PEMBANGUNAN APLIKASI KEDAI MAMAH EHA PADA *PLATFORM* ANDROID

Gabriella Citra A[1]

[1] Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
Email : citragabriella@gmail.com[1]

ABSTRAK

Kedai Mamah Eha merupakan salah satu tempat makan yang ada di kota Bandung. Banyak makanan dan minuman yang ditawarkan di Kedai Mamah Eha ini, salah satu menu unggulan disana adalah olahan ayam penyetnya. Saat ini, untuk memberikan informasi kepada konsumen mengenai menu makanan dan minuman yang ada di Kedai Mamah Eha masih menggunakan buku menu. Dalam buku menu tersebut berisi informasi nama-nama makanan dan minuman beserta harganya. Untuk melakukan pemesanan, konsumen harus memanggil pelayan untuk menuliskan pesanannya. Setelah melakukan pemesanan, konsumen tidak mengetahui apakah pesanannya sedang diproses atau belum sehingga sering kali konsumen menanyakan pesanannya ke pelayan.

Untuk membantu konsumen dalam pemesanan makanan dan memantau pesanan bisa dilakukan dengan beberapa cara salah satunya yaitu dengan pembangunan aplikasi menu digital untuk membantu konsumen dalam memesan makanan, dan membuat status pemesanan untuk membatu konsumen dalam memantau proses pemesanan menu makanan dan minuman.

Berdasarkan hasil pengujian, maka dapat diperoleh kesimpulan bahwa aplikasi yang dibangun bermanfaat untuk konsumen dan pihak Kedai Mamah Eha, aplikasi ini dapat memudahkan konsumen dalam melakukan proses pemesanan dan memantau proses pemesanan makanan. Selain itu, untuk tampilan aplikasi menarik untuk digunakan olek konsumen maupun pihak Kedai Mamah Eha.

Kata kunci : aplikasi, menu, digital, kedai, mamah, eha

1. PENDAHULUAN

Kedai Mamah Eha merupakan salah satu tempat makan yang ada di kota Bandung. Banyak makanan dan minuman yang ditawarkan di Kedai Mamah Eha ini, salah satu menu unggulan disana adalah olahan ayam penyetnya. Kedai Mamah Eha mempunyai konsep tempat yang modern dan didukung dengan suasana tempat yang nyaman. Berdasarkan hasil wawancara dengan pemilik Kedai Mamah Eha bahwa seiring dengan makin populernya Kedai Mamah Eha maka makin banyak juga konsumen yang datang untuk makan ke tempat tersebut. Saat ini, untuk memberikan informasi kepada konsumen mengenai menu makanan dan minuman yang ada di Kedai Mamah Eha masih menggunakan buku menu. Dalam buku menu tersebut berisi informasi nama-nama makanan dan minuman beserta harganya. Untuk melakukan pemesanan, konsumen harus memanggil pelayan untuk menuliskan pesanannya. Setelah melakukan pemesanan, konsumen tidak mengetahui apakah pesanannya sedang diproses atau belum sehingga sering kali konsumen menanyakan pesanannya ke pelayan.

Selain melayani pemesanan makanan dan minuman di tempat, Kedai Mamah Eha juga memiliki layanan pesan antar (*delivery service*) dengan melakukan pemesanan melalui telepon dan SMS. Namun cara tersebut dirasa kurang efektif karena banyak konsumen yang sering kali menanyakan kembali pesanannya karena konsumen tidak bisa memantau langsung pesanan yang mereka pesan. Selain itu pemilik Kedai Mamah Eha juga memiliki kendala dalam melakukan perekapan nota pemesanan karena harus direkap secara manual dari setiap nota pemesanan yang ada, sehingga waktu untuk pembuatan laporan penjualan menjadi lama.

Maksud dari penelitian ini yaitu untuk membantu menyelesaikan permasalahan yang ada di Kedai Mamah Eha. Maka perlu dibangun aplikasi Kedai Mamah Eha pada platform Android.

Adapun tujuan dari dibuatnya Aplikasi Kedai Mamah Eha Pada *Platform* Android adalah sebagai berikut:

1. Memudahkan konsumen dalam memesan dan memantau makanan yang dipesan di tempat maupun secara *delivery service*.
2. Membantu konsumen untuk mengetahui sudah sampai mana pesanan tersebut dikirim saat ini.
3. Memudahkan pemilik Kedai Mamah Eha dalam pembuatan laporan penjualan.

1.1 Aplikasi *Mobile*

Pengertian aplikasi adalah program yang digunakan orang untuk melakukan sesuatu pada

sistem komputer. *Mobile* dapat diartikan sebagai perpindahan yang mudah dari satu tempat ke tempat yang lain, misalnya telepon *mobile* berarti bahwa terminal telepon yang dapat berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Sistem aplikasi *mobile* merupakan aplikasi yang dapat digunakan walaupun pengguna berpindah dengan mudah dari satu tempat ketempat lain lain tanpa terjadipemutusan atau terputusnya komunikasi. Aplikasi ini dapat diakses melalui perangkat nirkabel seperti pager, seperti telepon seluler dan PDA. Adapun karakteristik perangkat *mobile* yaitu[9]:

1. Menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.
2. Memory yang terbatas : Perangkat *mobile* juga memiliki memory yang kecil, yaitu *primary (RAM)* dan *secondary (disk)*.
3. Daya proses yang terbatas : Sistem *mobile* tidaklah setangguh rekan mereka yaitu *desktop.*
4. Mengkonsumsi daya yang rendah : Perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin desktop
5. Kuat dan dapat diandalkan : Karena perangkat *mobile* selalu dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan, dan sesekali tetesan-tetesan air.
6. Konektivitas yang terbatas : Perangkat *mobile* memiliki bandwith rendah, beberapa dari mereka bahkan tidak tersambung.

1.2 JSON

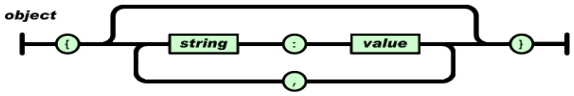
JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data[10]. JSON terbuat dari dua struktur:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
2. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemprograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemprograman yang juga berdasarkan pada struktur data ini.

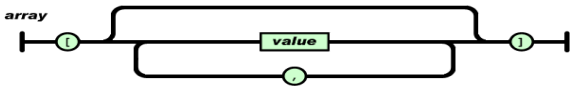
JSON menggunakan bentuk sebagai berikut:

1. **Objek** adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).



Gambar 1 JSON Object

2. **Larik** adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).



Gambar 2 JSON Array

2. ISI PENELITIAN

2.1 Analisis Komunikasi Data

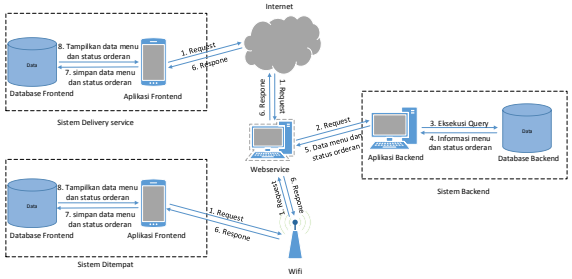
Analisis komunikasi data menjelaskan cara pengiriman dan penerimaan data antara *frontend* dan *backend*. Dalam penelitian ini pertukaran data menggunakan teknik *JSON*, dimana aplikasi *frontend* akan mengirimkan *request* kepada aplikasi *backend* melalui jaringan internet, kemudian aplikasi *backend* akan mengirim respon kepada aplikasi *frontend* sesuai dengan *request* yang dikirim oleh aplikasi *frontend*.

Pertama, aplikasi *client* akan mengirim *request* ke aplikasi *backend* melalui layanan internet,

kemudian aplikasi *backend* menerima *request* tersebut. Selanjutnya aplikasi *backend* akan melakukan eksekusi *query* terhada *database server*. Hasil dari eksekusi yaitu informsi data materi, data teori dan rumus, data contoh soal, dan data latihan soal.

Setelah itu, aplikasi *backend* akan mengirim respon ke aplikasi *frontend*, respon tersebut berupa data hasil dari eksekusi *query* sesuai dengan *request* yang dikirim oleh aplikasi *frontend* tadi. Setelah aplikasi *frontend* menerima respon dari aplikasi *backend* maka respon tersebut akan disimpan sementara pada database *frontend* pada aplikasi *frontend* untuk ditampilkan kepada siswa.

Berikut adalan gambaran komunikasi data pada aplikasi Kedai Mamah Eha yang akan dibangun.



Gambar 3 Mekanisme komunikasi data aplikasi Kedai Mamah Eha

2.2 Analisis Sistem Yang Sedang Berjalan

1. Prosedur Pemesanan di Tempat

Berikut adalah prosedur pemesanan makanan dan minuman ditempat yang sedang berjalan:

a. Konsumen memanggil pelayan untuk memesan menu makanan dan minuman.
b. Pelayan mendatangi konsumen dan memberikan buku menu.
c. Konsumen memilih menu makanan dan minuman yang ada dalam buku menu tersebut.
d. Pelayan menanyakan menu apa saja yang akan dipesan oleh konsumen.
e. Konsumen menyebutkan menu yang dipesan kepada pelayan.
f. Pelayan menuliskan pesanan pada sebuah kertas kecil yang nantinya akan diberikan kepada koki.

2. Prosedur Pemesanan untuk *delivery service*

Berikut adalah prosedur pemesanan makanan dan minuman untuk *delivery service*:

a. Konsumen melakukan pemesanan makanan dan minuman dengan cara sms atau telepon ke Kedai Mamah Eha.
b. Pelayan menanyakan menu apa saja yang akan dipesan oleh konsumen.
c. Konsumen menyebutkan menu yang akan dipesan.
d. Pelayan mencatat menu yang dipesan oleh konsumen.
e. Pelayan menanyakan alamat konsumen untuk mengirim pesanannya.
f. Konsumen menunggu pesanannya datang.
g. Konsumen membayar pada saat pesanannya sampai.

3. Prosedur Pendaftaran member

Berikut ini adalah prosedur yang sedang berjalan untuk melakukan pendaftaran menjadi member Kedai Mamah Eha:

a. Konsumen mendatangi petugas.
b. Petugas memasukan data yang dibutuhkan untuk menjadi seorang member.
c. Konsumen memberikan biaya pendaftaran sebagai deposit untuk kartu member.
d. Petugas memberikan kartu member sebagai tanda bahwa konsumen tersebut sudah terdaftar sebagai member Kedai Mamah Eha.

2.3 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi Kebutuhan Perangkat Lunak berisi mengenai semua kebutuhan dalam perancangan kebutuhan perangkat lunak, diantarnya spesifikasi kebutuhan fungsional dan non-fungsional.

Tabel 1 Kebutuhan Fungsional

Nomor	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-F-01	Admin melakukan *Login* untuk masuk ke dalam sistem
SKPL-F-02	Sistem dapat membantu admin dalam melakukan pengolahan data makanan dan minuman yang meliputi tambah, ubah, hapus.
SKPL-F-03	Sistem dapat membantu admin dalam melakukan pengolahan data member yang meliputi tambah, ubah, hapus.
SKPL-F-04	Sistem dapat membantu admin dalam melakukan pengolahan data pemesanan makanan dan minuman
SKPL-F-05	Sistem dapat membantu admin dalam melakukan pengolahan data penjualan dan laporan penjualan
SKPL-F-06	Sistem dapat membantu konsumen dalam memesan makanan dan minuman langsung di tempat ataupun *delivery service*.
SKPL-F-07	Sistem dapat membantu konsumen dalam memantau pesanannya.

Tabel 2 Kebutuhan Non-Fungsional

Nomor	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-NF-01	Sistem dibangun dalam bentuk website untuk *backend* dan mobile untuk *frontend*.
SKPL-	Sistem yang dibangun bersifat online,

NF-02	sehingga dibutuhkan koneksi internet untuk mengaksesnya.
SKPL-NF-03	Bahasa pemrograman yang digunakan dalam membangun aplikasi ini adalah PHP dan MYSQL untuk aplikasi *backend*, sedangkan untuk aplikasi *frontend* mengunakan JAVA dan SQLite.
SKPL-NF-04	Format pertukaran data yang digunakan yaitu dengan format JSON.
SKPL-NF-05	Data yang diolah yaitu data menu makanan dan minuman, data member, data pemesanan, serta data laporan penjualan.

2.4 Analisis Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini yaitu komputer dan *smartphone*. Komputer digunakan dalam merancang dan membangun aplikasi Kedai Mamah Eha. Sedangkan *smartphone* digunakan sebagai alat untuk implementasi aplikasi Kedai Mamah Eha yang akan dibangun pada penelitian ini.

Adapun kebutuhan spesifikasi minimum komputer dan *smartphone* yang digunakan dalam penelitian ini dapat dilihat pada tabel 3 dan tabel 4.

Tabel 3 Spesifikasi Perangkat Keras Komputer

No	Spesifikasi	
1	Prosesor	2.4 GHz
2	Memori	2 GB
3	VGA	2 GB
4	Harddisk	500 GB

Tabel 1 Spesifikasi Perangkat Keras *Smartphone*

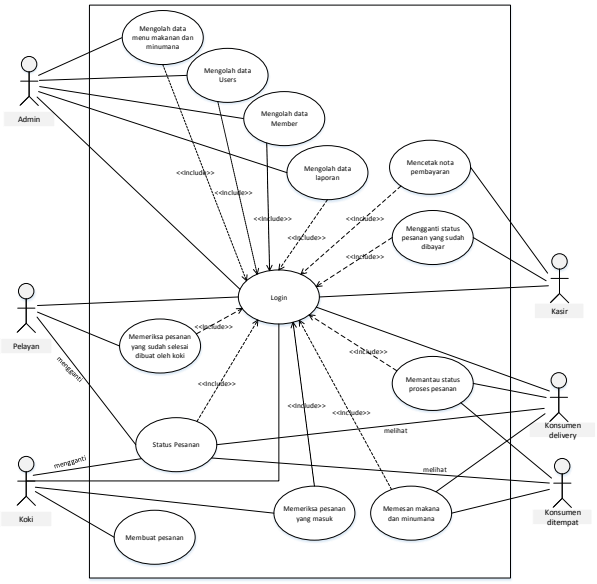
No	Spesifikasi	
1	Memori Internal	4 GB
2	Memori Eksternal	Up to 32 GB
3	Layar	4 inchi
4	Resolusi	HVGA (420x800)
5	Jaringan	Online

1.5 Use Case

Diagram *Use Case* merupakan deskripsi fungsi dari sebuah sistem dan perspektif pengguna. Diagram ini juga mendeskripsikan apa yang akan dilakukan oleh sistem.

Tabel 5 Aktor sistem

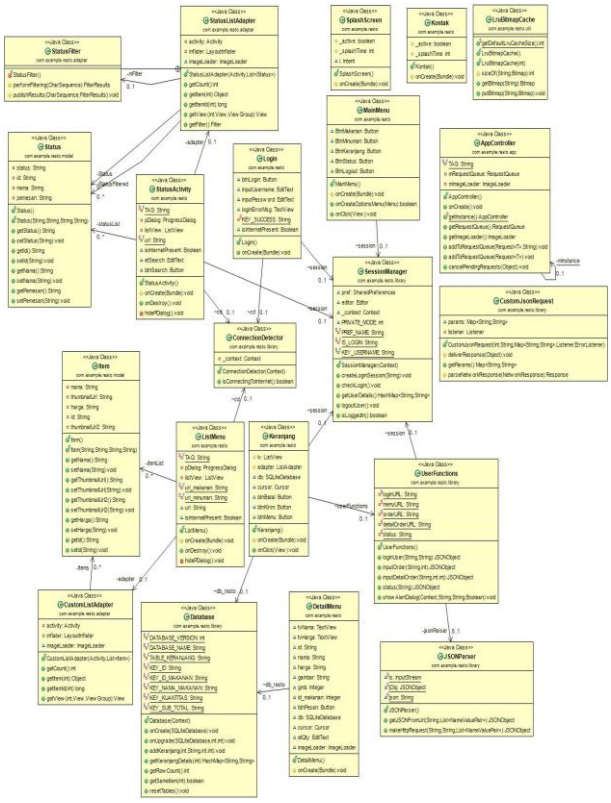
No	Aktor	Deskripsi
1	Admin	Aktor ini merupakan orang yang mengelola data yang berhubungan dengan aplikasi Kedai Mamah Eha yang meliputi data menu makanan dan minuman, data pegawai, data member. Selain itu admin juga bertugas membuat laporan penjualan.
2	Koki	Aktor ini merupakan orang yang memeriksa pesanan yang masuk dan mengganti status pesanan yang sedang dan sudah dimasak.
3	Pelayan	Aktor ini merupakan pengguna dari aplikasi Kedai Mamah Eha yang bertugas memeriksa pemesanan makanan dan minuman serta bertugas mengganti status pemesanan tersebut.
4	Kasir	Memeriksa pesanan yang belum dibayar dan mengganti status pesanan yang sudah dibayar.
5	Konsumen	Aktor ini merupakan pengguna dari aplikasi Kedai Mamah Eha yang bisa melakukan pemesanan makanan dan minuman serta memantau proses *delivery service*.



Gambar 4 *Use Case*

1.6 Class Diagram

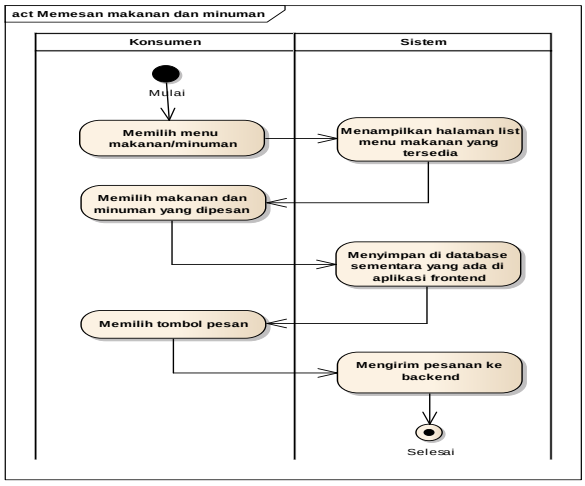
Diagram kelas (*Class diagram*) merupakan diagram struktural yang memodelkan sekumpulan kelas, interface, kolaborasi dan relasinya. Diagram kelas digambarkan dengan kotak yang pada dasarnya terbagi atas tiga bagian, yaitu: nama kelas, atribut, dan opeerasi. Diagram kelas digunakan untuk menggambarkan proses statik dari aplikasi Kedai Mamah Eha ini. Berikut ini adalah kelas diagram dari aplikasi Kedai Mamah Eha



Gambar 5 *Class Diagram*

1.7 Activity Diagram

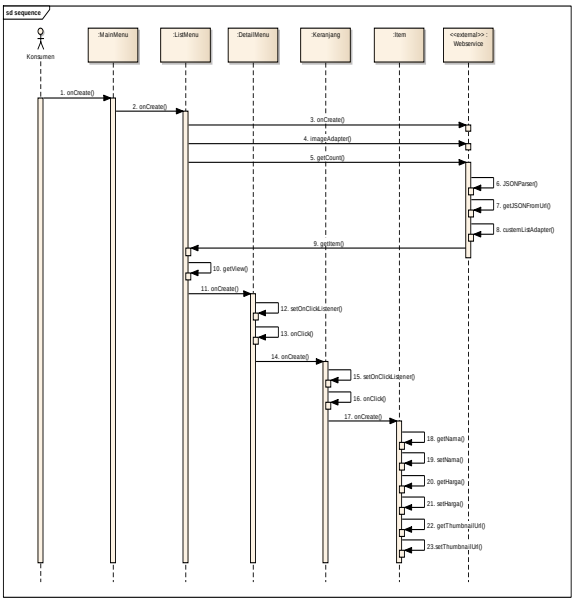
Activity Diagram merupakan suatu diagram yang mendeskripsikan logika procedural, proses bisnis, dan aliran kerja dalam banyak kasus. *Activity Diagram* memiliki peranan seperti halnya *flowchart*, akan tetapi perbedaan antara *activity diagram* dengan *flowchart* yaitu bisa mendukung perilaku pararel sedangkan *flowchart* tidak bisa. *Activity Diagram* memodelkan *event-event* yang terjadi didalam suatu *Use Case* dan digunakan untuk pemodelan aspek dinamis dari sistem.



Gambar 6 *Activity Diagram* Memesan Makanan dan Minuman

1.8 Squence Diagram

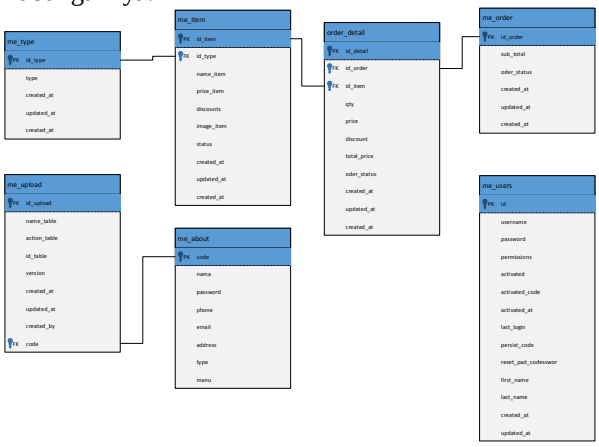
Squence Diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukan sejumlah contoh objek dan pesan yang diletakan diantara objek-objek ini didalam *use cas.*



Gambar 7 *Sequence Diagram* Memesan Makanan dan Minuman

1.9 Skema Relasi

Relasi antara tabel merupakan gabungan antar file yang mempunyai kunci utama yang sama, sehingga *file-file* tersebut menjadi satu kesatuan yang dihubungkan oleh *field* kunci (*Primary Key*). Pada proses ini elemen-elemen data dikelompokan menjadi satu *file* database beserta entitas dan hubungannya.



Gambar 8 Skema relasi

1.10 Implementasi Basisdata

Implementasi basisdata diambil berdasarkan perancangan data yang dibuat sebelumnya. Secara fisik, implementasi basisdata diimplementasikan menggunakan perangkat lunak MySQL sebagai *database server* yang ditempatkan pada aplikasi *backend*. Tabel-tabel berikut menggambarkan struktur tabel yang diimplementasikan pada *database server* dan *database frontend.*

a. Database Server

1. Tabel me_about

Tabel 6 Tabel me_about

```
CREATE TABLE `me_about` (
  `code` varchar(20) NOT NULL,
  `name` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL,
  `phone` varchar(30) NOT NULL,
  `email` varchar(100) NOT NULL,
  `address` text NOT NULL,
  `type` enum('Center','Branch') NOT NULL D
EFAULT 'Branch',
  `menu` enum('Full','Own') NOT NULL,
  `updated_at` timestamp NOT NULL DEFAU
LT CURRENT_TIMESTAMP ON UPDATE C
URRENT_TIMESTAMP,
  `created_at` timestamp NOT NULL DEFAUL
T '0000-00-00 00:00:00',
  PRIMARY KEY (`code`),
  UNIQUE KEY `kode` (`code`)
) ENGINE=InnoDB DEFAULT CHARSET=lat
in1
```

2. Tabel me_item

Tabel 7 Tabel me_item

```
CREATE TABLE `me_item` (
  `id_item` int(11) NOT NULL AUTO_IN
CREMENT,
  `id_type` int(11) DEFAULT NULL,
  `name_item` varchar(100) DEFAULT NU
LL,
  `price_item` double DEFAULT NULL,
  `discounts` double NOT NULL,
  `image_item` varchar(200) DEFAULT N
ULL,
  `image_item_lg` varchar(100) NOT NUL
L,
  `status` int(1) NOT NULL DEFAULT '0',
  `created_at` timestamp NULL DEFAULT
 NULL,
  `updated_at` timestamp NULL DEFAUL
T NULL,
  `created_by` varchar(5) DEFAULT NUL
L,
  PRIMARY KEY (`id_item`),
  KEY `id_type` (`id_type`)
) ENGINE=InnoDB AUTO_INCREMENT
=15 DEFAULT CHARSET=latin1
```

3. Tabel me_item_type

Tabel 8 Tabel me_item_type

```
CREATE TABLE `me_item_type` (
  `id_type` int(11) NOT NULL AUTO_INC
REMENT,
  `type` varchar(200) DEFAULT NULL,
  `created_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  `updated_at` timestamp NULL DEFAUL
T NULL ON UPDATE CURRENT_TIME
STAMP,
  `created_by` varchar(5) DEFAULT NUL
L,
  PRIMARY KEY (`id_type`)
) ENGINE=InnoDB AUTO_INCREMENT
=4 DEFAULT CHARSET=latin1
```

4. Tabel me_order

Tabel 9 Tabel me_order

```
CREATE TABLE `me_order` (
  `id_order` int(11) NOT NULL AUTO_IN
CREMENT,
  `sub_total` double NOT NULL,
  `order_status` varchar(10) NOT NULL,
  `created_at` timestamp NOT NULL DEF
AULT CURRENT_TIMESTAMP ON UP
DATE CURRENT_TIMESTAMP,
  `updated_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  `created_by` char(5) NOT NULL,
  PRIMARY KEY (`id_order`)
) ENGINE=InnoDB AUTO_INCREMENT
=9 DEFAULT CHARSET=latin1
```

5. Tabel me_order_detail

Tabel 10 Tabel me_order_detail

```
CREATE TABLE `me_order_detail` (
  `id_detail` int(15) NOT NULL AUTO_IN
CREMENT,
  `id_order` int(11) NOT NULL,
CREATE TABLE `me_order` (
  `id_order` int(11) NOT NULL AUTO_IN
CREMENT,
  `sub_total` double NOT NULL,
  `order_status` varchar(10) NOT NULL,
  `created_at` timestamp NOT NULL DEF
AULT CURRENT_TIMESTAMP ON UP
DATE CURRENT_TIMESTAMP,
  `updated_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  `created_by` char(5) NOT NULL,
  PRIMARY KEY (`id_order`)
) ENGINE=InnoDB AUTO_INCREMENT
=9 DEFAULT CHARSET=latin1  `id_item
` int(11) NOT NULL,
  `own` enum('Y','N') NOT NULL,
```

```
  `qty` int(5) NOT NULL,
  `price` double NOT NULL,
  `discount` int(11) NOT NULL,
  `total_price` double NOT NULL,
  `order_status` varchar(20) NOT NULL,
  `created_at` timestamp NOT NULL DEF
AULT CURRENT_TIMESTAMP ON UP
DATE CURRENT_TIMESTAMP,
  `updated_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  `created_by` char(5) NOT NULL,
  PRIMARY KEY (`id_detail`)
) ENGINE=InnoDB AUTO_INCREMENT
=21 DEFAULT CHARSET=latin1
```

6. Tabel me_update

Tabel 11 Tabel me_update

```
CREATE TABLE `me_update` (
  `id_update` int(11) NOT NULL AUTO_I
NCREMENT,
  `name_table` varchar(50) NOT NULL,
  `action_table` varchar(10) NOT NULL,
  `id_table` int(11) NOT NULL,
  `version` int(11) NOT NULL,
  `created_at` timestamp NOT NULL DEF
AULT CURRENT_TIMESTAMP ON UP
DATE CURRENT_TIMESTAMP,
  `updated_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  `created_by` int(11) NOT NULL,
  `code` varchar(5) NOT NULL,
  PRIMARY KEY (`id_update`)
) ENGINE=InnoDB AUTO_INCREMENT
=373 DEFAULT CHARSET=latin1
```

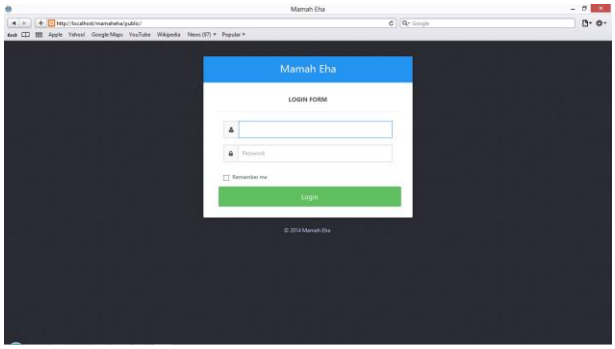
7. Tabel me_users

Tabel 12 Tabel me_users

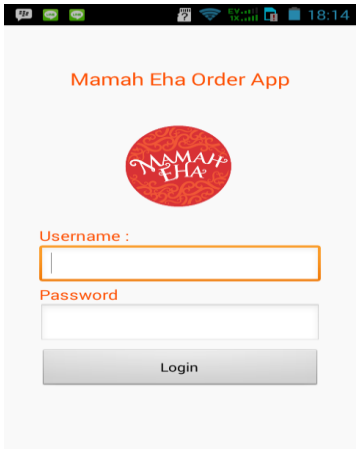
```
CREATE TABLE `me_users` (
  `id` int(10) unsigned NOT NULL AUTO_
INCREMENT,
  `username` varchar(255) COLLATE utf8_
unicode_ci NOT NULL,
  `password` varchar(255) COLLATE utf8_
unicode_ci NOT NULL,
  `permissionz` varchar(7) COLLATE utf8_
unicode_ci DEFAULT NULL,
  `activated` tinyint(1) NOT NULL DEFAU
LT '0',
  `activation_code` varchar(255) COLLAT
E utf8_unicode_ci DEFAULT NULL,
  `activated_at` timestamp NULL DEFAUL
T NULL,
  `last_login` timestamp NULL DEFAULT
NULL,
  `persist_code` varchar(255) COLLATE ut
f8_unicode_ci DEFAULT NULL,
  `reset_password_code` varchar(255) COL
LATE utf8_unicode_ci DEFAULT NULL,
  `first_name` varchar(255) COLLATE utf8
_unicode_ci DEFAULT NULL,
  `last_name` varchar(255) COLLATE utf8
_unicode_ci DEFAULT NULL,
  `created_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEF
AULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`),
  UNIQUE KEY `users_email_unique` (`us
ername`),
  KEY `users_activation_code_index` (`acti
vation_code`),
  KEY `users_reset_password_code_index`
(`reset_password_code`)
) ENGINE=InnoDB AUTO_INCREMENT
=7 DEFAULT CHARSET=utf8 COLLATE
=utf8_unicode_ci
```

1.11 Implementasi antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman yang dibuat baik aplikasi *frontend* yang dibangun di perangkat mobile maupun aplikasi *backend*yang dibangun menggunakan web. Berikut ini beberapa tampilan antarmuka yang telah diimplementasikan.



Gambar 9 Halaman Utama Backend



Gambar 10 Halaman Utama Frontend

1.12 Pengujian Sistem

Pengujian yang dilakukan terhadap sistem penelitian tugas akhir ini adalah pengujian secara fungsional (*alpha*) dan *betha.* Metode yang digunakan yaitu pengujian *blackbox* yang berfokus pada persyaratan fungsional dari sistem (perangkat lunak) yang dibangun.

Tabel 14 Pengujian List Materi

No	Pertanyaan	Jawaban						Total
		SS	S	BS	KS	CS	STS	
1	Menurut anda, apakah aplikasi ini mudah untuk digunakan?	14	22	12	1	1	0	50
2	Menurut anda, apakah tampilan antarmuka aplikasi Kedai Mamah Eha ini menarik?	5	23	20	1	1	0	50
3	Menurut anda, apakah aplikasi Kedai Mamah Eha bermanfaat bagi anda?	12	24	14	0	0	0	50
4	Menurut anda, apakah aplikasi Kedai Mamah Eha sudah mencapai tujuan sebagai salah satu alat bantu untuk memudahkan konsumen dalam memesan makanan?	15	23	12	0	0	0	50
5	Menurut anda, apakah dengan adanya aplikasi Kedai Mamah Eha ini dapat membantu anda dalam memantau proses pesanan yang anda pesan?	8	26	16	0	0	0	50

Berdasarkan hasil pengujian *betha* dapat ditarik kesimpulan bahwa sistem yang dibangun mudah untuk digunakan, bermanfaat dalam membantu konsumen untuk memesan makanan dan mematau proses pesanannya. Selain itu aplikasi ini juga membantu pihak Kedai Mamah Eha dalam pembuatan laporan. Tetapi untuk tampilan dari aplikasi ini masih biasa saja sesuai dengan hasil persentase jawaban setiap responden terhadap pertanyaan yang diajukan.

3. PENUTUP

Kesimpulan dari penelitian skripsi yang berjudul: "Pembangunan Aplikasi Kedai Mamah Eha Pada *Platform* Android" adalah bahwa aplikasi Kedai Mamah Eha yang dibangun dapat menjadi salah satu alat bantu pemesanan menu makanan dan minuman bagi konsumen yang melakukan pembelian. Dengan adanya aplikasi Kedai Mamah Eha yang bisa bersifat moveble maka konsumen bisa memesan makanan dan minuman dimanapun dan kapanpun.

DAFTAR PUSTAKA

[1] Safaat, Nazrudin. 2012. *Pemrograman Aplikasi Mobile Smartphone Dan Tablet PC Berbasis Android.* Informatika. Bandung.

[2] Milette, Greg. Stroud, Adam. 2012. *Professional Android Sensor Programming*. Wiley.

[3] Dharma K. Akhmad. 2013. *Kolaborasi Dahsyat ANDROID dengan PHP dan MySQL*. Lokomedia. Yogyakarta.

[4] JSON. "*Pengenalan JSON*". http://www.json.org/json-id.html (diakses tanggal 12 November 2014).

[5] Harianto, Bambang, Ir.,MT. 2004. *Rekayasa Sistem Berorientasi Objek*. Informatika. Bandung.

[6] Pudjo W, Prabowo. Herlawati. 2011. *Menggunakan UML*. Informatika. Bandung.

[7] Sholiq. 2006. *Pemodelan Sistem Informasi Berorientasi Objek Dengan UML*. Informatika. Bandung.

[8] Harianto, Bambang. 2004. *Sistem Manajemen Basisdata*. Informatika. Bandung.

[9] Murya, Yosef. 2013. *Pemrograman Android Black Box*. Jasakom. Jakarta.

[10] Liu, Chun. 2014. *Principle and Application Progress in Location-Based Service*. Tongji University. Shanghai-China.