



Self-Service Technology Berbasis Android Menggunakan RestFul Web Service Pada Bisnis Restoran

Yoseph Halim¹, Sandy Kosasi², Tony Wijaya³, Susanti M. Kuway⁴

^{1,3} Program Studi Teknik Informatika, STMIK Pontianak

^{2,4} Program Sistem Informasi, STMIK Pontianak

¹yosephhalim96@gmail.com, ²sandykosasi@yahoo.co.id, ³mail.tonywijaya@gmail.com, ⁴shanty_stmikptk@yahoo.com

Abstract

The process of ordering food and beverage menus at the Meatzilla restaurant still uses paper and stationery. Constraints that are often encountered are when restaurant visitors are crowded, but waiters are limited, often mistaken of orders and loss and order of paper orders and various other obstacles. This can cause feelings of disappointment and negative impressions which will certainly result in reduced customer satisfaction. This study aims to produce mobile device software that can support the running of food and beverage ordering systems at Meatzilla Restaurant self-service so that orders arrive quickly and right desires. This system is made using XML and java programming language with android studio tools version 3.5.3. Research in the form of case studies, software design methods using extreme programming models and system modeling tools using Unified Modeling Language (UML). This system is connected to the internet network where the admin, cashier, kitchen and customers are integrated with a restful web service. The final result of this study is a food and beverage ordering system consisting of 4 (four) parts, namely customer and admin software that is accessed by using a smartphone and the cashier and kitchen which is accessed by a PC / laptop. this system is very helpful in facilitating the management of customer orders.

Keywords: Self-Service Technology, Smartphone, Meatzilla Restaurant, Restful, XP, UML.

Abstrak

Proses pemesanan menu makanan dan minuman pada restoran Meatzilla, masih menggunakan media kertas dan alat tulis. Kendala yang sering dihadapi adalah ketika pengunjung restoran sedang ramai, namun pelayan (waiters) terbatas, sering kelirunya urutan pesanan serta kehilangan dan tidak urutnya lembaran kertas pesanan dan berbagai kendala lainnya. Hal ini dapat menyebabkan rasa kecewa serta kesan negatif yang tentunya akan mengakibatkan berkurangnya kepuasan pelanggan. Penelitian ini bertujuan untuk menghasilkan software perangkat mobile yang dapat mendukung dalam menjalankan sistem pemesanan makanan dan minuman pada Restoran Meatzilla berorientasi self-service sehingga pesanan pelanggan datang dengan cepat, dalam jumlah yang tepat dan sesuai keinginan. Sistem ini dibuat dengan menggunakan bahasa pemrograman XML dan java dengan tools android studio versi 3.5.3. Penelitian berbentuk studi kasus, metode perancangan perangkat lunak menggunakan model extreme programming (XP) dan alat pemodelan sistem menggunakan Unified Modeling Language (UML). Sistem ini terkoneksi dalam jaringan internet dimana bagian admin, kasir, dapur dan pelanggan saling terintegrasi dengan restful web service. Hasil akhir dari penelitian ini adalah sistem pemesanan makanan dan minuman terdiri dari 4 (empat) bagian yaitu software pelanggan dan admin yang diakses dengan menggunakan smartphone dan bagian kasir serta dapur yang diakses dengan pc/laptop. Pengaplikasian sistem ini sangat membantu dalam mempermudah pengelolaan pesanan pelanggan dimana pelanggan dapat menginputkan sendiri pesanan yang diinginkan

Kata kunci: Self-Service Technology, Smartphone, Restoran Meatzilla, Restful, XP, UML.

1. Pendahuluan

Di era revolusi industri 4.0 saat ini, teknologi memainkan peran penting dalam kehidupan manusia [1]. Hal ini terjadi karena teknologi menawarkan kemudahan dalam membantu menyelesaikan setiap pekerjaan manusia. Peran teknologi informasi dan komunikasi menjadi sangat penting karena banyaknya tuntutan kebutuhan akan pertukaran informasi yang cepat dan tepat [2]. Salah satu bidang bisnis yang membutuhkan kecepatan dan ketepatan adalah bisnis restoran. Bisnis

restoran yang mampu menangani kebutuhan konsumen secara cepat, tepat dan akurat dapat menjadi tolak ukur keberhasilan suatu restoran [3].

Saat ini banyak restoran yang berhasil menarik pelanggan, tetapi banyak pula yang gagal. Faktor penyebabnya selain cita rasa dan harga dari menu yang disajikan, yang tidak kalah penting adalah pelayanan (*servis*) yang diberikan. Pelayanan kepada konsumen terutama pada bidang jasa makanan di restoran telah banyak diteliti oleh beberapa penelitian terdahulu.



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

Kualitas layanan dalam hal *service* / pelayanan restoran memiliki pengaruh yang signifikan serta positif terhadap kepuasan pelanggan [3]. Kualitas layanan terus menjadi topik penelitian yang luas dan sekarang bahkan telah muncul dalam bentuk *self-service technology* yang memiliki efek mendalam pada cara konsumen berinteraksi dengan perusahaan untuk menciptakan hasil layanan positif terhadap konsumen [4].

Proses pemesanan menu pada restoran Meatzilla sendiri, masih menggunakan metode pemesanan menu dengan media kertas dan alat tulis. Terdapat beberapa kendala yang dihadapi adalah jarak antara dapur dan konsumen yang jauh, ketika pengunjung restoran sedang ramai, namun pelayan (*waiters*) terbatas, kasir harus menginput manual pesanan pelanggan, sulit merubah daftar menu dan harga, sering kelirunya urutan pesanan. Sedangkan permasalahan yang dihadapi pemilik dan manajer adalah sulit memantau transaksi berjalan jika tidak sedang berada di lokasi restoran, sulit menentukan bahan-bahan masakan yang harus dibeli atau kurang, pemilik restoran sulit memantau penghasilan harian, serta waktu-waktu kapan saja restoran dalam keadaan ramai pengunjung sehingga persiapan mengantisipasi permasalahan-permasalahan yang timbul saat ramai pengunjung menjadi kurang.

Dari berbagai permasalahan yang ada tentunya akan mengakibatkan berkurangnya kepuasan pelanggan. Berdasarkan permasalahan yang ada, maka diperlukan sebuah sistem pemesanan yang dapat menunjang proses bisnis pada restoran meatzilla. Sistem pemesanan dengan teknologi *self-service* berbasis *smartphone* android dirasa cocok dimana banyak kemudahan yang dapat diberikan dalam memesan menu makanan melalui dukungan teknologi *mobile devices*, salah satunya adalah meminimalisasi kekeliruan dan kerangkapan pemesanan, kesalahan perhitungan jumlah pembayaran, dan mengurangi waktu tunggu (antrian), selain itu fleksibilitas juga menjadi alasan dalam memilih *platform mobile* dalam mengembangkan sistem pemesanan makanan [5].

Sedangkan penggunaan *self-service technologies* terbukti sangat membantu dalam penyediaan layanan yang nyaman bagi konsumen dan menjadi sangat penting terutama dalam mencapai produktivitas dan kepuasan [6]. *Self-service technology* dapat mengganti sentuhan manusia dalam proses pemesanan makanan karena melalui *self-service technology* konsumen melayani diri sendiri dengan antarmuka teknologi [7]. Teknologi *self-service* telah dipergunakan di restoran-restoran di Indonesia dan berdasarkan hasil penelitian menunjukkan bahwa *self-service technology* mudah digunakan dan dapat memberikan kesenangan bagi penggunanya [8]. Untuk memadukan teknologi *self-service* dan *smartphone* berbasis android agar menghasilkan *self-service system* diperlukan *web service* sebagai perantaranya. Fungsi dari *web service* yaitu untuk peningkatan jumlah integrasi dan

fleksibilitas untuk pengembangan demi kepentingan dalam mengintegrasikan *service* ke dalam proses bisnis yang ada [9].

Jenis *web service* yang digunakan adalah Restful. Alasan penggunaan restful adalah restful sangat baik dalam mengoptimalkan kinerja. Ketika server restful diakses dalam skala besar dan secara bersamaan dalam sistem terdistribusi di internet, karena memiliki kinerja yang tinggi dan stabil [10]. Dengan mengimplementasikan restful *web service* dalam *service oriented architecture* (SOA) akan memudahkan pengembangan aplikasi perangkat lunak di luar sistem maupun dengan bahasa pemrograman atau *platform* berbeda [11]. Penelitian terdahulu yang relevan adalah sistem pemesanan makanan dan minuman untuk bisnis restoran yang membahas tentang Perancangan Prototipe Sistem Pemesanan Makanan dan Minuman Menggunakan *Mobile Device*.

Sistem ini memiliki model arsitektur dengan fokus kepada kebutuhan pengguna menjadikan sistem pemesanan menjadi lebih efisien dan efektif serta meningkatkan hubungan personalisasi pengunjung/pelanggan restoran secara online [12]. Penelitian terdahulu yang juga relevan adalah *Self-Service Technology* Berbasis *Smartphone Device* pada Sistem Pemesanan Menu Makanan. Sistem yang dihasilkan adalah bentuk layanan konsumen berbasis *mobile device* yang memungkinkan konsumen melayani dirinya sendiri. Sistem *self-service technology* yang telah diterapkan bukan hanya sebatas memberikan kepuasan konsumen sebagai penggunanya, akan tetapi dapat meningkatkan keuntungan bagi pemilik bisnis [13].

2. Metode Penelitian

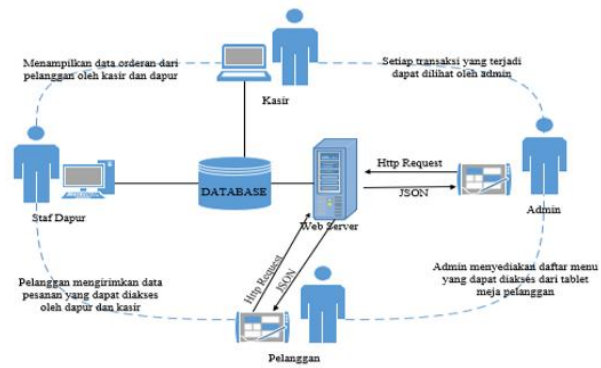
Penelitian ini menggunakan bentuk studi kasus dengan metode penelitian *agile method* dan metode perancangan *extreme programming*. Perancangan sistem *self-service technology* menggunakan perangkat *mobile devices / smartphone*. Metode *extreme programming (XP)* terdiri dari *planning, design, coding* dan *testing*. *Extreme programming* merupakan salah satu metodologi dalam rekayasa perangkat lunak dan juga merupakan satu dari beberapa *agile software development methodologies* yang berfokus pada coding sebagai aktivitas utama di semua tahap pada siklus pengembangan perangkat lunak. *Extreme Programming (XP)* dikenal dengan metode atau "*technical how to*" bagaimana suatu tim teknis mengembangkan perangkat lunak secara efisien melalui berbagai prinsip dan teknik praktis pengembangan perangkat lunak. XP menjadi dasar bagaimana tim bekerja sehari-hari. [14]. Pemilihan metode perancangan *extreme programming* dikarenakan dalam membangun *self-service technology* dimana *user / pengguna* aplikasi memiliki peran penting dalam memberi masukan untuk aplikasi yang akan di bangun serta metode perancangan *extreme programming* cocok

untuk proyek skala kecil hingga menengah mengingat sistem ini hanya untuk internal restoran. Pemodelan sistem menggunakan UML (*Unified Modeling Language*). Instrumen penelitian melalui teknik wawancara dan observasi. Variabel bebas dalam penelitian ini adalah *self-service technology* pada restoran Meatzilla, sedangkan variabel terikat dalam penelitian ini adalah berbasis android menggunakan *Restful web service*. Metode pengujian perangkat lunak yang digunakan adalah *black-box testing*. Kelebihan *black-box testing* adalah dapat meminimalisir fungsi yang tidak benar atau tidak ada, kesalahan antarmuka (*interface errors*), kesalahan pada struktur data dan akses basis data, *black-box testing* menguji spesifikasi fungsional tanpa menguji kode program [15].

3. Hasil dan Pembahasan

Dalam perancangan aplikasi ini terdapat empat bagian pengguna yaitu admin yang melakukan pengelolaan data menu dan karyawan, pelanggan melakukan pencatatan data pesanan dengan menggunakan perangkat *mobile* /tablet, staf dapur menyediakan dan mempersiapkan pesanan pelanggan sesuai dengan data yang kirim oleh pelanggan dari perangkat *mobile*, sedangkan kasir melakukan penagihan terhadap pesanan pelanggan sesuai dengan data yang pesanan yang telah diolah staf dapur. Dalam merancang aplikasi ini digunakan metode perancangan *extreme programming*. Berdasarkan tahapan-tahapan yang ada pada metode perancangan *extreme programming*, maka penulis membangun aplikasi secara bertahap mengacu pada langkah-langkah metode perancangan tersebut. Diawali dengan melakukan perencanaan / *planning* berupa identifikasi kebutuhan pengguna (*user requirements*) dan Analisis kebutuhan sistem (*system requirements*).

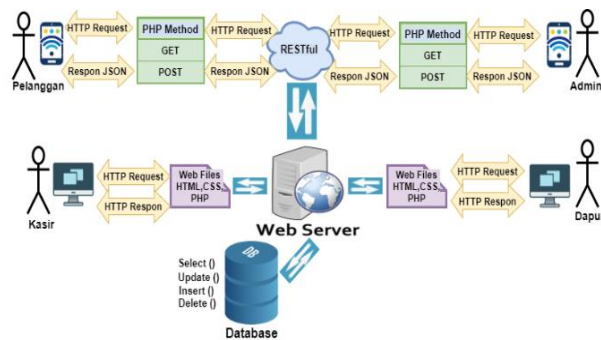
Tahap identifikasi kebutuhan pengguna dilakukan dengan cara observasi, wawancara dan studi dokumentasi seperti yang telah dijelaskan pada teknik pengumpulan data. Analisis kebutuhan sistem dilakukan dengan cara mempelajari masalah-masalah yang terjadi pada Restoran Meatzilla. Kemudian membangun sebuah sistem berbasis *android* dan *web-based* untuk membantu menyelesaikan masalah-masalah yang ada. Tahap selanjutnya adalah *design* / rancangan sistem. Perancangan sistem dilakukan dengan menggambarkan arsitektur sistem dan dimodelkan dalam bentuk diagram-diagram UML. Rancangan sistem untuk aplikasi dijelaskan melalui *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity-relationship diagram*. Perancangan arsitektur sistem ini dimaksudkan untuk memberikan gambaran dari kebutuhan *hardware* dan pengguna yang terlibat serta model dari arsitektur yang mendukung dalam penerapan sistem *self-service technology* pada restoran meatzilla secara lengkap dan terperinci



Gambar 1 Arsitektur Sistem

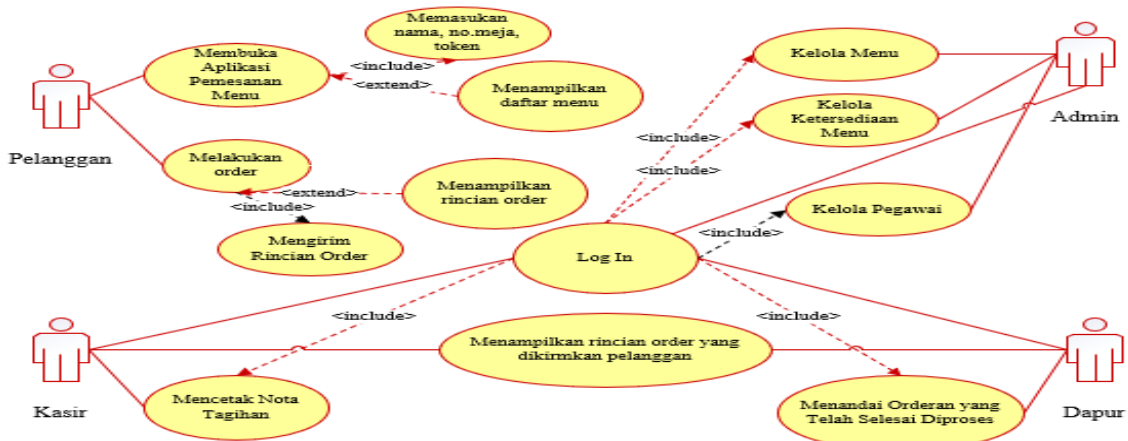
Gambar 1 merupakan arsitektur pengambilan data menggunakan *RESTful web service* oleh site, pelanggan dan sistem kasir. Bila pelanggan dan kasir akan melakukan permintaan / *request* ke server, maka data tersebut akan dikirim ke *web server* yang dibangun. *REST API* akan melakukan proses permintaan ke *database* yang telah dihosting online menggunakan *web server 000webhost* yang disediakan oleh hostinger.

Kemudian hasil dari proses permintaan / *request* akan dikembalikan lagi ke pengguna dalam bentuk *JSON* dan diuraikan oleh perangkat lunak sehingga data hasil proses bisa digunakan dan ditampilkan. Untuk kasir dan dapur akan langsung terhubung ke database tanpa harus melalui *RESTful web service* / *REST API*. Bagian pelanggan dan admin menggunakan perangkat *mobile* sedangkan kasir dan dapur menggunakan perangkat *computer/pc*. Semua perangkat harus terhubung dengan jaringan internet.



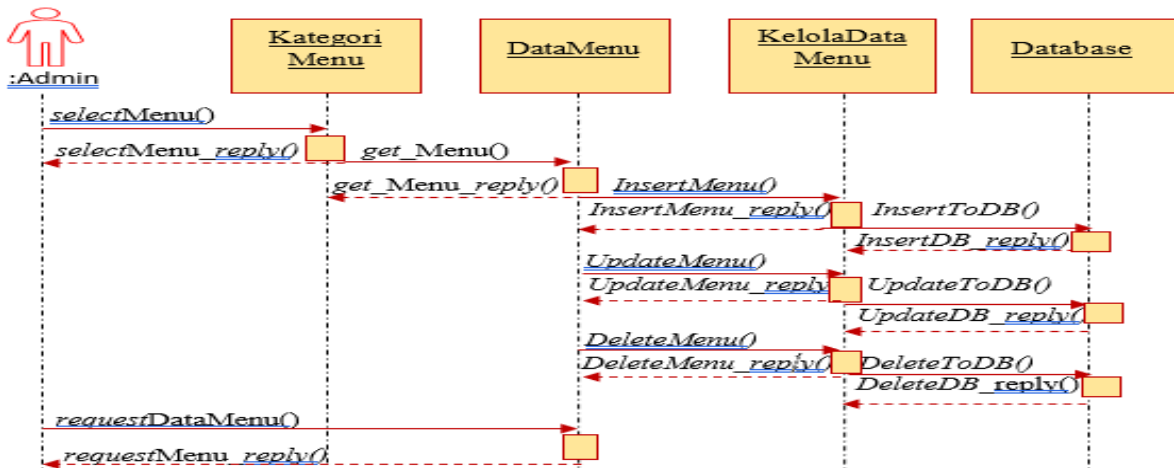
Gambar 2 Skema Sistem

Gambar 2 merupakan gambaran umum dari sistem yang di bangun. Berikut adalah detail dari tiap proses terkait skenario yang dilakukan, file java pada android yang terlibat, file php, url restful pada web server yang digunakan, serta tabel database yang digunakan. Semua bagian akan terhubung kedalam web server yang sama. Penjelasan detail terdapat pada tabel 1 dan tabel 2.



Gambar 3 Use Case Self-Service Restoran Meatzilla

Gambar 3 menjelaskan *use case* sistem, dimana mengelola pembayaran dan menandai pesanan yang pelanggan menginputkan nama, no. meja dan token telah dibayar. Admin login untuk mengelola menu dan untuk mencatat semua pemesanan makanan dan minuman, dapur login untuk mengelola pesanan dan menandai pesanan yang telah selesai. Kasir login untuk *system* menggunakan *mobile devices*.



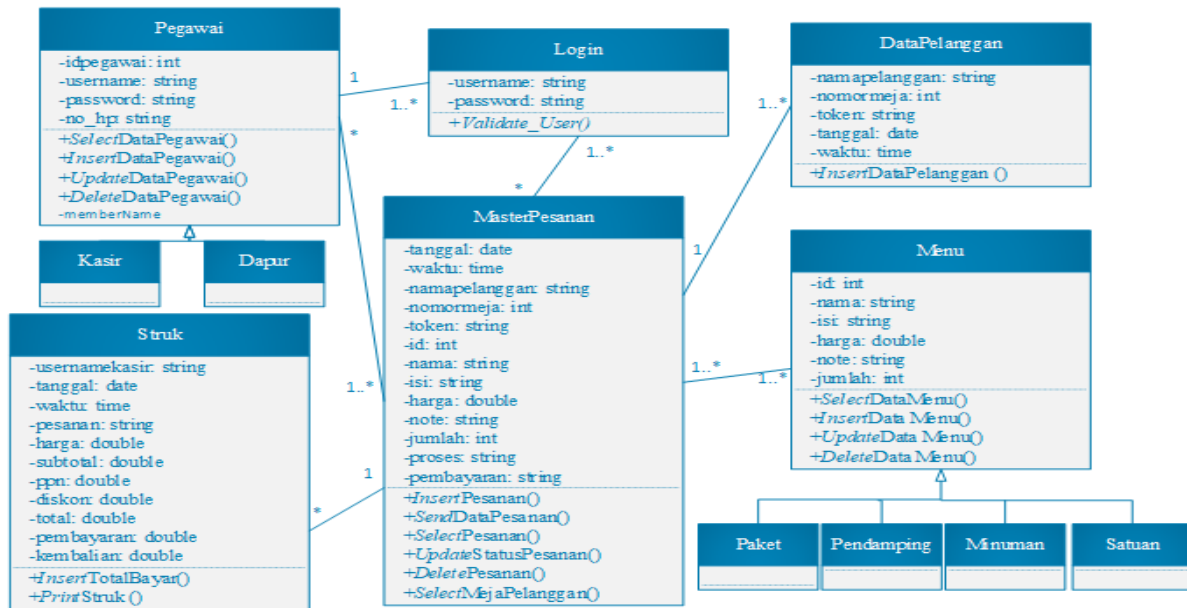
Gambar 4 Sequence Kelola Menu Oleh Admin

Gambar 4 menjelaskan tentang *sequence* sistem untuk menggambarkan interaksi antar objek di dalam sistem yaitu admin dan perangkat lunak berupa proses kelola menu yang dilakukan oleh admin. Pertama admin masuk ke halaman menu, kemudian akan tampil data menu yang hendak dikelola apakah akan diubah, dihapus maupun ditambah. Kemudian akan masuk ke halaman kelola menu dan ketika admin menekan tombol simpan maka data menu baru akan tersimpan ke *database* dan akan tertampil pada data menu dengan tampilan terbaru.

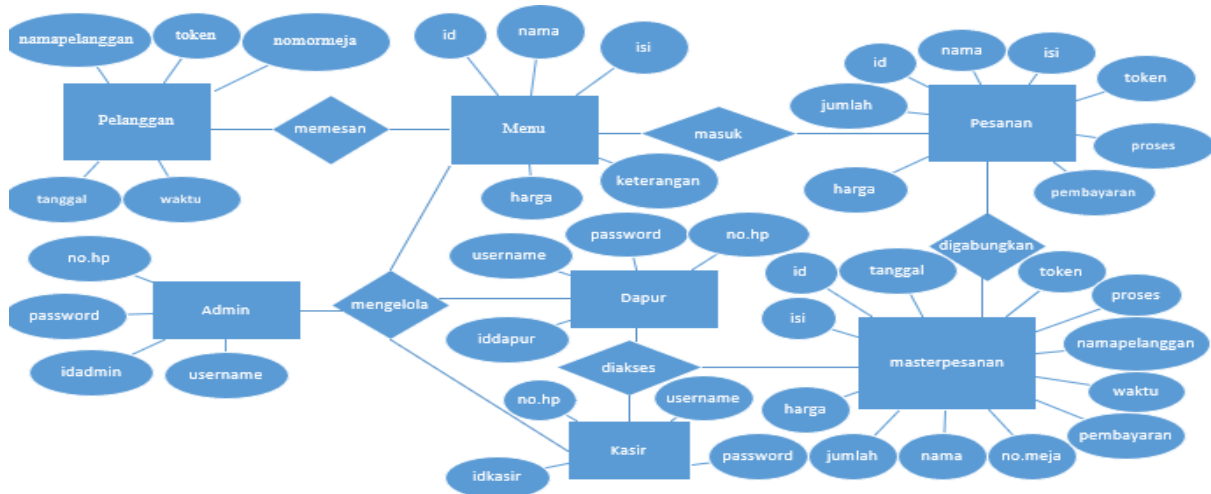
Sequence admin memiliki beberapa *method* dimana semua *method* ini merupakan *method* yang akan di jalankan pada *back end* aplikasi. Beberapa *method* tersebut antara lain *selectMenu()* adalah untuk aplikasi android melakukan request data menu dari database yang sebelum itu dikonversi dalam bentuk JSON. *get_menu()* adalah *method* untuk menentukan kategori

menu yang akan dipilih, sedangkan *insertMenu()* adalah untuk memasukan menu apa saja yang akan di pesan. *updatemenu()* dan *deletemenu()* masing-masing fungsinya adalah untuk memperbaharui dan menghapus menu.

Gambar 5 menjelaskan *class diagram* ini terdiri dari 6 (enam) *class* yang masing-masing entitas saling ketergantungan antara yang satu dengan yang lainnya. Sebuah pesanan dapat terdiri dari banyak pelanggan. Pelanggan dapat memesan satu atau banyak pesanan. Sebuah master pesanan terdiri dari sebuah pesanan. Sebuah pesanan terdiri dari banyak menu. Sebuah pesanan terdiri dari satu atau banyak menu. Sebuah menu terdiri dari satu atau sebaliknya. Dan menu terdiri dari empat buah sub menu. Admin dapat menambahkan satu atau banyak menu, kasir dan staf dapur sebaliknya menu kasir dan staf dapur dikelola satu admin. Kasir dan dapur dapat menampilkan satu atau banyak pesanan



Gambar 5 Class Diagram Self-Service System Restoran



Gambar 6 ERD Self-Service System Restoran

Gambar 6 merupakan *entity relationship diagram* yang menjelaskan hubungan antar entitas yang ada dalam *self-service system* pada restoran meatzilla. Ketika pelanggan memesan menu, akan masuk ke dalam entitas pesanan dan akan digabungkan antara entitas pelanggan dan entitas pesanan kedalam entitas masterpesanan yang memuat data gabungan dari kedua entitas tersebut. Kemudian entitas master pesanan dapat diakses oleh entitas dapur dan kasir untuk melihat pesanan yang masuk dan untuk melakukan transaksi. Sedangkan entitas admin bertugas mengelola entitas menu, kasir dan dapur. Setiap entitas memiliki atributnya. Tahap ketiga adalah hasil rancangan sistem. Dalam perancangan sistem ini, terdapat dua buah bagian utama, yaitu rancangan perangkat lunak bagian *front end* serta rancangan perangkat lunak bagian *back end*. Bagian

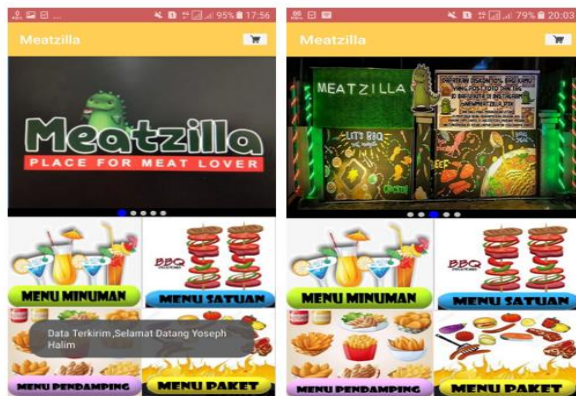
front end berupa bagian di mana *end user* atau pengguna akhir dalam hal ini admin, pelanggan, kasir dan staf dapur restoran akan berinteraksi sepanjang waktu. Bagian ini pada dasarnya adalah bagian antarmuka pengguna karena menyajikan elemen dari apa yang dimiliki perangkat lunak atau sistem yang akan disajikan dan terlihat oleh pengguna.

Bagian *back end* ini terdiri dari *database* yang mengontrol perangkat lunak dan tidak ditampilkan kepada pengguna akhir. Perubahan pada bagian ini akan mempengaruhi tampilan dan fungsi keseluruhan dari perangkat lunak. Bagian ini adalah bagian yang mendukung tindakan seperti pengguna akhir mengklik tombol, meminta data atau informasi, atau mengeluarkan perintah untuk eksekusi program, merespon aksi atau perintah dari pengguna akhir.



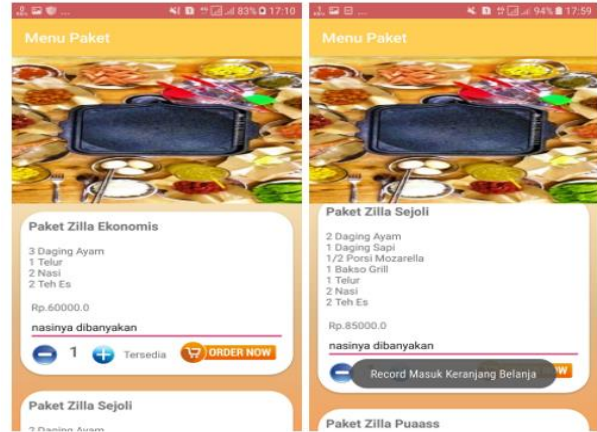
Gambar 7 Tampilan Input Data Pelanggan

Gambar 7 pada tampilan *input* data pelanggan, pelanggan diminta untuk memasukkan data berupa nama pelanggan, nomor meja sesuai tempat duduk yang dipakai oleh pelanggan, dikarenakan sistem yang dibangun memerlukan sebuah *device* pada setiap mejanya maka nomor meja akan otomatis terdaftar pada sistem dan token yang berupa kode unik yang dapat dimasukan secara bebas oleh pelanggan berupa kombinasi huruf dan angka (kode unik dan nomor meja akan terisi otomatis dari sistem yang di bangun), kemudian terdapat sebuah *button* kirim untuk mengirim data tersebut kedalam tabel pelanggan pada *database*. Kode unik / token yang ada pada layout ini akan menjadi penghubung antara data pelanggan serta pesanan dari pelanggan agar tidak tertukar pesanan antar pelanggan yang satu dengan pelanggan lainnya.



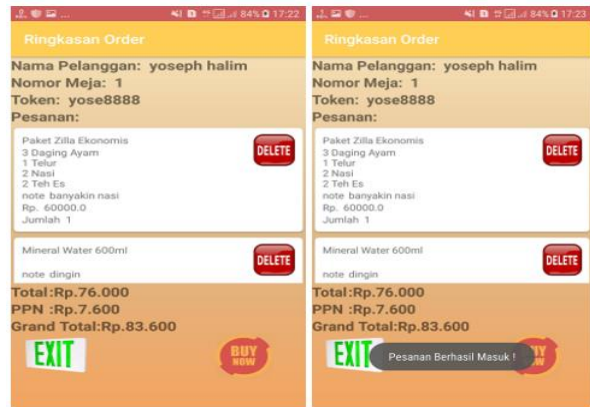
Gambar 8 Tampilan Dashboard Menu

Gambar 8 pada tampilan menu terdapat empat pilihan kategori menu yaitu menu minuman, menu satuan, menu pendamping dan menu paket. Masing-masing pilihan menu tersebut akan menampilkan isi menu pada restoran Meatzilla. Pada bagian atas pilihan menu terdapat *image slider* yang berfungsi untuk menampilkan gambar-gambar yang terdiri dari lima buah gambar yang dapat bergerak atau bergeser secara otomatis.



Gambar 9 Tampilan Menu Paket

Gambar 9 pada tampilan menu paket menggunakan *recycler view* untuk menampilkan pilihan menu paket yang dapat di *scroll* ke bawah. Kemudian data menu ditampung dalam *card view* yang memuat nama paket, isi dari paket, harga, serta terdapat sebuah *edit text* untuk menambahkan *note/catatan* untuk setiap *item* pesanan. Selain itu terdapat tiga buah *button* yaitu *button minus* untuk mengurangi peanan, pelanggan tidak dapat mengurangi pesanan hingga di bawah 1, *button plus* untuk menambah pesanan, keterangan ketersediaan menu yang berfungsi jika menu tidak tersedia maka pelanggan tidak dapat memesan menu tersebut, serta *button order* untuk mengirim data pesanan ke keranjang belanja.



Gambar 10 Tampilan Ringkasan Order

Gambar 10 pada tampilan rincian order terdapat *textview* berupa nama pelanggan, nomor meja, token, total, ppn, grand total, serta sebuah *scroll view* yang dapat di *scroll* yang memuat isi pesanan untuk menampilkan data pesanan yang di inputkan pelanggan secara terperinci memuat nama pesanan, *note*, jumlah pesanan, dan harga per *item* menu dan pelanggan dapat menghapus tiap item pesanan serta dapat *me-refresh* layout untuk mengupdate total, ppn dan grand total. Kemudian di pojok kanan bawah terdapat tombol *buy now* dapat digunakan untuk mengirim data pesanan ke *database* dan tombol *exit* untuk keluar jika telah selesai memesan.

Token	Tanggal	Waktu	Nama Pelanggan	Nomor Meja	Nama Pesanan	Isi Pesan	Note	Jumlah	Status Button	Status Proses
you81	2020-03-17	18:57:15	Saiful	5	Paket Zilla Pausan Pausan Rice (Nasi Putih)	2 Daging Ayam 2 Daging Sapi 1 2 Paus Meatzilla 1 French Fries 1 Bako Grill 3 Nasi 1 Zilla Rice 2 Paus Rice 1 Teh Es	masi pando	1	Cancel	selesai
you8	2020-03-17	18:51:33	Wijaya	4	Masakan Sate 600ml Sate Baku (Sate Kacang)		gag baw	2	Cancel	selesai
you83	2020-03-17	18:11:01	Karey	3	Paket Zilla Pausan Shrimp(oleng) Aronan(Yellow) Chawanmushi(Teh Kacang) Egg(Telar Kanyang) Bawal(Angi) Ippu(spedal)	2 Daging Ayam 2 Daging Sapi 1 2 Paus Meatzilla 1 French Fries 1 Bako Grill 3 Nasi 1 Zilla Rice 2 Paus Rice 1 Teh Es	tidak wa	1	Cancel	selesai
you82	2020-03-17	18:07:29	David	2	Masakan Sate 600ml Paus Rice (Nasi Putih)		masi pando	2	Cancel	selesai
you88	2020-03-17	18:01:29	Yoseph Halim	1	Paket Zilla Ekonomis Masakan 300ml Bako (Lumer Endok)	3 Daging Ayam 3 Telur 2 Nasi 2 Teh Es	masi pando	1	Cancel	selesai

Gambar 11 Halaman Rincian Order Dapur Meatzilla

Gambar 11 merupakan halaman rincian order pelanggan yang masuk ke dapur meatzilla. Halaman ini akan tampil setelah staf dapur melakukan login terlebih dahulu. Pada halaman ini staf dapur dapat melihat tampilan rincian order berupa token, tanggal, waktu, nama pelanggan, nomor meja, nama pesanan, isi pesan, note dan jumlah dari pesanan pelanggan itu sendiri. Kemudian terdapat *button* untuk menghapus yang berfungsi menghapus pesanan yang di batalkan pelanggan dikarenakan jika pelanggan telah mengirimkan pesanan ke database maka tidak dapat dibatalkan. Juga dapat menandai pesanan yang telah selesai dengan *button* selesai

Nama Pesanan	Harga	Jumlah	Sub Total
Paket Zilla Ekonomis	10000	1	10000
Masakan 300ml	5000	1	5000
Endok (Lumer Endok)	10000	1	10000

Total: Rp. 75000
 Diskon: 15%
 PPN: Rp. 7500
 Grand Total: Rp. 71250

Gambar 12 Halaman Tampilan Kasir Meatzilla

Gambar 12 merupakan halaman tampilan kasir. Halaman ini akan muncul setelah kasir memilih dan mengklik meja mana yang hendak melakukan transaksi pada halaman grid sebelumnya. Halaman ini akan menampilkan token, no. meja, nama pelanggan, nama kasir, waktu, tanggal, pesanan apa saja yang dipesan oleh pelanggan, total harga, diskon (jika ada), ppn, dan grand total. Kemudian kasir hanya perlu menginputkan total pembayaran yang dilakukan oleh pelanggan pada kolom pembayaran dan akan muncul kembalian (jika ada) pada kolom kembalian. Setelah itu kasir perlu menekan tombol *checkout* untuk mencetak struk kepada pelanggan.

Berikut ini merupakan beberapa contoh *rest api / restful web service* yang digunakan dalam penelitian ini yang ditulis menggunakan bahasa pemrograman php (*Hypertext Preprocessor*) versi 7.3.12. Terdapat 35 *restful web service* yang digunakan untuk mengelola masing-masing fungsi yang berjalan pada sistem yang dibangun, namun akan ditampilkan beberapa contoh saja.

```

/public_html/login_admin.php
<?php
require 'conn.php';
$username = $_GET['username'];
$password = $_GET['password'];
$sql_get_admin = "SELECT * FROM admin where
username='$username' AND password='$password'";
$query= $conn->query($sql_get_admin);
$result= mysqli_num_rows($query);
$idCount = null;
if($result==1){
    $idCount[] = (array("idCount"=>"1"));
}
else{
    $idCount[] = (array("idCount"=>"0"));
}
header('Content-Type: application/json');
$response = $idCount;
echo json_encode($response);?>
    
```

File yang memuat *script* ini berfungsi untuk mengecek kecocokan antara username dan password yang diinputkan admin dan data yang ada pada *database*. Kemudian menggunakan *php array* untuk hasil *result*, jika data sama maka *idCount1* dan jika berbeda maka *idCount0*. Ditampilkan menggunakan *json_encode*.

```

/public_html/tambah_masterpesanan.php
<?php
require 'conn.php';
$token = $_GET['token'];
mysql_query($conn,"insert into masterpesanan
select * from pelanggan join pesanan
using(token) where token=$token");
$sql_get_masterpesanan = "SELECT * FROM
masterpesanan";
$query= $conn->query($sql_get_masterpesanan);
$response_data = null;
while ($data = $query->fetch_assoc()) {
    $response_data[] = $data;
}
if (is_null($response_data)) {
    $status = true;
}
header('Content-Type: application/json');
$response = $response_data;
echo json_encode($response);
?>
    
```

Pada file *tambah_masterpesanan.php* memuat *script* untuk memasukkan data kedalam tabel *masterpesanan* dalam *database*. Data tersebut diambil dari kedua tabel yaitu tabel *pelanggan* dan tabel *pesanan* menggunakan *foreign key* yaitu *token*. Penggunaan *foreign key* *token* untuk menghubungkan kedua tabel yakni tabel *pelanggan* dan *pesanan* sehingga dapat disatukan menjadi sebuah tabel yaitu tabel *masterpesanan* berdasarkan *token* yang sama.

```

/public_html/get_transaksi.php
<?php
require 'conn.php';
$tanggal1 = $_GET['tanggal1'];
$tanggal2 = $_GET['tanggal2'];
    
```

```
$sql_get_masterpesanan = "SELECT * FROM
masterpesanan WHERE tanggal BETWEEN '$tanggal'
AND '$tanggal2'";
$query= $conn->query($sql_get_masterpesanan);
$response_data = null;
while ($data = $query->fetch_assoc()) {
    $response_data[] = $data;
}
if (is_null($response_data)) {
    $status = true;}
header('Content-Type: application/json');
$response = $response_data;
echo json_encode($response);?>
```

File ini memuat script untuk menerima inputan dari android dan mengambil inputan dengan metode php GET. Data yang diambil adalah merupakan dua buah tanggal. File ini berfungsi untuk melihat transaksi yang terjadi sesuai dengan rentang tanggal transaksi yang di inputkan oleh user. Kemudian data yang di dapat dari database di parsing kedalam bentuk JSON menggunakan json_encode agar dapat ditampilkan ke dalam aplikasi android.

Berikut adalah detail dari tiap proses terkait skenario yang dilakukan, file java pada android yang terlibat, file php yang digunakan, url restful pada web server yang digunakan, serta tabel database yang digunakan, namun hanya akan ditampilkan beberapa contoh saja.

Tabel 1 Proses Sistem Pelanggan

Skenario	Java (Android)	PHP	Restful	Tabel Database
Input nama, no.meja, token (data pelanggan)	Pelanggan.java	tambah_pelanggan.php	http://000webhost.com/tambah_pelanggan.php	pelanggan
Request menu oleh aplikasi	MenuPaket.java, MenuSatuan.java, Minuman.java, MenuPemandangan.java	get_menu_paket.php, get_menu_satuan.php, get_menu_minuman.php, get_menu_pemandangan.php	http://000webhost.com/, get_menu_paket.php, http://000webhost.com/, /	menupaket, menusa, minuman, menupemandangan
Input pesanan	MenuModel.java, MenuAdapter.java	tambah_pesanan.php	http://000webhost.com/tambah_pesanan.php	Pesanan
Request hasil pesanan	RingkasanOrder.java	get_masterpesanan.php	http://000webhost.com/get_masterpesanan.php?token=token	pelanggan, pesanan

Berdasarkan tabel 1 diatas, menjelaskan tentang proses pemesanan menu yang dilakukan oleh pelanggan. Proses dimulai dari pelanggan menginputkan data pelanggan, berupa nama, kemudian sistem akan memberikan token secara otomatis dan nomor meja. Setelah itu aplikasi akan melakukan pengiriman data ke database dan sistem akan merequest data menu yang ada untuk ditampilkan

berdasarkan kategori menu yang ada (menu paket, menu satuan, menu pendamping dan menu minuman). Pelanggan menginputkan pesannya, yang nantinya akan masuk ke keranjang belanja yang dapat dilihat oleh pelanggan. Pelanggan dapat mengubah, menghapus dan menambahkan pesanan sebelum mengirimkannya ke database sistem yang dibangun. Dalam menjalankan fungsi-fungsi pada aplikasi android yang dibangun, diperlukan restful web service. Contohnya adalah ketika pelanggan menginputkan pesannya, maka akan menjalankan perintah yang ada dari restful http://000webhost.com/tambah_pelanggan.php yang terdapat pada file php tambah_pelanggan.php, dimana file ini di hosting ke web server dan url nya dimasukan kedalam java android.

Tabel 2 Proses Admin Mengelola Data

Skenario	Java (Android)	PHP	Restful	Tabel Database
Login admin	LoginAdmin.java	login_admin.php	http://000webhost.com/login_admin.php	admin
Tambah menu, Ubah dan Hapus.	TambahMenu.java,	tambah_menu.php, ubah_menu.php, hapus_menu.php	http://000webhost.com/tambah_menu.php, http://000webhost.com/ubah_menu.php, http://000webhost.com/hapus_menu.php	menu
Tambah pegawai Ubah dan Hapus pegawai	Tambah_Kasir.java, Tambah_Dapur.java	tambah_kasir.php, tambah_dapur.php	http://000webhost.com/tambah_kasir.php, http://000webhost.com/tambah_dapur.php	kasir, dapur
Lihat Laporan Penjualan	LihatLaporan.java	laporan.php	http://000webhost.com/laporan.php	transaksi

Tabel 2 menjelaskan tentang bagaimana proses yang terjadi pada sistem, dimana admin mengubah data, baik data menu, data pegawai maupun melihat data penjualan. Setiap fungsi yang digunakan oleh admin akan menggunakan restful web service yang berbeda. Setiap restful dari tabel di atas memiliki fungsinya masing-masing yang terintegrasi kedalam aplikasi android yang dibangun. Setiap file Java android memuat code program berupa logika aplikasi agar dapat menjalankan tugasnya masing-masing. Dalam menjalankan fungsi-fungsi pada aplikasi android yang dibangun, dalam hal ini aplikasi admin juga diperlukan

restful web service. Contohnya adalah ketika admin melakukan penambahan data menu restoran, maka akan menjalankan perintah yang ada dari restful http://000webhost.com/tambah_menu.php yang terdapat pada file php tambah_menu.php, dimana file ini di hosting ke web server dan url nya dimasukan kedalam java android. Begitu pula untuk setiap fungsi lainnya, semuanya akan menjalankan fungsi yang ada pada restful web service yang dibangun, dimana url yang mewakili fungsi masing-masing akan dimasukan kedalam java android. Java di dalam aplikasi android yang di bangun adalah memuat logika dari program itu sendiri.

Tabel 3 Proses Sistem Dapur dan Kasir

Skenario	PHP	URL	Tabel Database
Login Dapur	login_dapur.php	http://000webhost.com/login_dapur.html	dapur
Lihat pesanan	tampilan_dapur.php	http://000webhost.com/tampilan_dapur.php	masterpesanan
Hapus pesanan	tampilan_dapur.php	http://000webhost.com/tampilan_dapur.php	masterpesanan
Ubah status proses pesanan (belum-selesai).	tampilan_dapur.php	http://000webhost.com/tampilan_dapur.php	masterpesanan
Login Kasir	login_kasir.php	http://000webhost.com/login_kasir.html	kasir
Tampilan daftar meja pelanggan , pilih meja yang akan melakukan transaksi	tampilan_meja.php	http://000webhost.com/tampilan_meja.php	masterpesanan
Daftar pesanan pelanggan yang akan melakukan transaksi.	sistemkasir.php	http://000webhost.com/sistemkasir.php	masterpesanan
Cetak struk	struk.php	http://000webhost.com/struk.php	transaksi

Tabel 3 menjelaskan tentang proses yang terjadi setelah pesanan pelanggan masuk kedalam database. Staff dapur akan melakukan login kedalam sistem, kemudian melihat pesanan yang masuk, staff dapur dapat menghapus pesanan pelanggan yang hendak pelanggan batalkan. Kemudian jika proses penyiapan pesanan telah selesai maka staff dapur akan merubah statusnya dari yang sebelumnya belum menjadi selesai. Setelah itu kasir akan melakukan login untuk melakukan transaksi nasabah, kasir dapat menginputkan persenan diskon jika ada dan dapat mencetak struk melalui sistem. Isi dari tabel 3 tidak memuat tentang restful web service dikarenakan sistem yang di bangun untuk dapur dan kasir restoran adalah *web-based*. Untuk perangkat yang akan digunakan pihak restoran untuk mengakses sistem kasir dan dapur akan dikembalikan ke pihak restoran itu

sendiri, apakah akan menggunakan tablet atau menggunakan personal computer (pc) / laptop.

Tahap terakhir dari metode penelitian *extreme programming* adalah melakukan *testing* dari sistem yang dibangun. Penelitian ini menggunakan metode pengujian *black-box testing* untuk menguji setiap fungsi dari sistem yang di bangun apakah sudah berjalan sebagaimana mestinya.

Tabel 4 *Black-Box Testing* Aplikasi

No	Skenario Pengujian	Test Case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Pada Layout Pelanggan Terdapat tiga <i>Edit Text</i> , Nama Pelanggan No. Meja, Token	Klik <i>Button Kirim</i>	Dapat Mengirimkan Nama Pelanggan dan Nomor Meja Ke Web Service	Sesuai Harapan	<i>Valid</i>
2.	Pada Menu Utama Terdapat Lima Gambar <i>Image Slider</i>	Menggeser <i>Image Slider</i>	Image Slider Dapat Bergerak Secara Otomatis	Sesuai Harapan	<i>Valid</i>
3.	Pada Menu Utama Terdapat <i>Image Button</i> Menu Paket	Klik <i>Image Button</i>	Dapat Menampilkan Menu Paket dari Web Service	Sesuai Harapan	<i>Valid</i>
4.	Pada Menu Utama Terdapat <i>Image Button</i> Menu Satuan	Klik <i>Image Button</i>	Dapat Menampilkan Menu Satuan dari Web Service	Sesuai Harapan	<i>Valid</i>
5.	Pada Menu Utama Terdapat <i>Image Button</i> Menu Pendamping	Klik <i>Image Button</i>	Dapat Menampilkan Menu Pendamping dari Web Service	Sesuai Harapan	<i>Valid</i>
6.	Pada Menu Utama Terdapat <i>Image Button</i> Menu Minuman	Klik <i>Image Button</i>	Dapat Menampilkan Menu Minuman dari Web Service	Sesuai Harapan	<i>Valid</i>
7.	Pada Setiap Menu Terdapat <i>Button Minus</i>	Klik <i>Button Minus</i>	Pelanggan Tidak Bisa Memesan Kurang Dari 1 Pesanan	Sesuai Harapan	<i>Valid</i>
8.	Pada Setiap Menu Terdapat <i>Button Plus</i>	Klik <i>Button Plus</i>	Pelanggan Dapat Menambah Jumlah Pesanan	Sesuai Harapan	<i>Valid</i>
9.	Pada Setiap Menu Terdapat <i>Edit Text</i> untuk menambahkan note	Mengetik Pada <i>Edit Text</i>	Pelanggan Dapat Memberi Catatan/Note Pada Setiap Pesanan	Sesuai Harapan	<i>Valid</i>

Tabel 4 menjelaskan tentang scenario pengujian yang dilakukan, kemudian kasus dari sistem yang akan dilakukan pengecekan / *testing*. Setelah itu memuat tentang hasil yang diharapkan dari sistem dan hasil pengujian dan kesimpulannya apakah telah sesuai harapan. Jika sudah makan kesimpulannya adalah *valid*. Contohnya adalah pada tampilan pelanggan, terdapat tiga *edit text*, yaitu nama pelanggan, nomor meja dan token. Hasil yang diharapkan adalah ketika *user* menekan tombol kirim maka sistem akan mengirimkan data berupa nama pelanggan, nomor meja, dan token. Jika berhasil maka dianggap sesuai harapan dan kesimpulannya adalah *valid*. *Testing* yang dilakukan pada tabel 4 ini adalah untuk sistem pelanggan / *user*. Untuk sistem yang lainnya memiliki kasus uji yang berbeda-beda.

4. Kesimpulan

Self-service system berbasis android menggunakan *restful web service* telah berhasil dibangun dengan mengacu kepada kebutuhan bagi pihak restoran. Sistem ini memerlukan 2 (dua) komputer / laptop, tablet tergantung dari banyaknya meja pelanggan dan sebuah tablet untuk admin serta 1 (satu) perangkat wireless yang menghubungkan perangkat tablet dan laptop dengan *database*. Pengaplikasian sistem ini sangat membantu dalam mempermudah pengelolaan pesanan pelanggan dimana pelanggan dapat menginputkan sendiri pesanan yang diinginkan, bagian dapur dan kasir saling terkoneksi sehingga pada saat pelayan menginputkan data pesanan menggunakan tablet, maka data pesanan tersebut akan langsung diterima oleh bagian dapur. Ketika bagian dapur melakukan proses terhadap pesanan maka data tersebut akan langsung terkirim ke bagian kasir.

Dengan demikian, proses pengelolaan pesanan mulai dari menerima pesanan dari konsumen sampai kepada bagian kasir dapat dilakukan dengan cepat, mudah dan tentunya membuat konsumen merasa nyaman karena dapat terlayani dengan cepat. Diharapkan dapat menjadi masukan agar penelitian selanjutnya dapat dikembangkan lebih baik lagi adalah sistem akan lebih baik jika sistem dapat berjalan di sistem operasi yang berbeda seperti (*iOS, windows mobile*). Sistem pencarian pada menu tidak hanya berdasarkan kategori tertentu ataupun nama menu, tetapi juga berdasarkan *tags*. Penambahan fitur *attention message* dari pelanggan kepada staf dapur untuk mengingatkan mengenai makanan yang sudah ditunggu lama tetapi belum dihidangkan. Memberikan fitur pembayaran dan *top up* saldo melalui *smartphone* untuk metode pembayaran selain *cash/tunai*.

Ucapan Terimakasih

Peneliti mengucapkan terima kasih kepada pihak STMIK Pontianak yang telah memberikan kesempatan untuk melakukan penelitian ini. Peneliti juga berterima kasih kepada reviewer jurnal yang telah menyediakan waktu untuk memberikan bimbingan dan saran sehingga penelitian ini dapat diselesaikan.

Daftar Rujukan

- [1] R. Setiawan and R. Munajah, "Evaluation of the Application of Online Learning in Indonesian Universities", TEM Journal, vol.9, no.3, pp.1194-1199, 2020.
- [2] F. Bachtiar, et al, "Edukasi Mengenai Dampak Penggunaan Smartphone Sebagai Upaya Pencegahan Gangguan Muskuloskeletal Pada Remaja". Jurnal Abdimas Unwas, vol. 5, no. 1, pp.28-32, 2020.
- [3] J. Rewah, "Pengaruh Kualitas Layanan Online Terhadap Kepuasan Pelanggan Telkomsel Manado," *CogITO Smart Journal*, Vol. 2, No. 2, pp. 250-263, 2016.
- [4] Almohaimmed, "Restaurant Quality and Customer Satisfaction". *International Review of Management and Marketing*, Volume-7, Issue-3, hal 42-49, 2017.
- [5] Iqbal., "Sistem Pemesanan Menu Makanan Di Rumah Makan Berbasis Web Service Pada Platform Mobile", *Jurnal Mahasiswa Teknik Informatika Universitas Indonesia, Jakarta*, Vol 9 :75-80, 2018.
- [6] Kosasi, Sandy, "Perancangan Prototipe Sistem Pemesanan Makanan dan Minuman Menggunakan Mobile Device", *Jatiti*, Vol. 1, No. 2, hal. 175-187, 2015.
- [7] H. T. Tsou, & H. Y. Hsu, "Self-Service Technology Investment, Electronic Customer Relationship Management Practices, and Service Innovation Capability," *IJCSMC*, vol 4 :142-149, 2019.
- [8] Fachira, "The Influence Of Self-service Technology (SST) Toward Customer Satisfaction", *Journal Of Business and Management*, Vol. 4, No. 6, pp. 728-742, 2017.
- [9] Hollyvia, "Self-Service Technology (Sst) Acceptance Analysis In Restaurants In Indonesia By Usingtechnology Acceptance Model (TAM)", *Jurnal Hospitality dan Manajemen Jasa*, Vol. 6, No. 1, pp. 156-158, 2018.
- [10] Sommerville, Ian., "Software Engineering Rekayasa Perangkat Lunak", Erlangga, Jakarta, 2018.
- [11] J. Meng, "RESTful Web Services: A Solution for Distributed Data Integration, Comput. Intell. Softw. Eng", *CiS. Int. Conf.*, vol. 6, no. 4, pp. 1-4, 2016.
- [12] Munir, dkk, "Analisis Dan Perancangan Representational State Transfer (REST) Web Service Sistem Infromasi Akademik STT Terpadu Nurul Fikri Menggunakan YII Framework", *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol-2, No.2, ISSN 2477-0043, hal 120-121, 2016.
- [13] Kosasi, Sandy, "Perancangan Prototipe Sistem Pemesanan Makanan dan Minuman Menggunakan Mobile Device", *Jatiti*, Vol. 1, No. 2, hal. 175-187, 2015.
- [14] Yuliani, I.D.A.E, "Self-Service Technology Berbasis Smartphone Device pada Sistem Pemesanan Menu Makanan", *Jurnal Seminar Nasional Sistem Informasi dan Teknologi Informasi STMIK Pontianak, Pontianak*, 2018.
- [15] Lubis, B. O, "Penerapan Global Extreme Programming Pada Sistem Informasi Workshop, Seminar Dan Pelatihan Di Lembaga Edukasi". *Jurnal Informatika*, Vol 3(2), 234-246, 2016.
- [16] Rosa dan Shalauhuddin. M, "Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek", Informatika, Bandung, 2016.