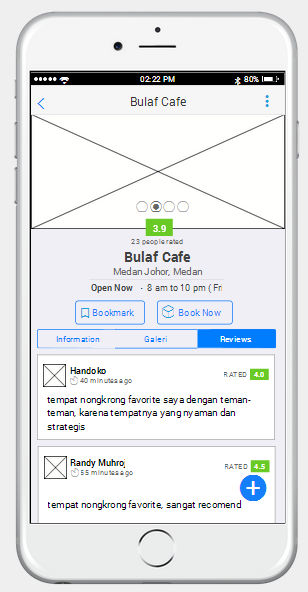


Gambar 9. Homepage & Search page

Keterangan :

1. Sistem akan mengarakan aplikasi ke halaman pencarian.
2. *Form* untuk memasukkan *keyword* yang ingin dicari seperti nama kafe.
3. Hasil pencarian.
   1. *Detail café page*

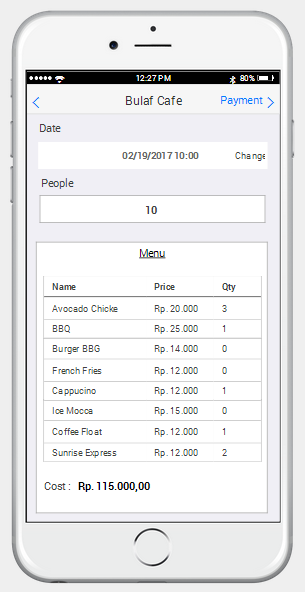
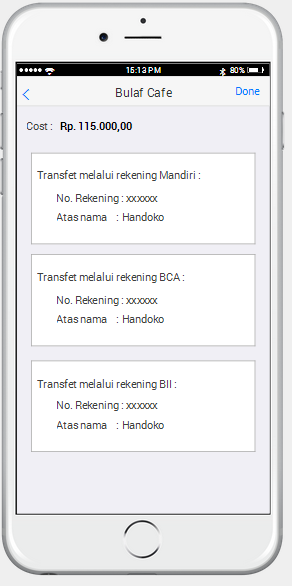
Gambar 10. Detail Café page



***z***

Keterangan :

1. *Slider* yang dimasukkan oleh pemilik kafe berupa tampilan kafe tersebut.
2. *Rating* yang dimiliki kafe tersebut.
3. Menampilkan setatus kafe pada waktu sekarang, apakah lagi buka atau tutup.
4. Bookmark kafe tersebut, yang berfungsi untuk menyimpan kafe tersebut, dan sistem secara otomatis menyimpan pada bookmarks page.
5. *Reservasi* kafe tersebut
6. Informasi tentang kafe seperti alamat, deskripsi, dan fasilitas yang dimiliki kafe tersebut
7. Galeri pada kafe tersebut, seperti gambar kafe, gambar menu, dan lain-lain.
8. Daftar *review* pada kafe tersebut.
9. Apabila user ingin memberikan *review* pada kafe tersebut,
   1. *Booking page & Payment page*

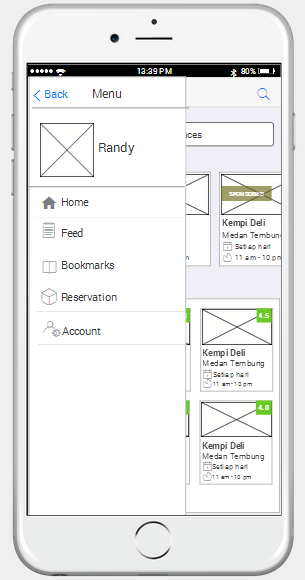


Gambar 11. Booking page & Payment page

Keterangan :

1. *User* dapat mengganti tanggal dan waktu untuk reservasi kafe.
2. *User* dapat mengubah banyak orang, yang nantinya akan dipersiapakan mejanya oleh pemilik kafe.
3. Daftat menu yang ingin dipesan oleh *user*.
4. Apabila data telah diisi, maka user akan diarahkan ke halaman payment
5. *User* akan diberkan waktu 1 jam untuk transer uang ke salah satu rekening tersebut, apabila telah ditransfer lalu user menekan *button* tersebut.
   1. *Feed page*

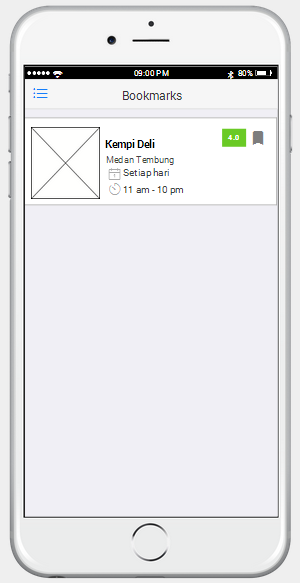
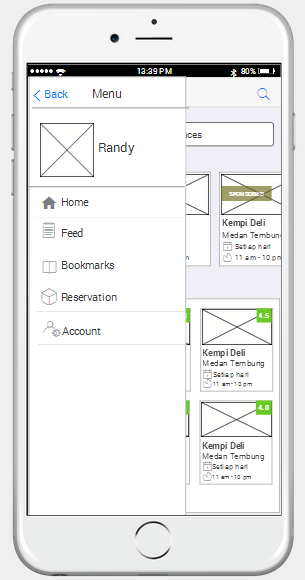
*Feedpage* merupakan halaman yang menampilkan review dari semua user terhadap kafe, dan akan diurutkan berdasarkan yang terbaru.



Gambar 12. Feedpage

* 1. *Bookmarks page*

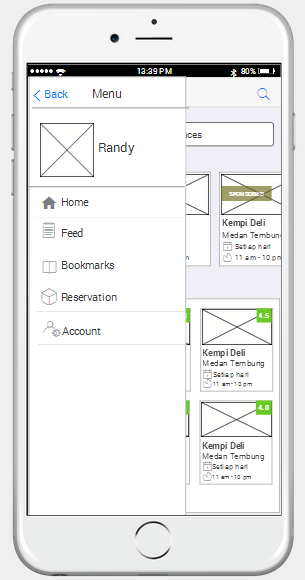
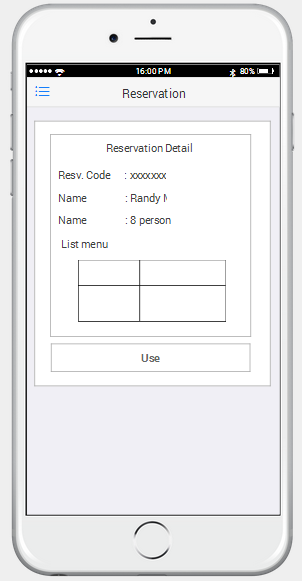
Gambar 13. Bookmarks Page



Keterangan :

1. Tampilan kafe yang telah di *bookmarks*.
   1. *Reserv Page*

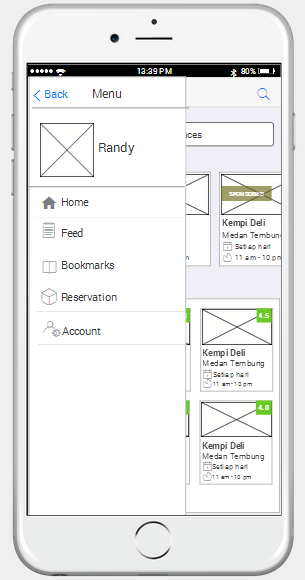
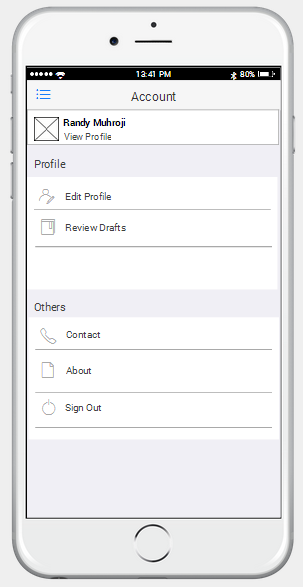
Gambar 14. Reserv Page



Keterangan :

1. Tampilan ini digunakan sebagai bukti reservasi kepada pemilik kafe, bahwa *user* telah melakukan reseervasi terhadap kafe tersebut.
2. Apabila *user* telah reservasi, maka pemilik kafe menekan *button* tersebut, sebagai bahwa bahwa struk reservasi telah digunakan.
   1. *Account page*

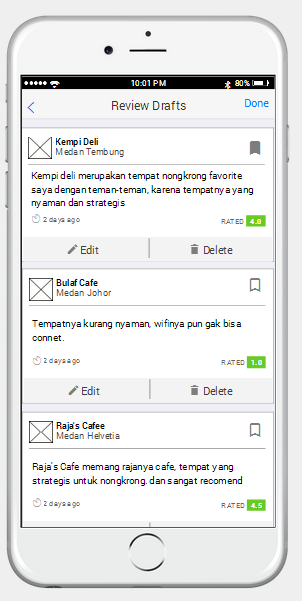
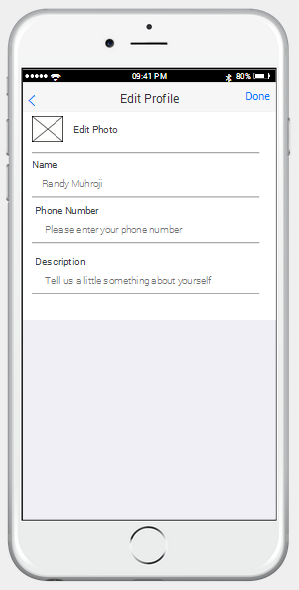
Gambar 15. Account Page



Keterangan :

1. Untuk memperbaharui user profile seperti nama, nomot telephone, dan sebagainya.
2. Untuk menampilkan semua aktivitas review user terhadap kafe.
   1. *Edit profile page & Review drafts page*

Gambar 16. Review Drafts Page & Edit profile page



Keterangan :

1. Isi review yang dilakukan user.
2. Apabila user ingin memperbaharui isi review.
3. Apabila user ingin menghapus revies.
4. Form untuk mengedit profile.