

Penerapan *Firestore Realtime Database* Pada *Prototype* Aplikasi Pemesanan Makanan Berbasis Android

<http://dx.doi.org/10.28932/jutisi.v4i3.870>

George Richard Payara¹, Radius Tanone²

¹Mahasiswa Fakultas Teknologi Informasi Jurusan Teknik Informatika, Universitas Kristen Satya Wacana Salatiga

¹georgerichardpayara@gmail.com

²Staff Pengajar Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana Salatiga

²radius.tanone@uksw.edu

Abstract— *In this modern era, the action of information retrieval process and meals ordering in the culinary business become a trending issue, from the information about the lowest until the expensive prices, and from the foods stalls until the luxurious restaurant. One of the samples in Warung Lawas in Salatiga city, whereas the process of searching and ordering foods are still manual which is the customers must find the place to eat and order the meals manually, therefore the process of culinary business in this present time is not really effective yet in saving the time. Nowadays, by employing firestore technology named 'realtime database' which is highly developed, for this reason an application to order the meals is made by using 'cloud computing' technology with 'realtime database' firestore based on Android. The use of 'realtime database' firestore aims to transfer the data 'realtime' because in that way the process of ordering by using the application will be faster. The result of this research is to point out that the system that had been made to help the user in ordering the meals in Warung Lawas Salatiga turns into being more effective and efficient.*

Keywords—*Firestore Realtime Database, Android platform, Restaurant.*

I. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi yang semakin pesat dan berkembang saat ini berdampak juga terhadap perkembangan teknologi yang digunakan oleh masyarakat dalam menjalankan kesehariannya (aktivitas), segala sesuatu harus dilakukan mudah dan cepat. Masyarakat saat ini membutuhkan teknologi yang dapat membantu mengatasi masalah-masalah yang dihadapi dalam kehidupan sehari-hari yang mencakup seluruh aspek kehidupan masyarakat saat ini, mulai dari dunia hiburan, pendidikan, hingga meluas ke dunia bisnis. Munculnya berbagai macam aplikasi memberikan pilihan dalam peningkatan kerja suatu pekerjaan, baik yang bersifat *desktopbased*, *webbased* hingga yang sekarang ini

munculnya aplikasi-aplikasi baru yang berjalan dalam *mobile* seperti pada sistem platform android. Pemilihan *mobile* android untuk salah satu pengembangan aplikasi selain lebih mudah dalam pengoperasiannya, dan juga sifat dari *mobile* yang fleksibel menjadi salah satu alasannya. Banyak pelaku industri, lembaga pendidikan, lembaga kesehatan dan lainnya banyak menggunakan *mobile* dan internet sebagai media penyajian informasi mereka khususnya banyak digunakan dimasyarakat moderen di perkotaan.

Dalam proses bisnis pada usaha kuliner saat ini proses pencarian informasi dan pemesanan makanan merupakan hal yang sangat penting, mulai dari informasi harga dari yang murah sampai yang mahal, dan tempat makan yang biasa sampai yang mewah. Pada proses pencarian tempat makan pelanggan mencari tempat makan secara manual dengan informasi yang minim seperti tempat yang akan dikunjungi memiliki menu makan apa saja atau tempat yang akan dikunjungi dibuka atau tidak. Seperti halnya pada warung Lawas yang berada di kota Salatiga dengan lokasi yang mudah dicari dan menu makanan yang relatif murah. Pada proses pemesanan makanan di warung Lawas masih menggunakan cara biasa, terdapat dua cara manual dalam proses pemesanan makanan yaitu yang pertama pelanggan harus menghampiri pelayan untuk memesan makanan dengan cara mencatat semua pesanan makanannya pada nota pemesanan dan yang kedua adalah pelanggan harus menunggu pelayan datang ke meja makanan untuk memberikan nota pemesanan dan menu makanan. Disamping itu terkadang pelanggan kesulitan untuk memanggil pelayan datang ke meja pelanggan yang akan memesan menu makanan. Hal ini tentu akan membuat para pelanggan/pengunjung kesal karena harus mengorbankan waktu untuk menunggu saat melakukan proses pemesanan makanan dan tentu juga harus menunggu untuk

mendapatkan makanan yang sudah lama di pesan[1].

Firestore adalah API yang disediakan *google* untuk penyimpanan dan penyelarasan data ke dalam aplikasi Android, iOS, atau web. *RealtimeDatabase* adalah salah satu fasilitas yang menyimpan data ke *database* dan mengambil data darinya dengan sangat cepat tetapi *firebase* bukan hanya *realtimeDatabase*, jauh lebih dari itu. *Firestore* memiliki banyak fitur seperti *authentication*, *database*, *storage*, *hosting*, pemberitahuan dan lain-lain[2].

Berdasarkan latar belakang di atas maka dilakukan penelitian senada dengan judul " Penerapan *Firestore Realtime Database* Pada *Prototype* Aplikasi Pemesanan Makanan Berbasis Android". Rumusan masalah adalah apakah *Firestore RealtimeDatabase* dapat diterapkan dalam merancang aplikasi pemesanan makanan. Tujuan dan manfaat penelitian ini adalah merancang aplikasi pemesanan makanan dengan menerapkan Teknologi *Firestore RealtimeDatabase* sehingga dapat membantu dalam memberikan informasi tempat makan dan proses pemesanan makanan yang lebih mudah dan cepat. Batasan masalah pada penelitian ini adalah sistem yang dibuat untuk pemesanan makanan berbasis android dengan menerapkan teknologi *firebase realtimeDatabase*.

II. TINJAUAN PUSTAKA

Pada penelitian yang berjudul "Aplikasi Pemesanan Makanan Berbasis *Mobile* Pada Rumah Makan "Lek Nonong" membahas tentang bagaimana merancang aplikasi *mobile* untuk pemesanan makanan pada rumah makan "Lek Nonong" yang berada disekitar daerah Tembalang, Kota Semarang berbasis android. Pembuatan aplikasi menggunakan bahasa pemrograman *mobile*, HTML, PHP, *Javascript* dan MySQL. Selain itu untuk permodelan aplikasi menggunakan DFD (*Data Flow Diagram*). Hasil penelitiannya menghasilkan aplikasi order makanan berbasis *mobile* yang dapat di gunakan untuk memproses dan mengelola laporan transaksi yang ada di Rumah Makan tersebut[3].

Penelitian selanjutnya dengan judul "Aplikasi Pemesanan Makanan Berbasis *Cloud* dengan *Platform* Android" membahas tentang bagaimana merancang aplikasi *mobile* android yang memanfaatkan fasilitas internet (*cloud*) yang diterapkan pada pihak restoran dalam melakukan dan melayani pemesanan makanan. Aplikasi ini harus terhubung dengan internet (*cloud*) untuk bisa mengakses restoran yang terdaftar pada sistem. Hasil dari penelitian ini berupa aplikasi untuk memudahkan pemesanan makanan bagi masyarakat, khususnya bagi orang-orang yang memiliki banyak kesibukan [4].

Pada penelitian yang berjudul "Implementasi Teknologi *Firestore* Pada Aplikasi Pencarian Lokasi *Service* Kamera Berdasarkan *Rating* Berbasis Android" membahas tentang bagaimana merancang aplikasi *mobile* android yang memanfaatkan teknologi *firebase* untuk melakukan

pencarian lokasi *service* kamera berdasarkan *rating* berbasis android. Hasil dari penelitian ini memiliki fasilitas untuk menampilkan beberapa lokasi pencarian *service* kamera di wilayah Yogyakarta. Fungsionalitas dari penelitian ini adalah menampilkan *profile* lokasi seperti lokasi melalui *Google MAPS*, menghubungi langsung via *Message*, telepon seluler, dan *e-mail* serta pemberian *rating* terhadap lokasi tersebut. [5].

Mengacu pada penelitian-penelitian terdahulu maka dapat disimpulkan bahwa implementasi pemesanan makan dalam bentuk aplikasi berbasis android sangat membantu para konsumen atau pelanggan untuk melakukan pemesanan makanan. Selain itu aplikasi pemesanan makanan juga mempermudah pihak admin untuk mengelola data pemesanan. Berbeda dengan penelitian sebelumnya maka pada penelitian ini akan diteliti mengenai aplikasi penerapan teknologi *Firestore RealtimeDatabase* dalam pembuatan aplikasi pemesanan makanan. Fokus penelitian ini adalah membahas tentang perancangan aplikasi pemesanan makanan untuk menggunakan teknologi *firebase realtimeDatabase* berbasis android. Aplikasi yang dibangun diharapkan dapat membantu masyarakat dalam proses pemesanan makan yang lebih mudah dan efisien.

Android adalah sistem operasi untuk telepon seluler berbasis Linux sebagai kernelnya. Saat ini Android menjadi pesaing utama dari produk *smartphone* lainnya seperti *Apple* dan *Blackberry* karena Android memiliki beberapa kelebihan daripada *smartphone* lain, yaitu:

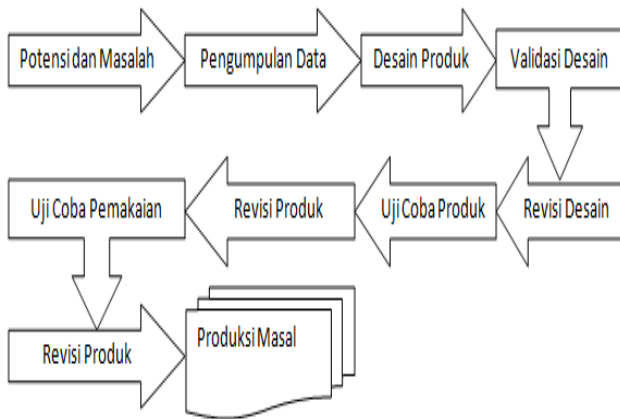
- Android bersifat *OpenSource* yang artinya pengembang (*Developer* Android) bebas untuk mengembangkan aplikasi pada *platform* ini.
- Lengkap artinya android menyediakan *tools* untuk membangaun *software* yang sangat lengkap dibanding dengan *platform* lain.
- Bebas berarti android merupakan *platform mobile* yang tidak memiliki batasan dalam mengembangkan aplikasinya. Tidak ada lisensi dalam mengembangkan aplikasi Android[6].

Firestore memiliki produk utama, yaitu menyediakan *database realtime* dan backend sebagai layanan (*Backend as a Service*). Layanan ini menyediakan pengembang aplikasi API yang memungkinkan aplikasi data yang akan disinkronisasi di klien dan disimpan di *cloud Firestore*. *Firestore* menyediakan *library* untuk berbagai *client platform* yang memungkinkan integrasi dengan Android, iOS, *JavaScript*, *Java*, *Objective-C* dan *Node* aplikasi Js dan dapat juga disebut sebagai layanan *DbaaS (Database as a Service)* dengan konsep *realtime*. *Firestore* digunakan untuk mempermudah dalam penambahan fitur-fitur yang akan dibangun oleh developer. *DatabaseRealtime* merupakan basis data dalam *firebase* yang berbasis *cloud* dan tidak memerlukan *query* berbasis SQL untuk menyimpan dan mengambil data. Basis data ini terkenal sangat handal dan supercepat dalam peroses *update* data dan sinkronisasi sehingga data tetap dipertahankan bahkan ketika *user* tidak

terhubung dengan internet sekalipun data tetap di pertahankan[7].

III. METODE DAN PERANCANGAN SISTEM

Dalam penelitian ini peneliti menggunakan metode penelitian dan pengembangan atau lebih dikenal dengan *Research and Development*. Metode penelitian dan pengembangan adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut [8]. Penelitian ini merupakan penelitian kualitatif dengan analisa deskriptif. Metode pengumpulan data dilakukan dengan metode wawancara serta observasi langsung di lapangan. Langkah-langkah penelitian dapat dilihat pada Gambar 1.

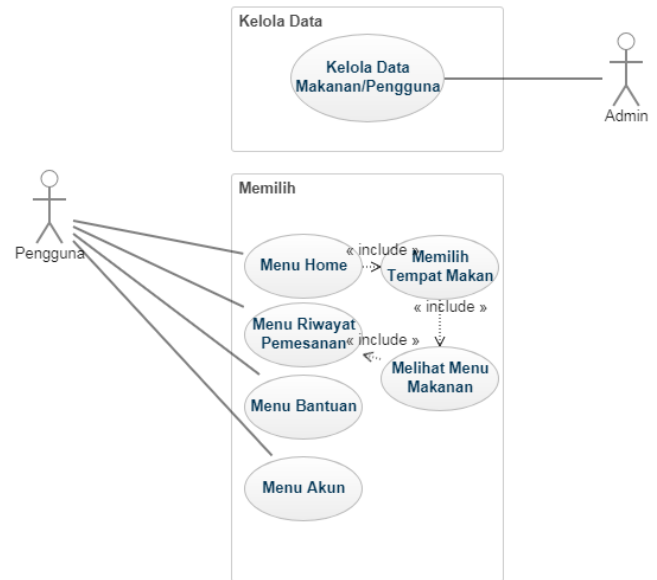


Gambar 1. Langkah Penelitian [8]

Pada penelitian ini tahapan penelitian hanya sampai pada uji coba pemakaian, tidak sampai pada revisi produk dan produksi masal. Berdasarkan tahapan-tahapan diatas penelitian ini dilakukan dari tahap potensi dan masalah sampai uji coba pemakain dapat dijelaskan sebagai berikut. Potensi dan masalah, peneliti melakukan observasi pada Warung Lawas yang berada di kota Salatiga. Tahap pengumpulan data dilakukan pengumpulan informasi tentang proses bisnis pada sejumlah rumah makan yang ada di Salatiga serta pengumpulan informasi tentang apa saja yang menjadi kendala dalam pada proses bisnis yang ada. Langkah *design* produk merupakan tahapan dimana peneliti mulai membuat *design* dalam bentuk diagram UML. Langkah validasi produk merupakan merupakan penilaian atas palikasi apakah sudah layak untuk dipakai atau tidak. Langkah revisi *design* adalah langkah dimana peneliti melakukan revisi atas *design* yang telah di kerjakan sebelumnya sehingga layak digunakan oleh *user*. Setelah direvisi maka aplikasi yang dibuat dapat diuji coba. Dalam pelaksanaan uji coba terdapat beberapa revisi aplikasi yang harus dikerjakan oleh peneliti. Hasil akhir dari revisi ini adalah dilakukan uji coba pemakaian oleh calon pengguna (pengujian *beta*) kemudian pengguna memberikan

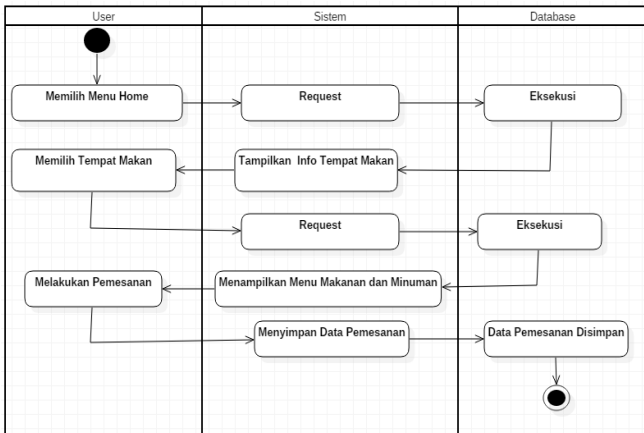
tanggapan melalu kuisisioner, hasil dari kuisisioner untuk mengetahui apakah aplikasi dapat menyelesaikan masalah pada penelitian ini.

Perancangan sistem dalam penelitian ini dibuat menggunakan *Unifed Modelling Language* (UML) yang meliputi *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*[9].



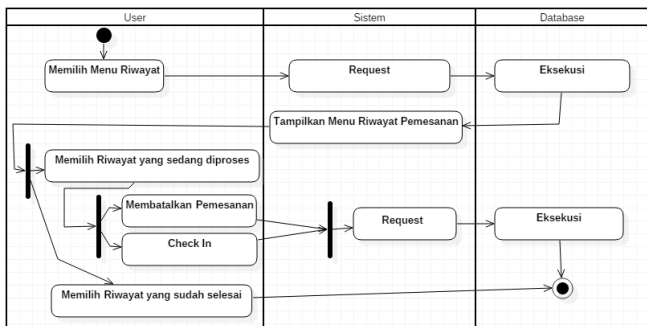
Gambar 2. Use Case Diagram

Gambar 2 merupakan *use case* dari sistem yang dibuat, dimana *user* sebagai pelanggan harus melakukan *login* terlebih dahulu sebelum masuk ke sistem. Setelah berhasil *login* maka sistem akan menampilkan empat menu utama yaitu menu *Home*, *Riwayat Pemesanan*, *Bantuan* dan *Akun*. Untuk melakukan pemesanan makanan *user* harus memilih terlebih dahulu tempat makan, setelah itu memilih menu makanan yang sesuai dengan keinginan baru bisa melakukan proses pemesanan makanan. Selanjutnya Menu *Riwayat Pemesanan* akan menampilkan semua aktivitas pemesanan dari *user* termasuk status pemesanan yang sudah sedang diproses atau sudah selesai diproses. Menu *bantuan* akan berisi panduan penggunaan aplikasi dan menu *Akun* akan menampilkan sejumlah data data pribadi dari *user* yang sudah menggunakan aplikasi tersebut serta jika sudah selesai menggunakan aplikasi maka *user* tinggal *logout*.



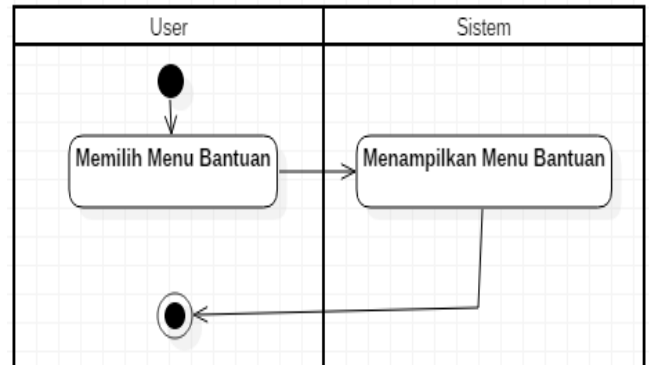
Gambar 3. Activity Diagram Menu Home

Gambar 3 merupakan Activity Diagram menu home. Jika user berhasil login maka sistem akan menjalankan. User memilih menu Home kemudian sistem akan menampilkan info tempat makan yang diambil dari database. Setelah user memilih tempat makan yang sesuai dan melakukan pemesanan maka data pemesanan akan diinputkan ke database dan akan ditampilkan pada menu Riwayat Pemesanan.



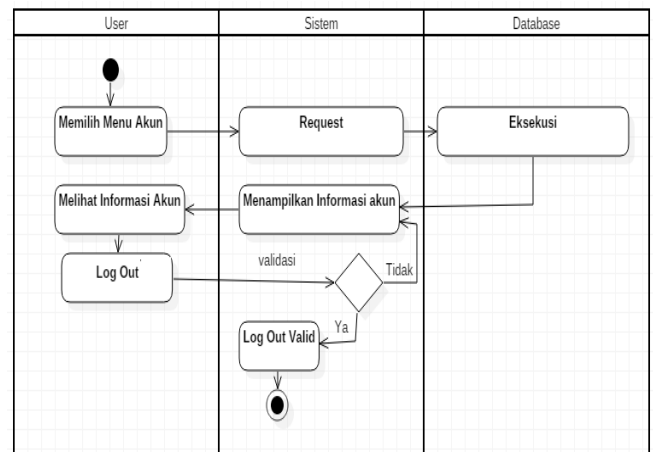
Gambar 4. Activity Diagram Menu Panduan

Gambar 4 merupakan Activity Diagram Menu Panduan. Pertama User memilih menu Riwayat Pemesanan kemudian sistem akan menampilkan data pemesanan pada user yang diambil dari database kemudian dipecah menjadi dua yaitu Riwayat yang sedang diproses dan riwayat yang sudah selesai diproses. Pada riwayat yang sedang diproses user dapat melakukan pembatalan pemesanan atau melakukan CheckIn. Gambar 5 menunjukkan activity diagram menu Riwayat Pemesanan.



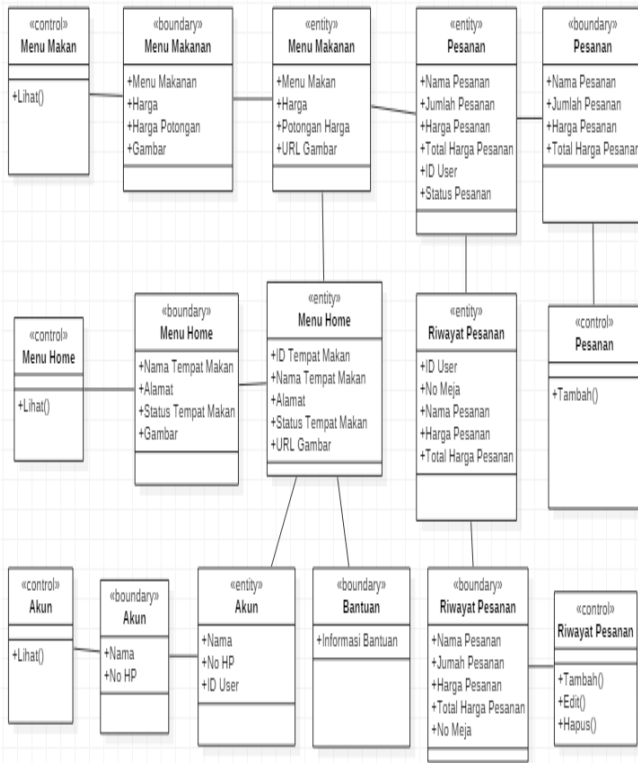
Gambar 5. Activity Diagram Menu Bantuan

Gambar 5 menunjukkan Activity Diagram Menu Bantuan . Pada activity ini user memilih menu Bantuan kemudian sistem akan menampilkan isi dari menu Bantuan.



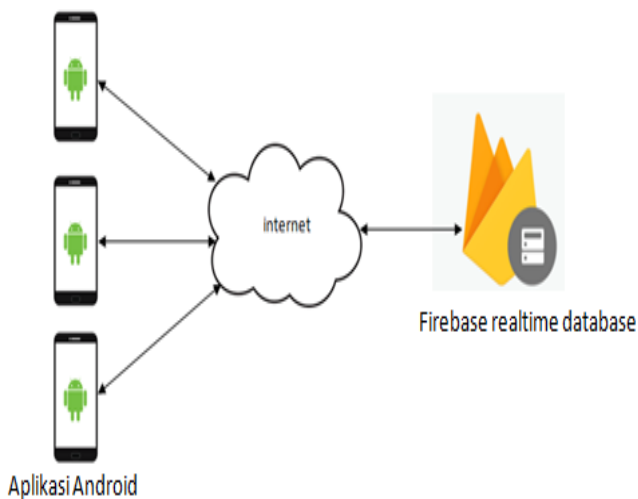
Gambar 6. Activity Diagram Menu Panduan

Gambar 6 menunjukkan activity diagram menu panduan. User memilih menu Akun kemudian sistem akan menampilkan isi dari menu Akun. Pada menu akun user dapat melakukan logout untuk keluar dari hak akses sistem.



Gambar 7. Class Diagram.

Gambar 7 menunjukkan class diagram pada aplikasi pemesanan makanan menggunakan cloud computing dengan firebase realtime database. Pada Gambar 7 terdapat lima kelas yang digunakan dalam pembuatan sistem. Class diagram menunjukkan bahwa aplikasi terdiri atas tiga bagian utama yaitu entity, control, dan boundary. Entity berhubungan langsung dengan data-data yang ada di database layaknya tabel dan field-field nya, control berhubungan langsung dengan fungsi-fungsi pengelolaan data seperti insert, update dan delete, dan boundary merupakan tampilan dari aplikasi tersebut.

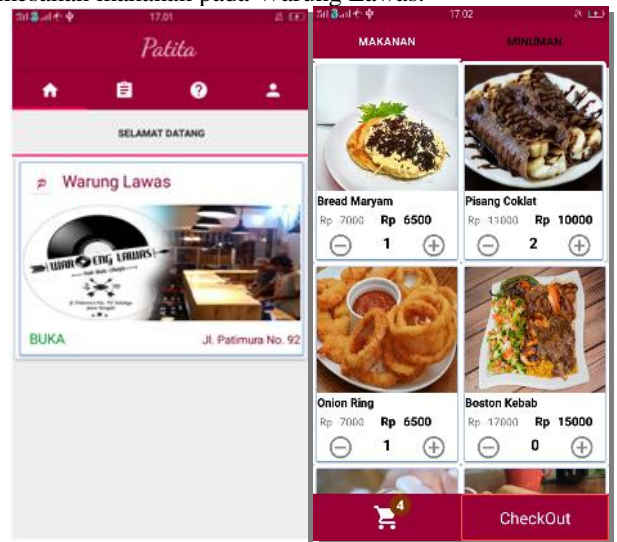


Gambar 8. Arsitektur Sistem[10]

Gambar 8 merupakan arsitektur sistem yang akan dibangun. Aplikasi berbasis android akan terhubung ke internet untuk komunikasi dengan server. Untuk melakukan transaksi data seperti insert, update atau delete pada database, digunakan firebase API sebagai jembatan antara mobile dan database. firebase API berupa library yang telah disediakan oleh firebase. Library tersebut menyediakan berbagai fungsi untuk berkomunikasi dengan server.

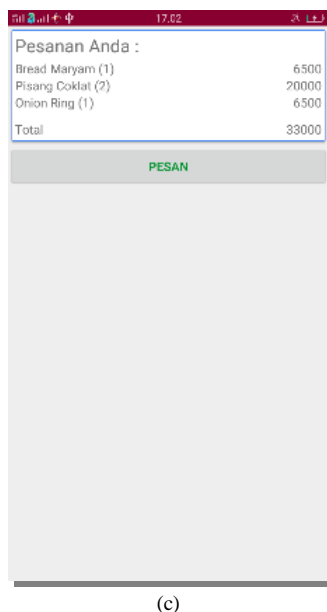
IV. HASIL DAN PEMBAHASAN

Hasil dan pembahasan meliputi aplikasi mobile android untuk pemesanan makanan dengan menerapkan firebase realtime database yang dapat digunakan sesuai dengan analisis kebutuhan yang telah dilakukan. Aplikasi ini berfungsi untuk membantu pelanggan dalam proses pemesanan makanan pada Warung Lawas.



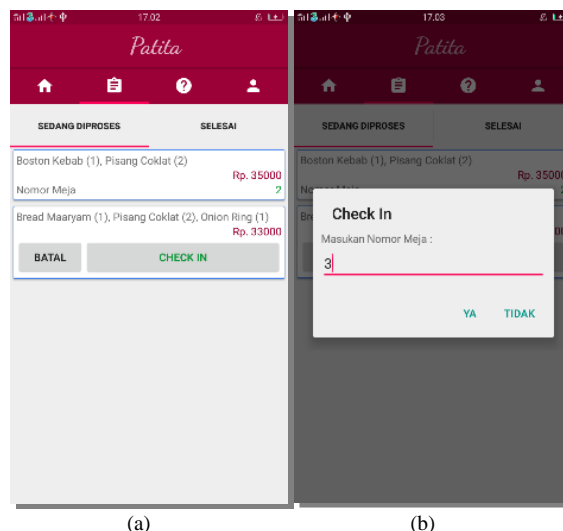
(a)

(b)

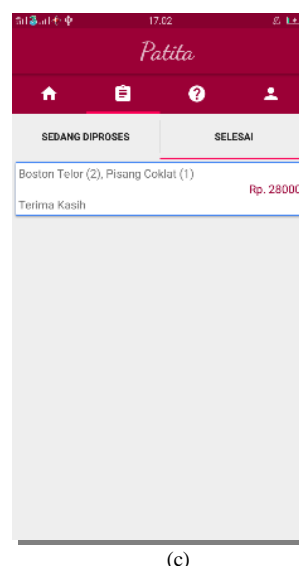


Gambar 9. (a) Menu Home (b) Menu Makanan (c) Menu Checkout

Setelah melakukan proses login maka akan masuk ke Menu Home. Gambar 9 a menampilkan menu home, pada menu home terdapat informasi tempat makan berupa nama tempat makan, alamat, dan status tempat makan apakah buka atau tidak. Untuk melakukan pemesanan maka user akan memilih tempat makan dan akan masuk pada form daftar menu makanan dan minuman seperti pada Gambar 9 b. Pada Gambar 9 b menampilkan informasi tentang menu makan seperti nama makanan, harga, dan gambar menu. Untuk melakukan pesanan selanjutnya user memilih menu yang diinginkan dengan cara menekan tombol + untuk menambahkan ke keranjang untuk membatalkannya user dapat menekan tombol - setelah itu user menekan tombol Checkout untuk melanjutkan ke pemesanan. Gambar 9 c menampilkan data pesanan yang telah dipilih data yang ditampilkan berupa nama menu, jumlah yang dipesan permenu, harga permenu, dan total harga. User melakukan proses pemesanan dengan menekan tombol PESAN.



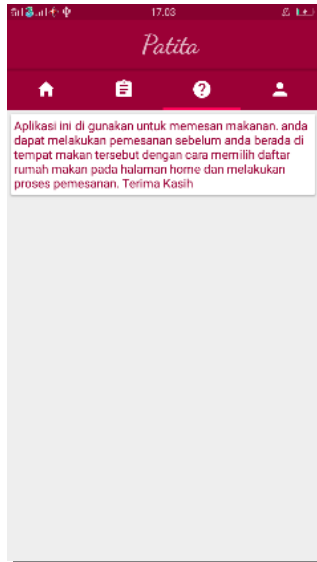
(a) (b)



(c)

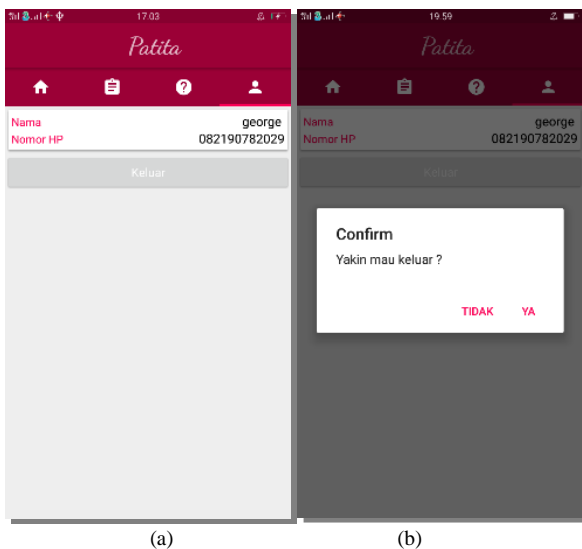
Gambar 10. (a) Riwayat pemesanan sedang diproses (b) Proses Checkin (c) Riwayat pemesanan yang telah selesai

Setelah melakukan pemesanan data akan dimunculkan pada menu riwayat, pada menu riwayat yang sedang diproses terdapat dua tombol yaitu tombol BATAL dan CheckIn tombol BATAL digunakan untuk membatalkan pesanan sedangkan tombol CheckIn digunakan untuk CheckIn seperti yang ditampilkan pada Gambar 10 a. Gambar 10 b menampilkan proses CheckIn dimana user akan menginputkan nomor meja untuk melakukan CheckIn. Gambar 10 c menampilkan pemesanan yang telah selesai diproses.



Gambar 11. Menu Bantuan

Gambar 11 menampilkan Menu Bantuan, pada menu bantuan terdapat informasi seputar aplikasi dan penjelasan singkat tentang cara penggunaan aplikasi. Pada menu bantuan ini informasi yang ditampilkan langsung menggunakan *EditText* pada android tidak mengambil data dari *database*.



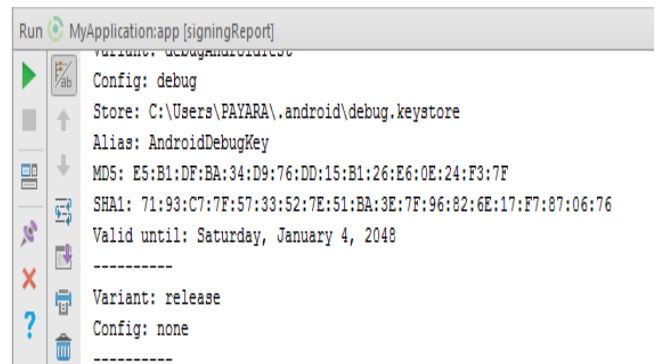
Gambar 12. (a) Menu akun (b) Proses Logout

Gambar 12 a menampilkan Menu akun, pada menu akun terdapat informasi *user* dan tombol Keluar. Gambar 12 b menampilkan *form* konfirmasi keluar jika *user* memilih YA maka aplikasi akan keluar dari sistem dan kembali ke *form login* jika TIDAK maka aplikasi tetap ada pada sistem.

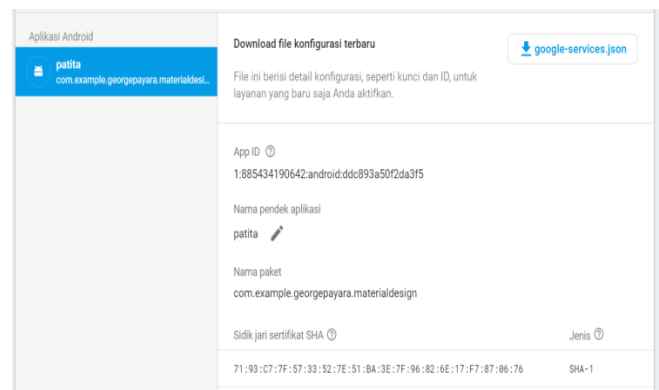
Kode Program 1 *Library firebase*

```
1. compile 'com.google.firebase:firebase-core:11.8.0'
2. compile 'com.google.firebase:firebase-auth:11.8.0'
3. compile 'com.google.firebase:firebase-database:11.8.0'
```

Kode Program 1 merupakan *library firebase* yang harus ditambahkan pada *methoddependencies* pada *build gradleapp* pada aplikasi android. Padabaris perintah ke-1 merupakan *library* utama *firebase*. Pada baris yang ke-2 merupakan *library firebase* untuk *authentication* ini digunakan mengakses *authenticationuser* sehingga dapat untuk membatasi pihak luar dalam proses pembacaan *database*. Pada sistem ini penulis memilih untuk menggunakan *Authentication* menggunakan nomor *handphoneuser*. Selanjutnya pada baris ke-3 merupakan *library firebase database* ini digunakan untuk mengakses dan melakukan pengolahan *database* pada *firebase*. Pada aplikasi ini menggunakan versi 11.8.0.



(a)



(b)

Gambar 13. (a) *Sertificate* SHA (b) Hasil pendaftaran aplikasi

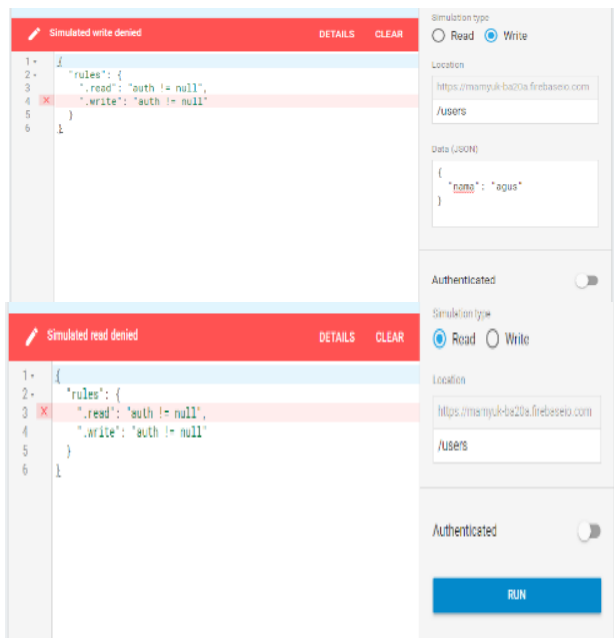
Gambar 13 merupakan proses hasil pendaftaran aplikasi pada *firebase*. Pada proses pendaftaran aplikasi pada *firebase* dibutuhkan nama *package* dari aplikasi serta kode *sertificate* SHA yang didapatkan dari *signingReport* pada aplikasi seperti pada yang ditampilkan pada Gambar 13 a. Gambar 13 b merupakan hasil pendaftaran aplikasi pada

firebase. Selanjutnya mengunduh file *google-service.json* dan dimasukkan ke dalam folder "\$folder_project\$/app". File tersebut berisi data seperti *project_id*, *firebase_url*, *api_key* dan *client_id*.



Gambar 14. Struktur Database Realtime.

Gambar 14 menampilkan struktur database untuk data *user* pada *firebase*. Database pada *firebase* tidak menggunakan sistem tabel tetapi sudah dalam bentuk pohon json. Sehingga untuk membaca data pada *firebase realtime* database tidak dibutuhkan lagi proses perubahan data dari tabel ke format json. Hal ini menyebabkan proses transfer data lebih cepat dibandingkan menggunakan database SQL.



Gambar 15. Simulasi write dan read data pada *firebase realtimedatabase*

Gambar 15 menampilkan proses simulasi *write* dan *read* data pada *firebase realtimedatabase*. Menunjukkan bahwa jika pada proses *write* atau *read* tidak memiliki *authenticationuser* maka data pada database tidak bisa diakses. *Firestore* memiliki fitur *rules* untuk mengatur hak

akses *user* dalam pembacaan data seperti pada Gambar 16 jika *authentication* tidak sama dengan *null* maka dapat melakukan *write* dan *read*. Sehingga data pada database aman tidak bisa diakses oleh pihak luar walaupun dengan menggunakan URL database.

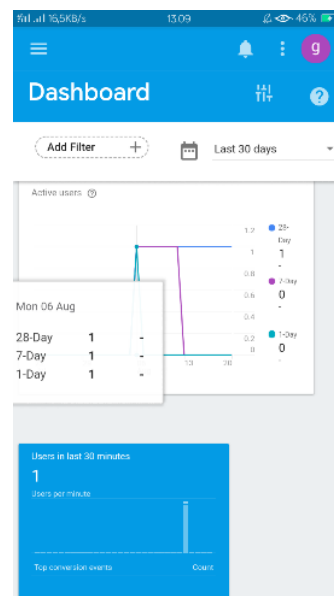
Kode Program 2 StartAuthentication

```

1. public void startauthentication(){
2.     SharedPreferences mPreferences;
3.     mPreferences = getSharedPreferences("User", MODE_PRIVATE);
4.     temporary = mPreferences.getString("saveuserid", "");
5.     if(temporary != null && !temporary.isEmpty()){
6.         mAuth = FirebaseAuth.getInstance();
7.         mFirebaseDatabase = FirebaseDatabase.getInstance();
8.         myRef = mFirebaseDatabase.getReference();
9.         FirebaseAuth currentUser = FirebaseAuth.getInstance().getCurrentUser();
10.        userid = currentUser.getId();
11.    } else{ Intent y = new Intent(MainActivity.this, LoginActivity.class);
12.        y.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
13.        y.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK);
14.        startActivity(y); } }

```

Kode Program 2 merupakan proses pembacaan *authentication* pada aplikasi pada baris 2 sampai dengan baris ke 4 merupakan proses pengambilan data *userID* data yang diambil merupakan data *preferencessaveuserid* ketika proses *login*. Pada baris ke 5 sampai dengan baris yang ke 10 merupakan proses pengecekan apakah *preferencessaveuserid* memiliki data atau tidak jika memiliki data maka dilakukan proses inialisasi data *firebase* sehingga dapat di akses dapat diakses. Pada baris 11 sampai dengan baris yang ke 14 proses perpindahan layout login jika *preferencessaveuserid* tidak memiliki data.



Gambar 16. Active Users

Dengan menggunakan *firebase* dapat dilakukan pengontrolan data *user* aktif untuk 28 hari terakhir 1 minggu terakhir dan 1 hari terakhir dan juga dapat melihat berapa

user yang aktif pada 30 menit terakhir seperti yang di tampilkan pada Gambar 16.

V. PENGUJIAN APLIKASI

Pengujian aplikasi dilakukan dengan menguji fungsi-fungsi dari aplikasi yang telah dibuat untuk mencari kesalahan/bug pada sistem. Pengujian aplikasi dilakukan agar sistem yang dibuat berjalan sesuai dengan yang diharapkan dan dapat memenuhi kebutuhan pengguna. Pengujian aplikasi ini menggunakan teknik pengujian yaitu pengujian alpha dan pengujian beta. Pengujian alpha menggunakan metode *Blackbox* yaitu pengujian fungsifungsi aplikasi secara langsung tanpa memperhatikan alur eksekusi program. Pengujian ini dilakukan dengan memperhatikan apakah fungsi telah berjalan sesuai rancangan dan sesuai yang diharapkan. Tabel I adalah hasil pengujian dari aplikasi yang telah dilakukan.

TABEL I
HASIL PENGUJIAN BLACKBOX

Fungsi yang diuji	Kondisi	Output yang diharapkan	Output yang dihasilkan	Status pengujian
Login	Login Verifikasi nomor handphone	Sukses login Mendapat kode verifikasi berupa sms ke nomor handphone	Sukses login Mendapat kode verifikasi berupa sms ke nomor handphone	valid
Menu Home	Data tempat makan Pemesanan	Menampilkan data tempat makan Menambahkan menu ke keranjang dan melakukan checkout	Menampilkan data tempat makan Menambahkan menu ke keranjang dan melakukan checkout	valid
Menu Riwayat	Data pemesanan	Menampilkan data pemesanan yang sedang diproses dan yang sudah selesai diproses serta dapat melakukan checkin	Menampilkan data pemesanan yang sedang diproses dan yang sudah selesai diproses serta dapat melakukan checkin	valid

Fungsi yang diuji	Kondisi	Output yang diharapkan	Output yang dihasilkan	Status pengujian
Menu Bantuan	Data Bantuan	Menampilkan informasi bantuan	Menampilkan informasi bantuan	valid
Menu Akun	Data akun Logout	Menampilkan data akun Melakukan Logout	Menampilkan data akun Melakukan Logout	valid

Berdasarkan hasil pengujian dari masing-masing proses pada Tabel I, dapat dilihat setiap fungsi *valid*, maka dapat disimpulkan bahwa aplikasi ini berjalan sesuai yang diharapkan.

Pengujian beta dengan kuisioner tanggapan pengguna terhadap aplikasi tersebut. Pengguna terlebih dahulu mencoba aplikasi kemudian pengguna diminta untuk memberikan tanggapan melalui kuisioner.

TABEL II
HASIL JAWABAN KUISIONER

No	Pertanyaan	STS	TS	CS	S	SS
1	Apakah aplikasi dapat memberikan informasi tempat makan?	0	2	6	13	9
2	Apakah aplikasi mempermudah dalam proses pemesanan makanan?	0	0	7	16	7
3	Apakah proses pemesanan menggunakan aplikasi lebih mudah dan cepat dalam proses pemesanan serta proses update data pemesanan?	0	1	3	12	14
4	Apakah tampilan pada aplikasi menarik untuk digunakan?	0	0	0	21	9
5	Apakah penggunaan menu dan fitur yang tersedia pada aplikasi mudah digunakan?	0	1	3	15	11

Setelah semua jawaban diketahui proses selanjutnya adalah mengukur hasil jawaban menggunakan skala *Likert*[11]. Sebelumnya tiap-tiap jawaban diberi skor, skor 1 untuk sangat tidak setuju, skor 2 untuk tidak setuju, skor 3 untuk cukup setuju, skor 4 untuk setuju, skor 5 untuk sangat setuju. Skor tertinggi adalah $5 \times 30 = 150$. *Interval* skor adalah $100/5 = 20$. Jadi kriteria interpretasi skornya berdasarkan interval skor dapat dilihat pada Tabel III.

TABEL III
KRITERIA INTERPRETASI SKOR

Rentang Skor	Kategori
0% - 19,99%	Sangat Tidak Setuju (STS)
20% - 39,99%	Tidak Setuju (TS)
40% - 59,99%	Cukup Setuju (CS)
60% - 79,99%	Setuju (S)
80% - 100%	Sangat Setuju (SS)

Jawaban pada pertanyaan 1 memiliki index 79,33% sehingga masuk pada kategori setuju. Jadi dapat disimpulkan aplikasi ini dapat memberikan informasi tempat makan. Jawaban pada pertanyaan 2 memiliki index 80% sehingga masuk pada kategori sangat setuju. Jadi dapat disimpulkan aplikasi ini dapat mempermudah dalam proses pemesanan makanan. Jawaban pada pertanyaan 3 memiliki index 86% sehingga masuk pada kategori sangat setuju, jadi dapat disimpulkan proses pemesanan menggunakan aplikasi ini lebih mudah dan cepat. Jawaban pada pertanyaan 4 memiliki index 86%, masuk pada kategori sangat setuju. Jadi dapat disimpulkan tampilan pada aplikasi ini menarik untuk digunakan. Jawaban pada pertanyaan 5 memiliki index 84%, masuk pada kategori sangat setuju. Jadi dapat disimpulkan bahwa penggunaan menu dan fitur yang tersedia pada aplikasi mudah digunakan.

VI. KESIMPULAN

Berdasarkan pembahasan dan hasil pengujian maka dapat disimpulkan bahwa perancangan aplikasi pemesanan makanan dapat dibuat dengan menerapkan teknologi *firebase realtime database* berbasis android karena *firebase* memiliki banyak fitur seperti *authentication*, *database*, *storage*, *hosting*, pemberitahuan sehingga dapat mempermudah pelanggan dalam melakukan proses pemesanan makanan pada Warung Lawas tanpa harus

menunggu lama. Melihat dari beberapa tahapan yang telah dilaksanakan serta hasil-hasil yang telah dicapai secara umum aplikasi ini telah bekerja cukup baik dimana pelanggan dapat melakukan pemesanan makanan sebelum berada di tempat makan sehingga dapat menghemat waktu dalam proses pemesanan makan. Dengan menggunakan *firebase* proses *transfer* data cepat serta aman. Saran pengembangan aplikasi adalah menambahkan fitur *google maps* sehingga pelanggan lebih dipermudah dalam pencarian tempat makan dan juga dapat menambahkan fitur android *pay* untuk proses pembayaran sehingga pelanggan tidak perlu lagi bertemu dengan pelayan untuk proses pembayaran.

DAFTAR PUSTAKA

- [1] Inayah, A. R. Aplikasi Pemesanan Menu Makanan Di Rumah Makan Berbasis *Web Service* Menggunakan *Mobile* Android. Palembang: Universitas Bina Darma. 2014.
- [2] Sing, N. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization)* Vol. 4, Issue 9, September 2016.
- [3] Nugraha, dkk. Aplikasi Pemesanan Makanan Berbasis *Mobile* Pada Rumah Makan "Lek Nonong". Semarang: Universitas Diponegoro Semarang. 2014.
- [4] Frediyatma, S.Y. Aplikasi Pemesanan Makanan Berbasis Cloud dengan Platform Android. Bali: Universitas Udayana. 2014.
- [5] Wahyujati, D. M. Implementasi Teknologi *Firebase* Pada Aplikasi Pencarian Lokasi *Service* Kamera Berdasarkan Rating Berbasis Android. Yogyakarta: Sekolah Tinggi Manajemen Informatika dan Komputer Akakom. 2017.
- [6] Akbar, dkk. Pembuatan Aplikasi Layanan Pesan Antar Makanan Pada Sistem Operasi Android. Semarang: Universitas Diponegoro Semarang. 2014.
- [7] Google *Firebase*, (2018). [Online]. Tersedia: <https://firebase.google.com/products/>
- [8] Sugiyono. Metode Penelitian Kuantitatif, Kualitatif, dan R&D, Penerbit: Alfabeta, Bandung. 2013.
- [9] Nugroho, Adi. Rekayasa Perangkat Lunak Menggunakan Uml dan Java. Yogyakarta: Andi. 2010.
- [10] *Firebase Realtime Database*, (2018). [Online]. Tersedia: <https://firebase.google.com/products/realtime-database/>.
- [11] Budiaji, W. Skala Pengukuran Dan Jumlah Respon Skala Likert. Serang: Universitas Sultan Ageng Tirtayasa. 2013.