

SISTEM LAYANAN ANTAR PADA RESTORAN BERBASIS ANDROID

Endyatna Puthut Bagus Pratama, Sumarsono, Elizabeth Nurmiyati Tamatjita
Sekolah Tinggi Teknologi Adisutjipto, Bantul, Yogyakarta

Abstract

The business restaurant is growing. In this fast business is need application mobile for customer satisfaction. This system his many applications. They are customer, manager, web kitchen and delivery application. This application is using android Operating System. Using that application by maps for knowing the places of customer. The results are 100% for delivery service, 93% to level of effective and efficient, 97% to time service system and 97% of delivery order system.

Keywords: android, restaurant, service

1. Pendahuluan

Perkembangan teknologi yang sangat cepat telah membawa manusia memasuki kehidupan yang berdampingan dengan informasi dan teknologi. Dengan teknologi informasi yang berkembang saat ini, pengelolaan informasi dapat dilakukan secara lebih aktual dan optimal. Penggunaan teknologi informasi bertujuan untuk mencapai efisiensi dalam berbagai aspek pengelolaan informasi, yang ditunjukkan untuk dengan kecepatan dan ketepatan waktu pemrosesan, serta ketelitian dan keakuratan informasi.

Banyaknya bisnis kuliner yang semakin bersaing, disamping menu yang disajikan, layanan restoran juga sangat berpengaruh yaitu dalam memberikan kepuasan kepada pelanggan. Dengan alternatif layanan *delivery* pada restoran akan memudahkan pelanggan dalam memesan makanan tanpa datang ke restoran. Melihat bisnis kuliner dunia nyata yang menerapkan layanan *delivery* restoran secara manual mempunyai dampak kinerja yang kurang efisien dan efektif. Sehingga dalam segi waktu dan pengelolaan restoran bergantung pada kinerja setiap pegawai restoran.

Untuk lebih meningkatkan pelayanan, maka diterapkan sistem aplikasi layanan *delivery* berbasis android. Karena dengan aplikasi tersebut pelanggan dapat melihat daftar menu melalui *smartphone* berbasis android dan memesan order pesanan dengan nyaman tanpa mengurus tenaga. Dengan sistem ini menu makanan yang dipesan oleh pelanggan akan diterima pada *server* yang terdapat pada restoran, kemudian *server* mengolah data pesanan dan menampilkan letak koordinat pemesan melalui GPS (*Global Positioning System*) sehingga memudahkan petugas *delivery* untuk mengetahui alamat pelanggan serta membawa *order* pesanan dari pelanggan dengan aman, nyaman dan tepat sasaran.

Dari uraian di atas akan dibangun sebuah sistem aplikasi "SISTEM LAYANAN ANTAR PADA RESTORAN BERBASIS ANDROID". Dimana aplikasi ini nantinya dapat membantu dan memajukan restoran dalam meningkatkan pelayanan *delivery* yang dapat diakses oleh pelanggan dan memudahkan pemilik restoran dalam mengelola restoran.

2. UML (Unified Modelling Language) dan ORM (Object Relational Mapping)










2.1 UML (Unified Modelling Language)

UML (Unified Modelling Diagram) adalah sebuah bahasa yang berdasarkan grafik/ gambar untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan sebuah sistem pengembangan *software* berbasis OO (Object Oriented). UML mempunyai 13 diagram, yang digunakan untuk membuat pemodelan sistem pada tugas akhir ini adalah 3 diagram yaitu :

1. *Activity diagram*.
2. *Use case description* dan *Use case diagram*.
3. *Class diagram*, adapun *class diagram* yang digunakan adalah ORM (Object Relational Mapping).

2.1.1 Activity Diagram





Diagram activity berfokus pada aktifitas-aktifitas yang terjadi dalam suatu proses tunggal. Diagram ini menunjukkan bagaimana aktifitas-aktifitas tersebut bergantung satu sama lain. Berikut simbol-simbol yang terdapat pada diagram *activity* dapat dilihat pada Gambar 1.

| Simbol | Keterangan |
|---|--|
|  | Titik awal |
|  | Titik Akhir |
|  | Aktifitas |
|  | Pilihan untuk pengambilan keputusan |
|  | Fork; Untuk menunjukkan kegiatan yang dilakukan secara paralel |
|  | Rake; menunjukkan adanya dekomposisi |
|  | Tanda waktu |
|  | Tanda penerimaan |
|  | Aliran akhir (Flow final) |

Gambar 1 Simbol Diagram *Activity*

2.1.2 Use Case Diagram

Use case adalah diagram yang menunjukkan fungsionalitas interaksi antara *actor* dan sistem. *Use case diagram* merupakan gambaran *graphical* dari beberapa atau semua *actor*, *use case*, dan interaksi diantara komponen-komponen yang memperkenalkan suatu sistem yang akan dibangun. Berikut simbol dari *use case diagram* dapat dilihat pada Gambar 2.

| | |
|---|----------------------------------|
|  | Aktor |
|  | <i>Use Case</i> |
|  | Batasan Sistem |
|  | Tingkat Relasi Hubungan |
|  | Relasi Hubungan <i>Include</i> |
|  | Relasi Hubungan <i>Extends</i> |
|  | Garis Penghubung Relasi Hubungan |

Gambar 2 Simbol *Use Case*

2.1.3 Class Diagram

Class Diagram adalah sebuah spesifikasi yang apabila diinstansiasi akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package*, dan *object* beserta hubungan satu sama lain seperti *containtment*, pewarisan, asosiasi, dan lain-lain. Berikut simbol *class diagram*. Dapat dilihat pada Gambar 3.

| | | | | |
|---|-------------------------|---------------|------------------|-------|
| <table border="1"> <tbody> <tr> <td>Class 1</td> </tr> <tr> <td>- Attribute 1</td> </tr> <tr> <td>- Operation 1 ()</td> </tr> </tbody> </table> | Class 1 | - Attribute 1 | - Operation 1 () | Kelas |
| Class 1 | | | | |
| - Attribute 1 | | | | |
| - Operation 1 () | | | | |
| Attribute name/derived attribute name | Atribut | | | |
| Operation name () | Nama Operasi/Metode | | | |
| <u>1..* Verb phrase 0..1</u> | Tingkat Relasi/Hubungan | | | |

Gambar 3 Simbol *Class Diagram*

2.2 ORM (Object Relational Mapping)

ORM (*Object Relational Mapping*) melakukan pemetaan tabel-tabel pada basis data relasional dengan suatu *class* yang ada pada bahasa pemrograman berorientasi obyek. ORM berfungsi untuk memetakan entitas menjadi tabel dan atribut menjadi kolom tabel.

3. Perancangan Sistem

3.1 Kebutuhan Hardware dan Software

Agar kinerja sistem aplikasi dapat berjalan secara maksimal diperlukan perlengkapan yang mendukung untuk merancang dan menjalankan sebuah sistem aplikasi. Dalam merancang aplikasi ini, perlengkapan pendukung tersebut adalah perangkat keras dan perangkat lunak.

3.1.1 Dukungan Hardware

Hardware atau perangkat keras merupakan peralatan fisik komputer. Beberapa kebutuhan perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah :

- a. *Laptop* dengan spesifikasi :
 1. Dell Vostro 1014
 2. *Processor Intel Core 2 Duo 2 GHz*
 3. *VGA Mobile Intel 4 Series Express Chipset Family*
 4. *Hard Disk 250 GB` `*
 5. RAM DDR2 4 GB
- b. *Smartphone* dengan spesifikasi :
 1. *Sony Ericsson W8*
 2. *RAM 168 MB*
 3. *Sitem operasi android v2.3 (Gingerbread)*
 4. *External memory 2 GB*

3.1.2 Dukungan Software

Dalam pembuatan aplikasi ini, digunakan *software* atau perangkat lunak sebagai berikut :

1. Sistem Operasi *Microsoft Windows 7*, sebagai sistem operasi yang digunakan komputer.
2. *Jdk-7-windows-i586* sebagai *java compiler* di sistem operasi *personal* komputer berbasis *windows*.
3. *Intellij IDEA* sebagai editor untuk *development* program berbasis *java*.
4. *Android SDK* adalah *emulator android* sebagai *development tool kit android emulator built in intellij IDEA*.

3.2 Perancangan Infrastruktur Jaringan

Pada dasarnya perancangan infrastruktur jaringan komputer tidak terpaku pada salah satu topologi jaringan saja, namun dapat disesuaikan menurut kondisi lapangan dan kebutuhan sistem. Untuk menghubungkan pelanggan dan restoran terdapat satu buah *web server*, pelanggan melakukan transaksi pesan melalui aplikasi android. Kemudian manajer mendapatkan notifikasi pesanan, setelah manajer menerima pesan dari pelanggan maka manajer membuat disposisi pesan kepada *kitchen*. Selanjutnya *kitchen* membuat disposisi kirim ke petugas *delivery*, setelah pesanan diantar pada pelanggan maka petugas *delivery* akan mengirimkan pesan kepada manajer bahwa pesanan telah dikirim.



Gambar 3.1 Perancangan Jaringan

3.3 Perancangan UML (Sistem Layanan Antar Pada Restoran)

Perancangan aplikasi ini menggunakan *UML (Unified Modelling Language)* sebagai pemodelan sistem layanan antar pada restoran. Penulis menggunakan tiga jenis pemodelan sistem pada *UML*, yaitu *activity diagram*, *use case description*, dan *use case diagram*.

3.3.1 Activity Diagram

Pada *activity diagram* sistem layanan antar pada restoran dapat dijelaskan aktifitas-aktifitas yang terjadi dalam suatu proses yang dilakukan oleh pelanggan dan admin. *Activity diagram* sistem layanan antar pada restoran dapat dilihat pada Gambar 3.2.



Gambar 3.2 Activity Diagram Sistem Layanan Antar Pada Restoran

3.3.2 Use Case Description

Pada *use case description* sistem layanan antar pada restoran menjelaskan tentang deskripsi atau keterangan dari pelanggan dan admin. *Use case description* pelanggan dan *use case description* admin dapat dilihat pada Gambar 3.3 dan Gambar 3.4.

| | | |
|---|--------------------------------------|-----------------------------|
| <i>Use Case Name</i> : Data pelanggan | <i>ID</i> : 0000000001 | <i>Importance Level</i> : 1 |
| <i>Primary Actor</i> : Pelanggan | <i>Use Case Type</i> : <i>longer</i> | |
| <i>Stakeholders and Interest</i> : - | | |
| <i>Brief Description</i> : Untuk menyimpan data pelanggan | | |
| <i>Trigger</i> : Identitas pelanggan | | |
| <i>Relationships</i> : include | | |
| <i>Normal Flow Of Events</i> : Input data pelanggan, pesan makanan, input pesanan | | |
| <i>Subflows</i> : lihat menu | | |
| <i>Alternate/Exceptional Flows</i> : - | | |

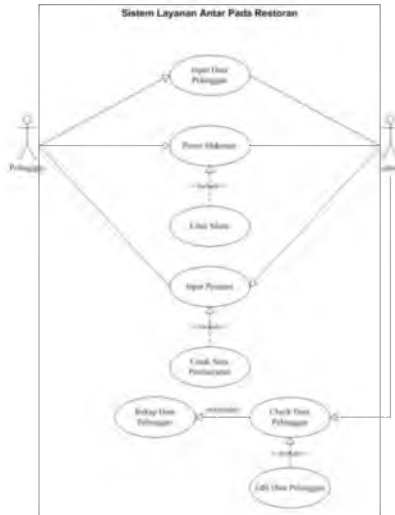
Gambar 3.3 Use Case Description Pelanggan

| | | |
|--|--------------------------------------|-----------------------------|
| <i>Use Case Name</i> : Data admin | <i>ID</i> : 0000000001 | <i>Importance Level</i> : 1 |
| <i>Primary Actor</i> : Admin | <i>Use Case Type</i> : <i>longer</i> | |
| <i>Stakeholders and Interest</i> : - | | |
| <i>Brief Description</i> : Untuk mengelola dan mengolah data pelanggan | | |
| <i>Trigger</i> : Pelanggan | | |
| <i>Relationships</i> : include, extends | | |
| <i>Normal Flow Of Events</i> : Rekap data pelanggan, cek data pelanggan, edit data pelanggan | | |
| <i>Subflows</i> : Cetak nota pembayaran | | |
| <i>Alternate/Exceptional Flows</i> : - | | |

Gambar 3.4 Use Case Description Admin

3.3.3 Use Case Diagram

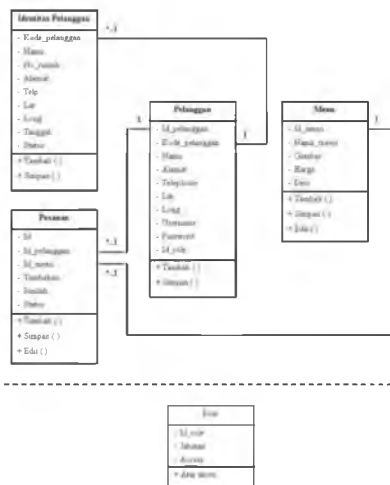
Pada *use case diagram* sistem layanan antar pada restoran dapat dijelaskan kebutuhan dari sisi pengguna (*user*), aktor dari sistem ini adalah Pelanggan dan *Admin*. *Use case diagram* sistem layanan antar pada restoran dapat dilihat pada Gambar 3.5.



Gambar 3.5 Use Case Diagram Sistem Layanan Antar Pada Restoran

3.4 Perancangan ORM (Object Relational Mapping)

Dalam perancangan basis data restoran menggunakan *ORM (Object Relational Mapping)*. *ORM* melakukan pemetaan terhadap tabel-tabel pada basis data relasional dengan suatu *class* yang ada pada bahasa pemrograman berorientasi obyek. *ORM* berfungsi untuk memetakan tabel beserta atribut menjadi kolom. Berikut adalah *ORM* Sistem Layanan Antar Pada Restoran, dapat dilihat pada Gambar 3.6.



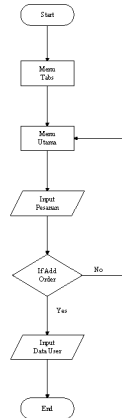
Gambar 3.6 ORM Sistem Layanan Antar Pada Restoran

3.5 Flowchart Sistem

Diagram alir atau *flowchart* adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program yang menyatakan arah alur program tersebut. Dalam diagram alir ini akan diketahui jalur dari program secara keseluruhan. Diagram alir

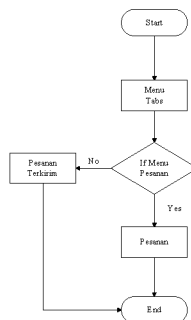
merupakan suatu model logika data yang dibuat untuk menggambarkan darimana asal data dan tujuan data yang keluar sistem, proses apa yang menghasilkan data tersebut. Pada *flowchart* sistem terdapat empat *flowchart*, *flowchart* untuk aplikasi pelanggan restoran, *flowchart* untuk aplikasi *manager* restoran, *flowchart* untuk aplikasi *web kitchen*, *flowchart* untuk aplikasi *delivery* restoran.

3.5.1 Flowchart Aplikasi Pelanggan Restoran



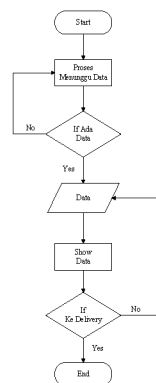
Gambar 3.7 Perancangan *Flowchart* Aplikasi Pelanggan Restoran

3.5.2 Flowchart Aplikasi Manager Restoran

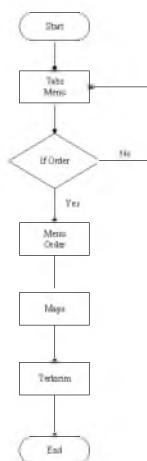


Gambar 3.8 Perancangan *Flowchart* Aplikasi *Manager* Restoran

3.5.3 Flowchart Aplikasi Web Kitchen



Gambar 3.9 Perancangan *Flowchart* Aplikasi *Web Kitchen*



Gambar 3.10 Perancangan *Flowchart* Aplikasi *Delivery*

3.6 Perancangan Antar Muka

Perancangan antar muka merupakan perancangan tampilan muka sebuah aplikasi dan perancangan antar muka ini akan digunakan untuk melakukan interaksi antara pelanggan dan aplikasi itu sendiri, sehingga perancangan antar muka ini harus diusahakan *user friendly* agar pengguna lebih mudah dalam memakai aplikasi. Dalam aplikasi ini terdapat empat buah *interface*, yaitu tampilan untuk aplikasi *user* pelanggan, tampilan untuk aplikasi restoran *manager*, tampilan untuk aplikasi petugas restoran *delivery*, dan tampilan untuk *web kitchen*.

3.6.1 Antar Muka *User Pelanggan*

Antar Muka utama adalah tampilan jendela utama dari aplikasi *user* pelanggan yang menghubungkan antara pelanggan dan restoran. Perancangan aplikasi *user* pelanggan dapat dilihat pada Gambar 3.11 menampilkan menu dari aplikasi, dan menu menu yang terdapat pada antar muka ini adalah sebagai berikut :

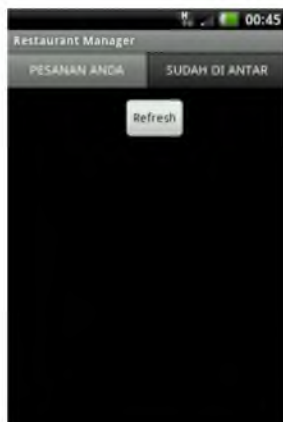
1. Daftar menu, terdapat beberapa menu restoran yang dapat dipilih oleh pelanggan.
2. Pada antar muka pesanan anda, terdapat tampilan menu yang dipilih oleh pelanggan.



Gambar 3.11 Antar Muka *User Pelanggan*

3.6.2 Antar muka *Manager*

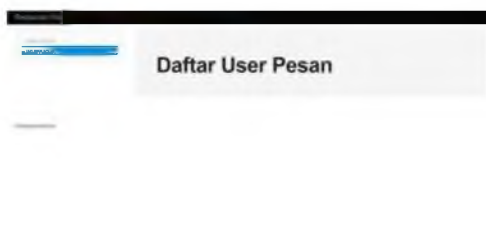
Antar muka utama pada *manager* terdapat beberapa menu untuk melihat pesanan dari pelanggan. Terdapat menu pesanan anda dan menu sudah diantar. Seperti pada Gambar 3.12.



Gambar 3.12 Antar Muka *Manager*

3.6.3 *Form Web Kitchen (Juru Masak)*

Antar Muka utama *web kitchen* hanya terdapat satu menu, yaitu menu *user pesan*. Berfungsi untuk menampilkan daftar menu makanan yang dipesan oleh pelanggan. Dapat dilihat pada Gambar 3.13.



Gambar 3.13 Antar Muka *Web Kitchen*

3.6.4 Antar Muka *Delivery*

Pada antar muka *delivery* akan tampil menu pesanan dari pelanggan yang siap diantar oleh petugas *delivery* kepada pelanggan. Setelah pesanan sampai kepada pelanggan, petugas *delivery* akan melaporkan pesan kepada *manager*. Dapat dilihat pada Gambar 3.14.



Gambar 3.14 Antar Muka *Delivery*

4. Aplikasi Sistem Layanan Antar Pada Restoran

4.1.1 Aplikasi Pelanggan

Aplikasi pelanggan adalah aplikasi yang digunakan sebagai penyedia layanan memesan *order* makanan oleh pelanggan. Untuk menjalankan aplikasi ini, *smartphone* android pelanggan harus terhubung ke jaringan *internet*. Setelah itu mulai mengoperasikan aplikasi pelanggan untuk memesan *order* makanan.

4.1.2 Aplikasi Restoran *Manager*

Aplikasi restoran *manager* adalah aplikasi yang digunakan *manager* untuk menerima *order* pesanan makanan dari pelanggan. Untuk menjalankan aplikasi ini, *smartphone* android *manager* harus terhubung dengan jaringan *internet*. Kemudian mulai mengoperasikan aplikasi *manager* untuk melihat *order* pesanan makanan dari pelanggan.

4.1.3 Aplikasi *Web Kitchen*

Halaman *web* ini berfungsi untuk menampilkan semua *order* pesanan makanan dari pelanggan, yang dikirim oleh *manager* melalui aplikasi restoran *manager*.

4.1.4 Aplikasi Restoran *Delivery*

Aplikasi restoran *delivery* adalah aplikasi yang digunakan petugas *delivery* untuk mengirim *order* pesanan makanan dan mencari lokasi tempat/ rumah *user* melalui *maps* yang ada dalam aplikasi tersebut. Untuk menjalankan aplikasi ini, *smartphone* android yang dibawa oleh petugas *delivery* harus terhubung dengan jaringan *internet*. Setelah itu mulai jalankan aplikasi restoran *delivery* untuk melihat *order* pesanan makanan siap antar dari *kitchen* dan mengirim *order* pesanan tersebut ke pelanggan.

4.2 Uji Aplikasi

4.2.1 Uji Aplikasi Pelanggan

Untuk dapat menjalankan aplikasi pelanggan, pelanggan harus mempunyai aplikasi pelanggan yang disediakan oleh restoran untuk diinstal pada *smartphone* android pelanggan. Adapun cara untuk mendapatkan aplikasi tersebut, pelanggan terlebih dahulu mendaftar dan mengisi formulir identitas biodata diri guna mendapatkan aplikasi pelanggan. Setelah *smartphone* android pelanggan terinstal aplikasi pelanggan, kemudian sambungkan *smartphone* android pelanggan dengan

koneksi *internet*. Berikut gambaran daftar menu makanan, dapat dilihat pada Gambar 4.1.



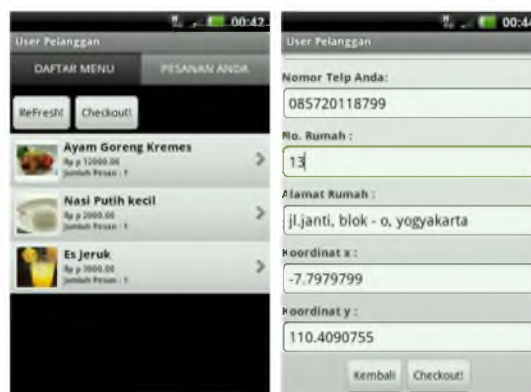
Gambar 4.1 Daftar Menu Makanan

Setelah pelanggan dapat menggunakan aplikasi pelanggan dan melihat daftar menu restoran. Pelanggan dapat melakukan transaksi pesan *order* makanan dengan cara memilih menu makanan yang ingin dipesan. Kemudian pelanggan mengisi jumlah dan keterangan pesanan yang ingin dipesan sebelum mengklik tombol pesan. Gambaran memilih menu makanan dan isi data pesanan oleh pelanggan, dapat dilihat pada Gambar 4.2.



Gambar 4.2 Memilih Menu Makanan dan Isi Data Pesanan Pelanggan

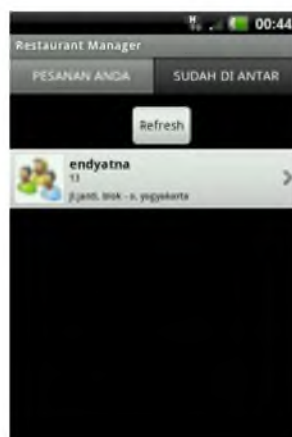
Setelah pelanggan memesan *order* makanan, pelanggan harus mengisi identitas pelanggan untuk menyelesaikan transaksi pesan *order* makanan, jika pelanggan tidak mengisi identitas pelanggan atau kurang lengkap dalam mengisi identitas pelanggan maka pelanggan tidak dapat melakukan transaksi pesan *order* makanan. Gambaran isi data identitas pelanggan, dapat dilihat pada Gambar 4.3.



Gambar 4.3 Isi Data Identitas Pelanggan (*Checkout*)

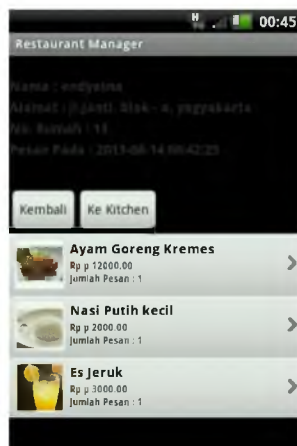
4.2.2 Pengujian Aplikasi Restoran *Manager*

Untuk dapat menjalankan aplikasi restoran *manager* yang digunakan oleh *manager*, *smartphone* android pengguna harus terinstal aplikasi restoran *manager* dan tersambung dengan koneksi *internet*. Kemudian untuk memulai kinerja aplikasi restoran *manager*, pengguna menunggu apakah ada *order* pesan makanan dari pelanggan yang masuk di aplikasi restoran *manager*. Gambaran *order* pesan makanan pelanggan pada aplikasi pengguna, dapat dilihat pada Gambar 4.4.



Gambar 4.4 *Order* Pesan Makanan Pada Aplikasi Restoran *Manager*

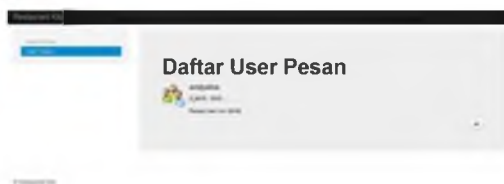
Selanjutnya jika ada *order* pesan makanan dari pelanggan, maka pengguna harus mengolah data pesanan dengan cara menekan atau menyentuh nama pelanggan yang memesan *order* makanan di aplikasi restoran *manager* untuk kemudian mengirim disposisi pesanan ke *kitchen*. Gambaran disposisi pesanan ke *kitchen*, dapat dilihat pada Gambar 4.5.



Gambar 4.5 Disposisi Pesanan Ke Kitchen

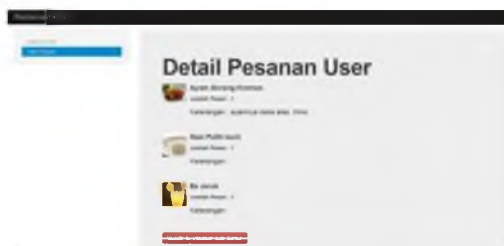
4.2.3 Pengujian Aplikasi Web Kitchen

Aplikasi ini dioperasikan di komputer *kitchen* yang tersambung dengan koneksi *internet*. Pada aplikasi *web kitchen* ini berfungsi untuk menampilkan informasi disposisi pesan yang dikirim oleh *manager*. Gambaran informasi disposisi pesan pada *web kitchen*, dapat dilihat pada Gambar 4.6.



Gambar 4.6 Informasi Disposisi Pesan Pada Web Kitchen

Setelah informasi masuk ke *web kitchen*, untuk melihat detail pesanan yang dikirim oleh *manager*, *kitchen* dapat mengklik tombol yang ada di *web kitchen* dan mulai memasak. Setelah selesai memasak kemudian *kitchen* mengirim disposisi antar makanan ke petugas *delivery*. Gambaran detail makanan dan disposisi antar ke petugas *delivery*, dapat dilihat pada Gambar 4.7.

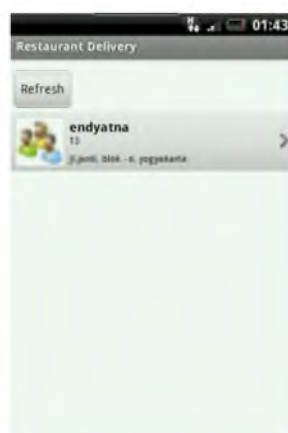


Gambar 4.7 Detail Pesanan dan Disposisi Antar Ke Petugas Delivery

4.2.4 Pengujian Aplikasi Restoran Delivery

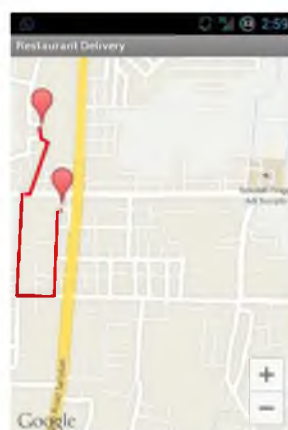
Untuk dapat menjalankan aplikasi restoran *delivery*, *smartphone android* yang digunakan oleh petugas *delivery* harus terinstal aplikasi restoran *delivery* dan

tersambung dengan koneksi *internet*. Aplikasi ini berfungsi untuk menyimpan data disposisi pesan antar yang dikirim oleh *kitchen*. Gambaran data disposisi pesan antar, dapat dilihat pada Gambar 4.8.



Gambar 4.8 Data Disposisi Pesan Antar

Selain menyimpan data disposisi pesan antar dari *kitchen*, aplikasi restoran *delivery* juga berfungsi sebagai alat pandu untuk mengetahui lokasi pelanggan melalui *maps* yang terdapat pada aplikasi restoran *delivery*. Untuk melihat lokasi pelanggan petugas *delivery* dapat menekan atau menyentuh data pesanan pelanggan di aplikasi restoran *delivery*. Gambaran *maps* lokasi pelanggan, dapat dilihat pada Gambar 4.9.



Gambar 4.9 Maps Lokasi Pelanggan

4.3 Hasil Uji

4.3.1 Hasil Uji Smartphone Android

Pengujian pada beberapa tipe *smartphone* android yang memiliki sistem operasi yang berbeda mempunyai tujuan untuk mengetahui apakah *smartphone* android yang diujicoba mendukung apabila diinstal aplikasi pelanggan, aplikasi *manager*, dan aplikasi *delivery*. Tipe *smarthphone* yang diuji antara lain, *Sony Ericsson (W8)*, *Sony Xperia Tipo (ST21i)*, *Samsung Galaxy Ace (S5830)*, dan *Sony Ericsson Xperia (X8)*. Uji *smartphone* android dapat dilihat pada Tabel 4.1.

Tabel 4.1 Tabel Ujicoba *Smartphone* Android

| No | Tipe <i>Smartphone</i> Android | Versi Sistem Operasi | Digunakan Untuk Aplikasi Pelanggan | Digunakan Untuk Aplikasi <i>Manager</i> | Digunakan Untuk Aplikasi <i>Delivery</i> |
|----|-----------------------------------|------------------------------------|------------------------------------|---|--|
| 1 | <i>Sony Ericsson (WE)</i> | V2.3 (<i>Gingerbread</i>) | Mendukung | Mendukung | Tidak Mendukung |
| 2 | <i>Sony Xperia Tipo (ST211)</i> | V4.0 (<i>Ice Cream Sandwich</i>) | Mendukung | Mendukung | Mendukung |
| 3 | <i>Samsung Galaxy Ace (S5830)</i> | V4.1 (<i>Jelly Bean</i>) | Mendukung | Mendukung | Mendukung |
| 4 | <i>Sony Ericsson Xperia (XS)</i> | V2.1 (<i>Eclair</i>) | Tidak Mendukung | Tidak Mendukung | Tidak Mendukung |

Dari hasil ujicoba pada Tabel 4.1 dapat diambil kesimpulan bahwa *smartphone* android yang mendukung untuk diinstal aplikasi pelanggan, aplikasi *manager*, dan aplikasi *delivery* adalah *smartphone* android versi 4.0 (*Ice Cream Sandwich*) keatas. *Smartphone* android versi 2.3 (*Gingerbread*) mendukung apabila diinstal aplikasi pelanggan dan aplikasi *manager*, untuk aplikasi *delivery* tidak mendukung. *Smartphone* android versi 2.1 (*Eclair*) sama sekali tidak mendukung apabila diinstal aplikasi pelanggan, aplikasi *manager*, dan aplikasi *delivery*.

4.3.2 Hasil Uji Responden

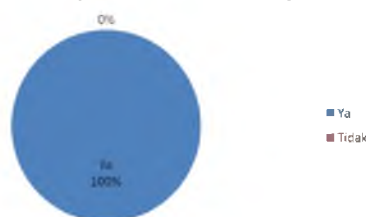
Pengujian ini dimaksudkan untuk mendapatkan data kegunaan aplikasi bagi pelanggan restoran. Data yang dikumpulkan dalam pengujian ini berasal dari 30 *responden*. Data yang diperoleh dari 30 *responden* ini kemudian diolah menjadi data grafik *pie* untuk memudahkan melihat hasil pengujian dari *responden* yang menggunakan aplikasi manajemen layanan restoran berbasis android berupa grafik *pie*.

Adapun pertanyaan fungsi analisis sistem yang diajukan ke *responden* adalah sebagai berikut :

- Apakah anda merasa nyaman menggunakan aplikasi ini ? (Ya/Tidak)
- Apakah efektif dan efisien kalau sistem pelayanan restoran seperti ini benar-benar diterapkan ? (Ya/Tidak)
- Apakah penerapan aplikasi ini dapat meningkatkan efisiensi waktu dalam pemesanan makanan ? (Ya/Tidak)
- Tingkat kepuasan anda atas hasil sistem pelayanan restoran melalui aplikasi layanan antar pada restoran berbasis android ini : (Sangat Puas, Cukup Puas, Kurang Puas, Tidak Puas).

Hasil jawaban responden berupa grafik *pie* adalah sebagai berikut :

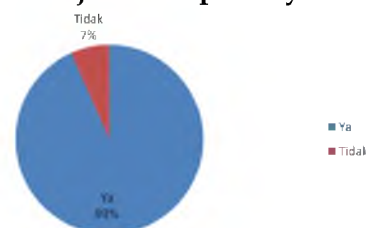
Hasil jawaban pertanyaan a



Gambar 4.10 Grafik Pie Hasil Pertanyaan A

Pada Gambar 4.10 merupakan prosentase atas hasil jawaban dari pertanyaan a, yaitu “Apakah anda merasa nyaman menggunakan aplikasi ini ? (Ya/Tidak)”. Dari 30 *responden* semuanya menjawab ya, sehingga prosentasenya 100%, sedangkan yang menjawab tidak prosentasenya 0%.

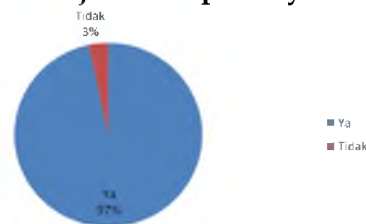
Hasil jawaban pertanyaan b



Gambar 4.11 Grafik Pie Hasil Pertanyaan B

Pada Gambar 4.11 merupakan prosentase atas hasil jawaban dari pertanyaan b, yaitu “Apakah efektif dan efisien kalau sistem pelayanan restoran seperti ini benar-benar diterapkan ? (Ya/Tidak)”. Dari 30 *responden* yang menjawab ya sebanyak 28 *responden* dengan prosentase 93%, sedangkan yang menjawab tidak sebanyak 2 *responden* dengan prosentase 7%.

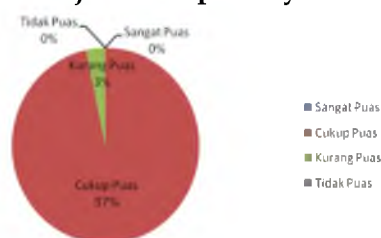
Hasil jawaban pertanyaan c



Gambar 4.12 Grafik Pie Hasil Pertanyaan C

Pada Gambar 4.12 merupakan prosentase atas hasil jawaban dari pertanyaan c, yaitu “Apakah penerapan aplikasi ini dapat meningkatkan efisiensi waktu dalam pemesanan makanan ? (Ya/Tidak)”. Dari 30 *responden* yang menjawab ya sebanyak 29 *responden* dengan prosentase 97%, sedangkan yang menjawab tidak sebanyak 1 *responden* dengan prosentase 3%.

Hasil jawaban pertanyaan d



Gambar 4.13 Grafik *Pie* Hasil Pertanyaan D

Pada Gambar 4.13 merupakan prosentase atas hasil jawaban dari pertanyaan d, yaitu “Tingkat kepuasan anda atas hasil sistem pelayanan restoran melalui aplikasi layanan antar pada restoran berbasis android ini : (Sangat Puas, Cukup Puas, Kurang Puas, Tidak Puas)”. Dari 30 *responden* yang menjawab sangat puas sebanyak 0 *responden* dengan prosentase 0%, yang menjawab cukup puas sebanyak 29 *responden* dengan prosentase 97%, kemudian yang menjawab kurang puas sebanyak 1 *responden* dengan prosentase 3%, sedangkan yang menjawab tidak puas sebanyak 0 *responden* dengan prosentase 0%.

5. Kesimpulan

Dari hasil pada uji coba yang dilakukan dapat ditarik kesimpulan, sistem layanan antar pada restoran berbasis android yang telah diterapkan ini yaitu :

1. Sistem dapat mempermudah melayani pelanggan restoran pengguna *smartphone* android dalam pemesanan makanan *delivery*. Memudahkan petugas *delivery* mencari lokasi alamat pelanggan untuk mengantarkan makanan yang dipesan oleh pelanggan.
2. Hasil uji kegunaan sistem dari responden menggunakan aplikasi sistem layanan antar pada restoran sebesar 100%, untuk tingkat efektif dan efisien menggunakan sistem layanan antar pada restoran sebesar 93%, untuk efisiensi waktu menggunakan sistem layanan antar pada restoran sebesar 97%, untuk tingkat kepuasan responden menggunakan sistem layanan antar pada restoran sebesar 97%.

Daftar Pustaka

- [1] Alan Dennis Dkk, 2005, *System Analysis And Design With UML Version 2.0*, Virginia : Phoenix Color Corp.
- [2] Alqod Elian Dkk, 2012. *Layanan Informasi Kereta Api Menggunakan GPS, Google Maps, dan Android*. Jurnal Teknik Informatika Institut Teknologi Sepuluh November Volume 1, Nomor 1, 2012 : Surabaya.
- [3] Antok Supriyanto , Bre Sukma Kirana, 2008, *Sistem Informasi Manajemen Pemesanan Makanan Cepat Saji Melalui Internet Dengan Multi Outlet*. Proceeding Seminar Nasional Informatika STIKOM ISSN 1979-2328 : Surabaya.
- [4] Awang Hendrianto Pratomo, 2008, *Pengembangan Aplikasi Sistem Pendukung Keputusan Pemilihan Makanan Sehat Berbasis SMS*. Proceeding Seminar Nasional Informatika UPN “Veteran” ISSN 1979-2328 : Yogyakarta.

- [5] Bangun Wijayanto, 2011. *Implementasi Object Relational Mapping (ORM) Menggunakan Hibernate*. Proceeding Seminar Nasional Aplikasi Teknologi Informasi Universitas Jenderal Soedirman ISSN 1907-5022 : Yogyakarta.
- [6] Elihu Setyo Nugroho, 2012. *Manajemen Web Untuk Mendukung Aplikasi Informasi Peta Digital Pada Smartphone Berbasis Android*. *Compiler Journal Adisutjipto Informatics Society* Volume 1, Nomor 2, 2012 : Yogyakarta.
- [7] Fariz Irawan, 2010. *Aplikasi Bergerak Untuk Pemesanan Makanan Di Cupcakes Heaven Via SMS (Short Message Service)*. *Jurnal Fakultas Ilmu Komputer Universitas Mercu Buana* Volume 9, Nomor 1, 2010 : Jakarta.
- [8] Gandhi Pawitan, 2008, *Aplikasi Analisis Pareto Dalam Pengendalian Inventori Bahan Baku Pada Bisnis Restoran*. *Jurnal Jurusan Administrasi Bisnis UNPAR* Volume 4, Nomor 1, 2008 : Bandung.
- [9] Ivan Michael Siregar Dkk, 2010, *Mengembangkan Aplikasi Enterprise Berbasis Android*, Gava Media : Yogyakarta.
- [10] Lily Wulandari, I Wayan Simri Wicaksana, 2006. *Toward Web Service*. Proceeding Seminar Ilmiah Nasional Komputer dan Sitem Intelijen Auditorium Gunadarma ISSN 1411-6286 : Depok.
- [11] Mulyadi, ST, 2010, *Membuat Aplikasi Untuk Android*, Multimedia Center Publishing : Yogyakarta.
- [12] Tjandra Susila Dkk, 2006. *Perancangan Alat Pemesanan Makanan Di Restoran Secara Wireless*. *Jurnal TESLA Teknik Elektro Universitas Tarumanagara* Volume 8, Nomor 2, 2006 : Jakarta.