
APLIKASI JASA OJEK PICK-UP BERBASIS LOKASI GEOSPASIAL DAN GOOGLE MAPS DENGAN METODE LOCATION BASED SERVICE

Royhan Restiandi

FAKULTAS TEKNIK UNIVERSITAS TANJUNGPURA PONTIANAK

Email: ft@untan.ac.id

Abstract

Pontianak city as a provincial capital can not be separated from the dense pace of the economy which is driven by much thing, One of them is a transportation service provider. In Pontianak city there is a unloading transportation service known as ojek pick-up service. Ojek pick-up service offers unloading services at low prices. However, the marketing management does not make it easier for people to order their services because there is to many steps to do.

Method that used in the creation of this application is Location Based Service (LBS) method. Meanwhile, design of the applications using the Unified Modelling Language (UML) method. And the application design method uses the waterfall method. Applications are made based on progressive web app and can be used if there is an internet connection.

From the results of the tests that have been carried out using blackbox and questionnaire method, it can be said that the application can run smoothly in assisting service users in ordering and looking for information about pick-up taxi services. And it can be a new alternative choice for pick-up service providers in marketing their pick-up taxi service providers.

Keywords : *pick-up taxi services, service ordering, service search, Location Based Service (LBS), geospacial location, google maps, progressive web app..*

Abstrak

Kota Pontianak sebagai sebuah ibu kota provinsi tidak lepas dari padatnya laju perekonomian yang didorong oleh banyak pihak. Salah satunya adalah penyedia jasa transportasi bongkar muat. Di kota Pontianak terdapat sebuah jasa transportasi bongkar muat yang dikenal sebagai jasa ojek pick-up. Namun cara pemasaran yang dilakukan tidak memudahkan masyarakat dalam memesan jasa milik mereka karena melalui terlalu banyak tahapan.

Metode yang dipakai dalam pembuatan aplikasi ini adalah metode Location Based Service (LBS). Sedangkan dalam perancangan aplikasi menggunakan metode Unified Modelling Language (UML). Dan metode perancangan aplikasi menggunakan metode waterfall. Aplikasi yang dibuat berbasis progressive web app dan dapat digunakan jika terdapat sambungan internet.

Dari hasil pengujian yang telah dilakukan menggunakan metode black box dan user acceptance testing (UAT) dapat dikatakan bahwa aplikasi dapat berjalan dengan lancar dalam membantu pengguna jasa dalam memesan dan mencari informasi mengenai jasa ojek pick-up. Serta dapat menjadi pilihan alternatif baru bagi para penyedia jasa ojek pick-up dalam memasarkan jasa ojek pick-up mereka.

Kata Kunci: Ojek *pick-up*, Pemesanan jasa, Pencarian jasa, Location Based Service (LBS), Google maps, Lokasi geospasial, Progressive web app.



Pendahuluan

“Jasa”. Aktivitas, kemudahan, manfaat, dan sebagainya yang dapat dijual kepada orang lain (konsumen) yang menggunakan atau menikmatinya. (KBBI Daring. 2016). “Ojek”. Sepeda atau sepeda motor yang ditambangkan dengan cara memboncengkan penumpang atau penyewanya (KBBI Daring. 2016). “Pikap (pikup)”. Kendaraan bermotor beroda empat, lebih kecil dari truk, memiliki bak tertutup, digunakan untuk mengangkut barang (KBBI Daring. 2016). Jadi dapat diartikan bahwa jasa ojek *pick-up* adalah sebuah perbuatan yang menawarkan bantuan pengangkutan (bongkar muat) barang maupun orang. “Bongkar muat”. Mengeluarkan dan memasukkan muatan dari atau ke kapal (kereta dan sebagainya) (KBBI Daring. 2016).

Kota Pontianak sebagai sebuah ibukota provinsi tentu memiliki tingkat populasi yang tinggi dan laju ekonomi yang cukup cepat. Faktor laju ekonomi yang cepat ini membuat banyak pihak memerlukan bantuan pihak lainnya agar dapat mengatasi masalah yang dihadapi dengan cepat. Salah satu pihak yang turut membantu ialah jasa ojek *pick-up*. Oleh karena itu saat ini di kota Pontianak banyak sekali pemilik ojek *pick-up* yang tersebar di segala penjuru kota, baik ditengah kota maupun di perbatasan kota. Ojek *pick-up* termasuk salah satu pilihan utama masyarakat untuk melakukan bongkar muat barang seperti properti, bahan material bangunan, belanjaan, bahan pokok, hewan ternak bahkan transportasi rombongan.

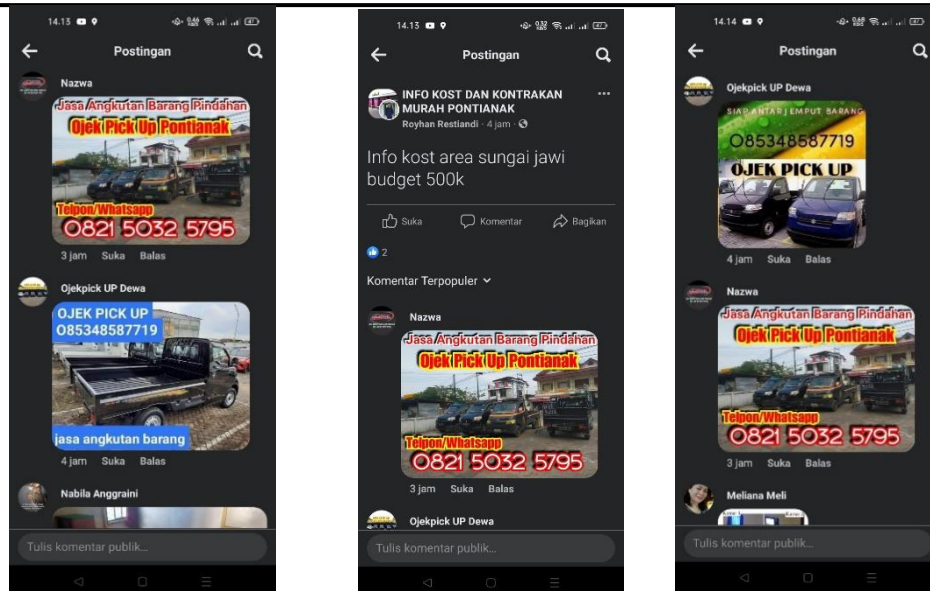
Jasa ojek *pick-up* yang ada di kota Pontianak melakukan pemasaran jasa dengan dua cara. Para pemilik jasa ojek *pick-up* menawarkan jasa mereka dengan cara memarkir mobil *pick-up* mereka di bahu jalan lalu memasang spanduk yang bertuliskan jasa ojek *pick-up* beserta kontak yang bisa dihubungi. Hal ini malah menimbulkan masalah, dimana mobil yang diparkir di bahu jalan dapat membuat kemacetan dan membahayakan pengguna jalan yang lain dan belum lagi memancing tingkat kriminalitas yang dapat merugikan penyedia jasa pada mobil *pick-up* yang mereka biarkan terparkir di tepi jalan tanpa pengawasan. Cara yang lainnya adalah dengan memanfaatkan media sosial dengan membuat postingan guna menawarkan jasa mereka dan menawarkan jasa mereka pada grup-grup yang tidak ada kaitannya sama sekali dengan ojek *pick-up*. Hal ini mengganggu pengguna *media* sosial yang lain karena mereka menilai tidak seharusnya para penyedia jasa ini menawarkan jasa mereka pada grup tersebut. Untuk contoh cara pemasaran yang dilakukan oleh penyedia jasa ojek *pick-up* dapat dilihat pada Gambar 1.1, 1.2 dan 1.3 dibawah ini.



Gambar 1. 1 Mobil *pick-up* parkir memakan bahu jalan (Jl. Tanjungpura)



Gambar 1. 2 Mobil *pick-up* parkir memakan bahu jalan di persimpangan tiga jalan (Jl. Prof. M. Yamin)



“Jasa”. Aktivitas, kemudahan, manfaat, dan sebagainya yang dapat dijual kepada orang lain (konsumen) yang menggunakan atau menikmatinya. (KBBI Daring. 2016). “Ojek”. Sepeda atau sepeda motor yang ditambangkan dengan cara memboncengkan penumpang atau penyewanya (KBBI Daring. 2016). “Pikap (pikup)”. Kendaraan bermotor beroda empat, lebih kecil dari truk, memiliki bak tertutup, digunakan untuk mengangkut barang (KBBI Daring. 2016). Jadi dapat diartikan bahwa jasa ojek *pick-up* adalah sebuah perbuatan yang menawarkan bantuan pengangkutan (bongkar muat) barang maupun orang. “Bongkar muat”. Mengeluarkan dan memasukkan muatan dari atau ke kapal (kereta dan sebagainya) (KBBI Daring. 2016).

Kota Pontianak sebagai sebuah ibukota provinsi tentu memiliki tingkat populasi yang tinggi dan laju ekonomi yang cukup cepat. Faktor laju ekonomi yang cepat ini membuat banyak pihak memerlukan bantuan pihak lainnya agar dapat mengatasi masalah yang dihadapi dengan cepat. Salah satu pihak yang turut membantu ialah jasa ojek *pick-up*. Oleh karena itu saat ini di kota Pontianak banyak sekali pemilik ojek *pick-up* yang tersebar di segala penjuru kota, baik ditengah kota maupun di perbatasan kota. Ojek *pick-up* termasuk salah satu pilihan utama masyarakat untuk melakukan bongkar muat barang seperti properti, bahan material bangunan, belanjaan, bahan pokok, hewan ternak bahkan transportasi rombongan.

Jasa ojek *pick-up* yang ada di kota Pontianak melakukan pemasaran jasa dengan dua cara. Para pemilik jasa ojek *pick-up* menawarkan jasa mereka dengan cara memarkir mobil *pick-up* mereka di bahu jalan lalu memasang spanduk yang bertuliskan jasa ojek *pick-up* beserta kontak yang bisa dihubungi. Hal ini malah menimbulkan masalah, dimana mobil yang diparkir di bahu jalan dapat membuat kemacetan dan membahayakan pengguna jalan yang lain dan belum

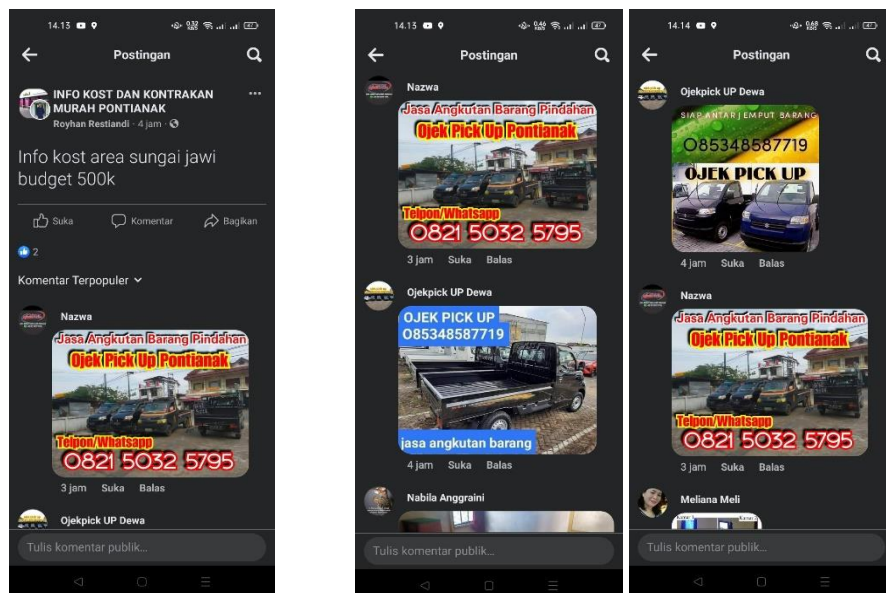
lagi memancing tingkat kriminalitas yang dapat merugikan penyedia jasa pada mobil *pick-up* yang mereka biarkan terparkir di tepi jalan tanpa pengawasan. Cara yang lainnya adalah dengan memanfaatkan media sosial dengan membuat postingan guna menawarkan jasa mereka dan menawarkan jasa mereka pada grup-grup yang tidak ada kaitannya sama sekali dengan ojek *pick-up*. Hal ini mengganggu pengguna *media* sosial yang lain karena mereka menilai tidak seharusnya para penyedia jasa ini menawarkan jasa mereka pada grup tersebut. Untuk contoh cara pemasaran yang dilakukan oleh penyedia jasa ojek *pick-up* dapat dilihat pada Gambar 1.1, 1.2 dan 1.3 dibawah ini.



Gambar 1. 3 Mobil *pick-up* parkir memakan bahu jalan (Jl. Tanjungpura)



Gambar 1. 4 Mobil *pick-up* parkir memakan bahu jalan di persimpangan tiga jalan (Jl. Prof. M. Yamin)



Gambar 1. 5 Penyedia jasa memasarkan jasa pada grup-grup yang tidak ada kaitannya dengan ojek *pick-up*.

Dari cara pemasaran tersebut membuat para pengguna jasa harus berusaha lebih dalam mencari informasi mengenai penyedia jasa ojek *pick-up* yang ada di lokasi mereka. Pengguna jasa harus pergi berkeliling daerah sekitar atau mencari ulang postingan pada sosial media yang telah lalu hanya untuk mengetahui informasi mengenai jasa ojek *pick-up* yang ada. Cara memesan jasa ojek *pick-up* juga dirasa cukup sulit dan harus melalui banyak tahapan seperti mencari informasi terlebih dahulu, melakukan *save* kontak penyedia jasa, lalu menghubungi penyedia jasa dan malah tidak sedikit yang berujung batal memesan dikarenakan biaya yang terlalu mahal dan kriteria *pick-up* yang tersedia tidak sesuai dengan yang dibutuhkan. Disisi lain penyedia jasa ojek *pick-up* yang letaknya berada di perbatasan kota merasa kesulitan untuk mendapatkan pesanan karena posisinya yang tidak terjangkau oleh banyak orang.

Maka dari itu penulis bertujuan untuk membuat sebuah aplikasi yang berfokus untuk mengelola penyedia jasa ojek *pick-up* yang ada di kota Pontianak agar mudah diakses oleh masyarakat untuk mendapatkan informasi dan mengatasi masalah-masalah yang timbul akibat cara pemasaran jasa ojek *pick-up* yang selama ini dilakukan.

Perumusan Masalah

Dari latar belakang yang telah disampaikan diatas. Maka dapat disimpulkan bahwa masalah yang dihadapi dalam penelitian ini adalah:

1. Pengguna jasa kesulitan dalam mencari informasi penyedia jasa ojek *pick-up* yang ada disekitar mereka atau di daerah kota Pontianak.
2. Pengguna jasa merasa waktu yang dibutuhkan untuk memesan jasa ojek *pick-up* cenderung lama dan melewati banyak tahapan.
3. Pengguna jasa kesulitan dalam memilih biaya jasa ojek *pick-up* yang sesuai dengan keinginan mereka.
4. Cara pemasaran yang dilakukan oleh beberapa penyedia jasa merugikan masyarakat dan tidak memudahkan pengguna dalam memesan jasa karena banyaknya tahapan yang harus dilalui.

Beberapa penyedia jasa yang terletak di perbatasan kota merasa kesulitan dalam mendapatkan pesanan dikarenakan posisi mereka yang susah terjangkau pengguna.

Metode Penelitian

Bahan penelitian berupa data dari jasa-jasa ojek *pick-up* yang didapat dari form pendaftaran yang diisikan oleh para pemilik jasa yang mendaftar ke dalam aplikasi Pickapps. Adapun data-data yang diperlukan antara lain mengenai identitas pemilik jasa ojek *pick-up*, kontak, jenis dan tipe jasa pengantaran yang dilakukan, dan lokasi jasa ojek *pick-up* serta data-data lainnya yang dianggap perlu.

Hasil dan Pembahasan

Berikut ini adalah hasil penelitian yang telah dibuat berdasarkan bab sebelumnya. Berdasarkan hasil analisis dan pengujian terhadap aplikasi Pickapps *customerr* (<https://pickapps.pandadev.my.id/users>) dan aplikasi Pickapps *driver* (<https://pickapps.pandadev.my.id/drivers>) dapat disimpulkan bahwa aplikasi ini dapat memberikan informasi seputar jasa layanan ojek *pick-up* yang ada di kota Pontianak. Informasi yang diberikan nama jasa ojek *pick-up*, jenis mobil *pick-up*, jarak jasa ojek *pick-up* dengan posisi

customer saat ini, harga yang ditawarkan, foto mobil *pick-up* dan profil dan *rating* dari para pemilik jasa ojek *pick-up*. Selain memberikan informasi seputar jasa ojek *pick-up*, aplikasi Pickapps juga dapat membantu pengguna untuk memesan jasa ojek *pick-up* yang diinginkan dengan taapan yang singkat. Berikut adalah penuturan mengenai hasil-hasil dari penelitian yang dilakukan.

Tampilan Antarmuka Halaman *Splash Screen* (*Customer & Driver Apps*)

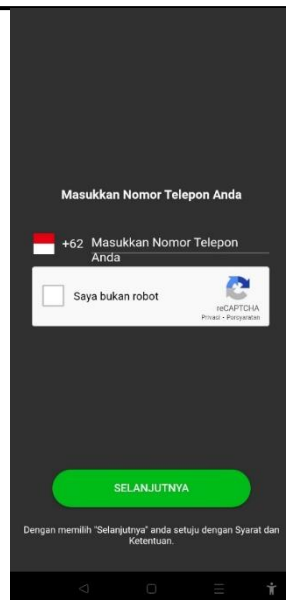
Antarmuka *splash screen* adalah halaman awal yang akan muncul sesaat setelah *customer* membuka aplikasi Pickapps. *Splash screen* hanya berisi sebuah *logo* dari aplikasi Pickapps dan hanya berlangsung selama 2 detik yang selanjutnya akan berganti menjadi halaman *input* nomor *handphone*. Untuk tampilan antarmuka halaman *splash screen* dapat dilihat pada gambar 4.1 berikut.



Gambar 4. 1 Tampilan Antarmuka *Splash Screen*

Tampilan Antarmuka Halaman *Input Nomor Handphone* (*Customer Apps*)

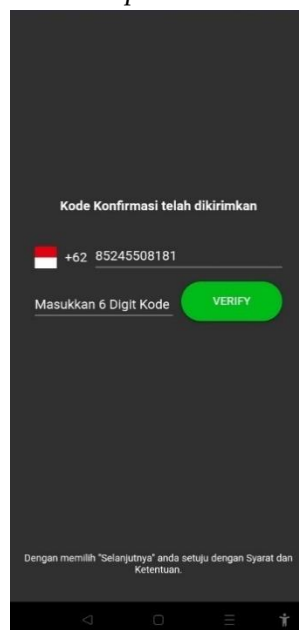
Antarmuka halaman *input* nomor *handphone* adalah halaman yang terbuka 2 detik setelah antarmuka *splash screen* berjalan. Halaman ini berfungsi untuk halaman *login customer* dimana *input* nomor *handphone* dikategorikan dengan *username*. *Customer* memasukkan nomor *handphone* pada kolom yang telah disiapkan dan melanjutkan ke halaman *input* kode OTP. Untuk tampilan antarmuka halaman *input* nomor *handphone* dapat dilihat pada gambar 4.2 dibawah ini.



Gambar 4. 2 Tampilan Antarmuka Halaman *Input Nomor Handphone*

Tampilan Antarmuka Halaman *Input Kode OTP (Customer Apps)*

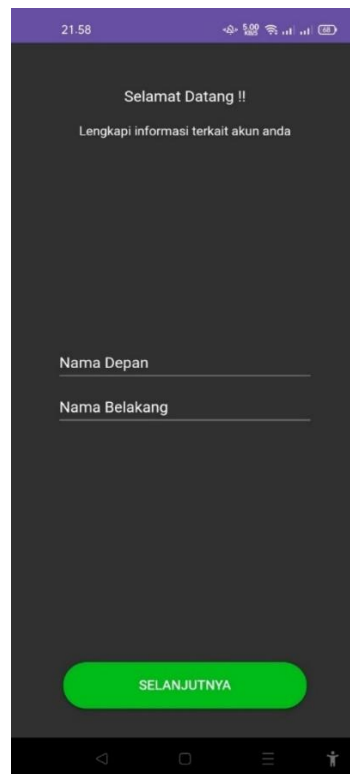
Antarmuka halaman *input* kode OTP adalah halaman lanjutan dari halaman *input* nomor *handphone*. Jika *input* nomor *handphone* dikategorikan sebagai *username*, maka *input* OTP ini dapat dikategorikan sebagai *password* untuk akses masuk kedalam aplikasi. Fitur yang digunakan untuk mendapatkan dalam aplikasi Pickapps adalah milik firebase sehingga kode OTP yang dikirimkan nantinya akan masuk via SMS pada nomor *handphone* yang dimasukkan tadi. Setelah memasukkan kode OTP dengan benar *customer* akan diarahkan ke halaman berikutnya sesuai kondisi yang ada. Untuk tampilan halaman *input* kode OTP dapat dilihat pada gambar 4.3.



Gambar 4. 3 Tampilan Antarmuka Halaman *Input Kode OTP (Customer Apps)*

Tampilan Antarmuka Halaman *Input Data Diri (Customer Apps)*

Antarmuka halaman *input data diri* adalah halaman lanjutan setelah *customer* menginputkan kode OTP. Halaman ini hanya terbuka jika *customer* sebelumnya tidak pernah *login* kedalam aplikasi. Pada halaman ini *customer* diminta mengisi *data diri* sebagai keperluan untuk melengkapi *data* profil *customer*. Untuk tampilan antarmuka halaman *input data diri* pada *customer apps* dapat dilihat pada gambar 4.4 berikut.



21.58

Selamat Datang !!

Lengkapi informasi terkait akun anda

Nama Depan

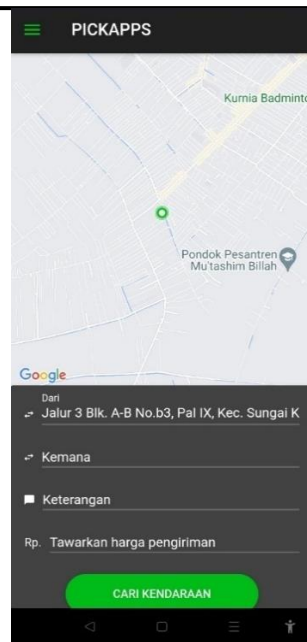
Nama Belakang

SELANJUTNYA

Gambar 4. 4 Tampilan Antarmuka Halaman *Input Data Diri (Customer Apps)*

Tampilan Antarmuka Halaman Beranda (*Customer Apps*)

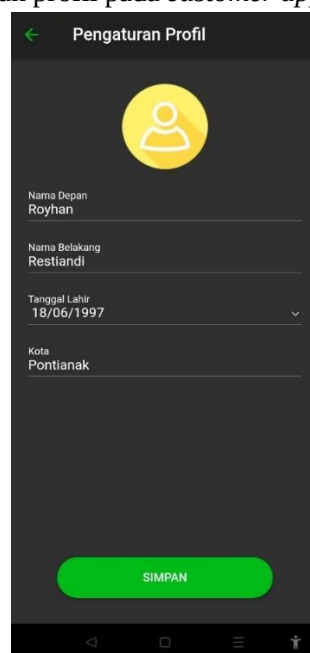
Antarmuka halaman beranda ini adalah halaman beranda yang dikhususkan untuk *customer*. Halaman ini adalah pusat dari segala kegiatan *customer* dalam memesan jasa ojek *pick-up*. Pada halaman beranda terdapat sebuah *maps* yang didapatkan melalui layanan google maps API dan titik lokasi *customer* yang terbaca lewat GPS perangkat *customer* serta ditampilkan dalam bentuk data *geolocation/geospasial*. *Customer* dapat melihat *data* lokasi diri sendiri, beberapa kolom *input* yang digunakan untuk mengisi data ketika ingin *order*, dapat mengakses *button-button* yang mendukung *customer* dalam melakukan *order* jasa ojek *pick-up* dan lainnya. Pada kolom input lokasi pengambilan barang akan otomatis terisi berdasarkan lokasi *customer* saat ini lewat pembacaan GPS. Layanan yang digunakan pada kolom ini adalah *geocoding* dari google maps API yang berfungsi merubah titik koordinat menjadi nama alamat dan sebaliknya. Hal yang sama juga berlaku pada kolom input lokasi pengambilan. Untuk tampilan antarmuka halaman beranda pada *customer apps* ini dapat dilihat pada gambar 4.5 dibawah.



Gambar 4. 5 Tampilan Antarmuka Halaman Beranda (*Customer Apps*)

Tampilan Antarmuka Halaman Profil (*Customer Apps*)

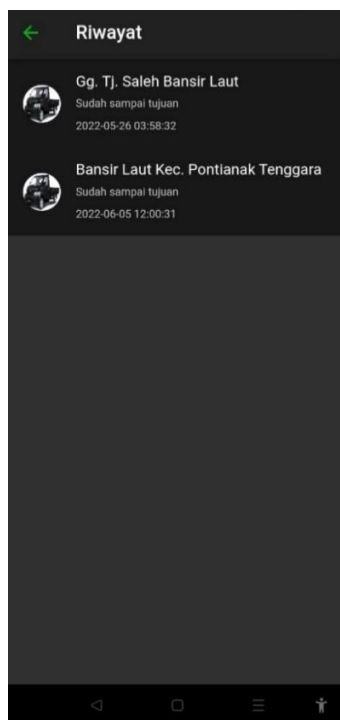
Antarmuka halaman profil adalah halaman dimana informasi *data customer* ditampilkan. *Data customer* yang ditampilkan berupa *avatar*, nama depan dan belakang *customer*, tanggal lahir dan kota. Pada halaman ini juga *customer* dapat mengubah *data* dirinya pada kolom yang telah disediakan. Untuk tampilan halaman profil pada *customer apps* ini dapat dilihat pada gambar 4.6.



Gambar 4. 6 Tampilan Antarmuka Halaman Profil (*Customer Apps*)

Tampilan Antarmuka Halaman Riwayat (*Customer Apps*)

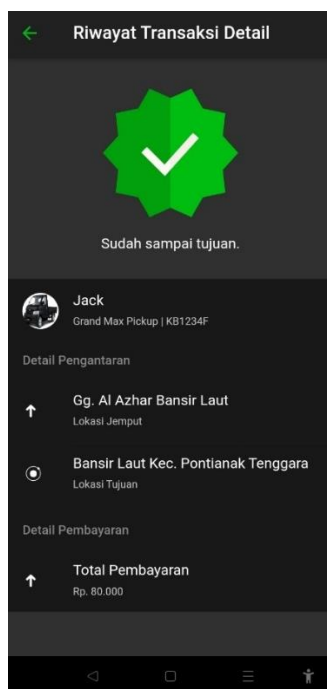
Antarmuka halaman riwayat ini adalah halaman yang berisi tentang *list data-data* dari orderan yang pernah *customer* lakukan. *Customer* dapat memilih salah satu *list* untuk melihat *detail* dari data tersebut. Untuk tampilan halaman riwayat pada *customer apps* dapat dilihat pada gambar 4.7 berikut.



Gambar 4. 7 Tampilan Antarmuka Halaman Riwayat (*Customer Apps*)

Tampilan Antarmuka Halaman Riwayat *Detail* (*Customer Apps*)

Antarmuka halaman ini adalah lanjutan dari halaman riwayat. Disini menampilkan data yang lebih lengkap dari salah satu riwayat yang dipilih oleh *customer* saat ada di halaman riwayat. Tampilan antarmuka halaman riwayat *detail* pada *customer apps* dapat dilihat pada gambar 4.8 dibawah ini.

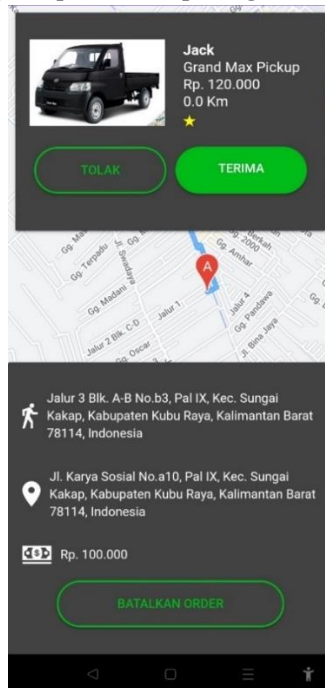


Gambar 4. 8 Tampilan Halaman Riwayat *Detail* (*Customer Apps*)

Tampilan Antarmuka Halaman Tunggu Tawaran Masuk (*Customer Apps*)

Halaman ini adalah halaman yang muncul ketika *customer* telah selesai mengisi data *order* dan menekan *button* “cari kendaraan”. Pada halaman ini nantinya *customer* akan menunggu untuk mendapatkan tawaran harga dari para penyedia jasa ojek *pick-up* yang ingin mengambil orderan

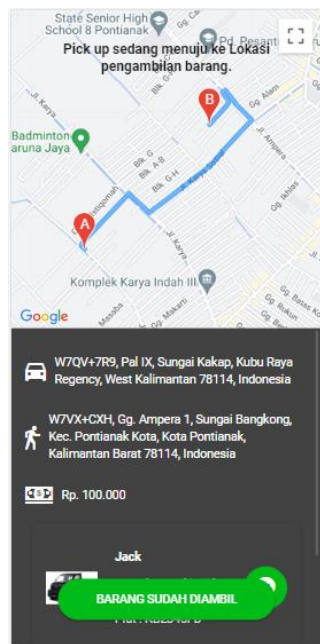
tersebut. Tawaran yang masuk akan tampil berupa list yang menampilkan data dari penyedia jasa, harga yang mereka tawarkan dan jarak penyedia jasa dari titik pengambilan barang. List ini nantinya akan diurutkan berdasarkan jarak terdekat dari penyedia jasa dengan lokasi pengambilan barang. Jadi metode *location based service* (LBS) akan diterapkan dalam mencari jarak terdekat menggunakan *haversine formula*. Selain tawaran, *customer* juga dapat melihat sebuah *maps* dari layanan google maps API yang berisi lokasi titik pengambilan dan pengantaran barang sehingga akan ditampilkan dalam bentuk *geolocation/geospasial*. Ada juga data yang sebelumnya diinputkan oleh *customer* pada halaman beranda. Untuk tampilan antarmuka halaman tunggu tawaran masuk pada *customer apps* dapat dilihat pada gambar 4.9.



Gambar 4. 9 Tampilan Halaman Tunggu Tawaran Masuk (*Customer Apps*)

Tampilan Antarmuka Halaman Tunggu *Driver* ke Lokasi Pengambilan (*Customer Apps*)

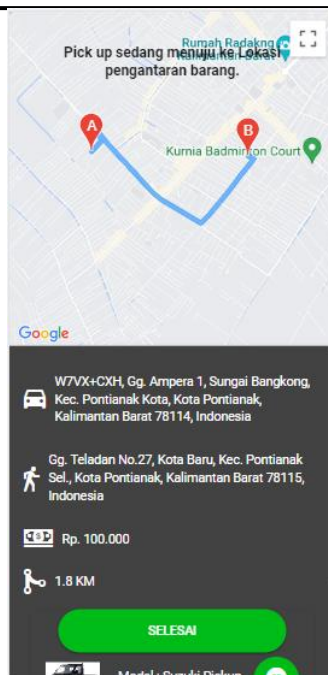
Antarmuka halaman ini adalah halaman lanjutan dalam proses *order ojek pick-up*. Setelah *customer* memutuskan *driver* mana yang dipilih, maka *customer* akan dibawa ke halaman ini. Pada halaman ini *customer* dapat melihat sebuah data *geolocation/geospasial* berupa sebuah *maps* dari layanan google maps API yang berisi titik lokasi pengambilan barang dan titik penyedia jasa yang sedang menjalankan orderan, titik lokasi *driver* disini adalah posisi terakhir *driver*. Masing-masing titik akan saling terhubung menjadi sebuah rute memanfaatkan layanan google direction API. Penggunaan layanan google direction API maka maps akan menampilkan petunjuk arah untuk beberapa moda transportasi. Selain itu akan ada *data* berupa nama dari lokasi titik tersebut dan harga yang telah disepakati. Selain itu ada juga profil dari *driver* yang bersangkutan. Terdapat *button* yang dapat digunakan oleh *customer* untuk menghubungi *driver* lewat Whatsapp. Pada desain awal aplikasi Pickapps akan menggunakan pelacakan posisi *driver* secara *real-time* dalam menjalankan orderan. Namun dalam eksekusinya tidak dapat dilakukan karena terkendala layanan firebase yang dibatasi. Sehingga *direct* ke halaman selanjutnya jika posisi *driver* sudah tiba ke titik pengambilan tidak dalam dilakukan. Lalu dibuatlah sebuah *manual button* untuk mengatasi hal tersebut. *Manual button* digunakan oleh *customer* berpindah ke halaman selanjutnya. Tampilan antarmuka tunggu *driver* ke lokasi pengambilan barang pada *customer apps* dapat dilihat pada gambar 4.10 berikut.



Gambar 4. 10 Tampilan Antarmuka Halaman Tunggu *Driver* ke Lokasi Pengambilan Barang (*Customer Apps*)

Tampilan Antarmuka Halaman Tunggu *Driver* ke Lokasi Pengantaran (*Customer Apps*)

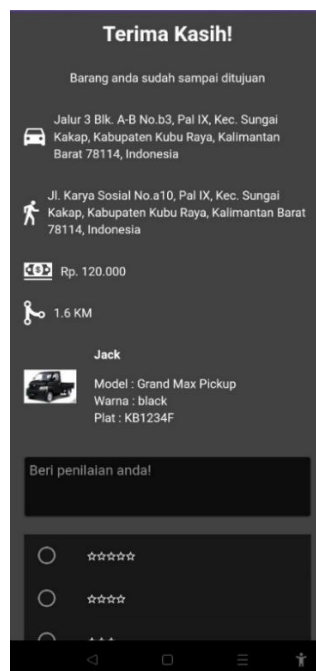
Hampir sama dengan antarmuka halaman tunggu *driver* ke lokasi pengambilan diatas. Antarmuka halaman ini juga berisi data geolocation/geospasial berupa *maps* dari layanan google maps API dan data-data lainnya. Bedanya pada antarmuka ini *maps* yang ada menampilkan lokasi pengantaran barang dari titik pengambilan ke titik pengantaran beserta rute yang lalui, rute yang didapat pada halaman ini juga menggunakan layanan google direction API. Selain itu pada halaman ini juga menampilkan data jarak yang ditempuh *driver* dari titik pengambilan ke titik pengantaran. Data perhitungan jarak ini didapatkan lewat layanan google *distance matrix* API, sehingga jarak yang didapat bukanlah data jarak garis lurus melainkan data jarak sesungguhnya dari rute yang dilalui dan tentu saja jarak yang didapat adalah jarak terdekat yang bisa dilalui. Untuk antarmuka halaman tunggu *driver* ke lokasi pengantaran barang pada *customer apps* dapat dilihat pada gambar 4.11 dibawah.



Gambar 4. 11 Tampilan Antarmuka Halaman Tunggu *Driver* ke Lokasi Pengantaran Barang (*Customer Apps*)

Tampilan Antarmuka Halaman Beri *Rating* (*Customer Apps*)

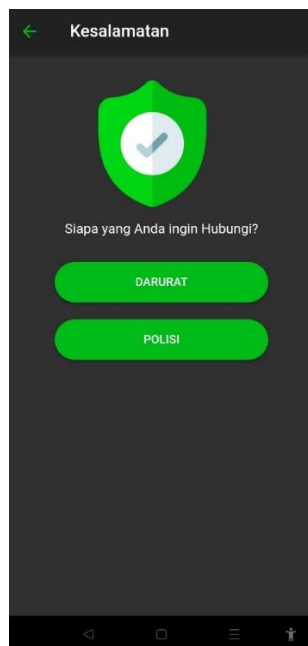
Halaman ini adalah halaman terakhir dalam sesi *order* jasa ojek *pick-up* yang dilakukan oleh *customer*. Berisi tentang ucapan terima kasih karena telah menggunakan jasa dan aplikasi Pickapps, lokasi titik pengambilan dan pengantaran barang, harga yang dibayarkan, jarak tempuh, profil milik *driver*, kolom beri komentar atas kinerja *driver* dan skala *rating* yang dapat dipilih untuk memberikan penilaian. Setelah *customer* memberikan penilai dan *rating* kepada *driver* yang bersangkutan maka data tersebut akan disimpan kedalam *database* tabel *rating*. Tampilan antarmuka dapat dilihat pada gambar 4.12.



Gambar 4. 12 Tampilan Antarmuka Halaman Beri *Rating* (*Customer Apps*)

Tampilan Antarmuka Halaman Keselamatan (*Customer Apps*)

Antarmuka halaman ini berisi kontak darurat yang dapat digunakan *customer* saat terjadi hal yang tidak diinginkan. Kontak tersebut dibagi menjadi dua yaitu darurat seperti ambulans dan damkar serta polisi. Untuk tampilan antarmuka halaman keselamatan pada *customer apps* dapat dilihat pada gambar 4.13 berikut.



Gambar 4. 13 Tampilan Antarmuka Halaman Keselamatan (*Customer Apps*)

Tampilan Antarmuka Halaman Bantuan (*Customer Apps*)

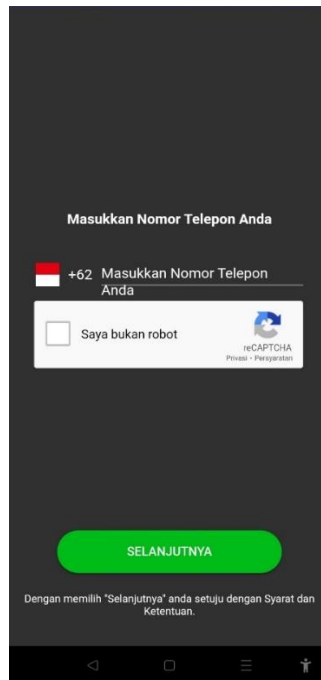
Antarmuka halaman ini berisi tentang informasi yang dapat *customer* gunakan jika mengalami kendala saat menggunakan aplikasi Pickapps. Informasi yang diberikan antara lain *tutorial* permintaan, titik A-B, tentukan harga, pilih pengemudi dan *contact person* aplikasi Pickapps. Untuk tampilan antarmuka halaman bantuan pada *customer apps* dapat dilihat pada gambar 4.14 dibawah ini.



Gambar 4. 14 Tampilan Antarmuka Halaman Bantuan (*Customer Apps*)

Tampilan Antarmuka Halaman Input Nomor Handphone (*Driver Apps*)

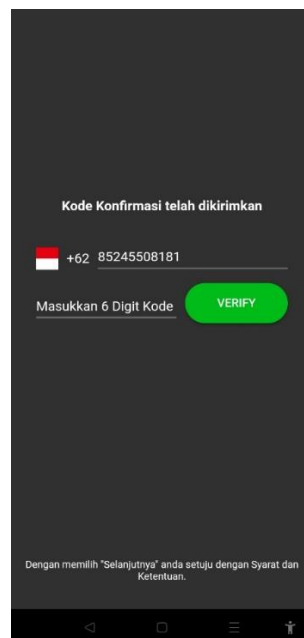
Antarmuka halaman *input* nomor *handphone* pada *driver apps* ini sama seperti yang dimiliki oleh *customer apps* sehingga penjelasannya dianggap sama. Untuk tampilan antarmuka dapat dilihat pada gambar 4.15.



Gambar 4. 15 Tampilan Antarmuka Halaman Input Nomor *Handphone* (*Driver*)

Tampilan Antarmuka Halaman *Input* Kode OTP (*Driver Apps*)

Tampilan antarmuka halaman ini sama dengan yang ada pada *customer apps*. Sehingga penjelasannya juga dianggap sama. Pada halaman ini juga fitur OTP yang digunakan adalah milik firebase. Untuk tampilan antarmuka halaman *input* kode OTP pada *driver apps* dapat dilihat pada gambar 4.16 dibawah ini.

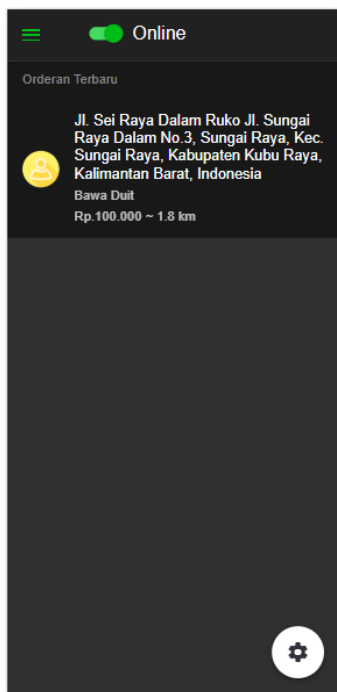


Gambar 4. 16 Tampilan Antarmuka Halaman *Input* Kode OTP (*Driver*)

Tampilan Antarmuka Halaman Beranda (*Driver Apps*)

Antarmuka halaman beranda *driver apps* hanya berisi *list* orderan masuk dari para *customer* yang mengajukan *order*. *List* pada halaman ini akan diurutkan berdasarkan jarak terdekat lokasi pengambilan barang dari *customer*. Sehingga disini akan diterapkan metode *location based*

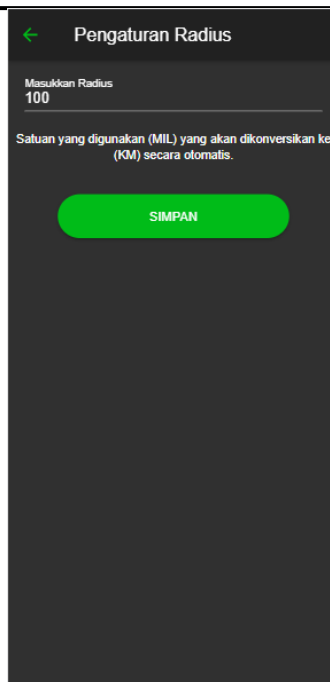
service (LBS) menggunakan formula haversine dalam menghitung jarak dan mencari jarak terdekat. Disini *driver* dapat memilih orderan mana yang ingin mereka ambil. Pertimbangannya adalah jarak dari posisi *driver* saat ini berada ke tempat pengambilan barang, jarak pengantaran dan/atau harga yang ditawarkan. Pada halaman ini juga terdapat sebutan *swipe button* yang digunakan *driver* untuk mengaktifkan atau menonaktifkan akun. Tampilan antarmuka halaman dapat dilihat pada gambar 4.17 berikut ini.



Gambar 4. 17 Tampilan Antarmuka Halaman Beranda (*Driver Apps*)

Tampilan Antarmuka Halaman Atur Radius (Driver)

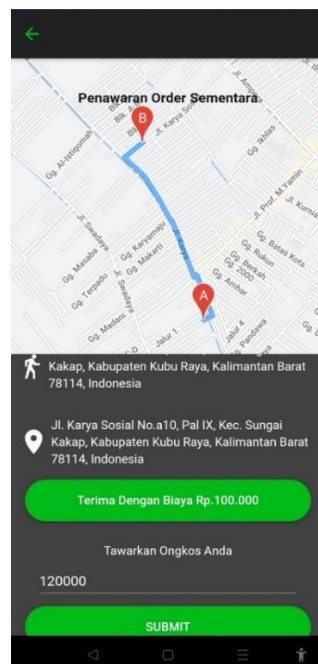
Pada halaman ini driver dapat mengatur radius jarak orderan dari customer yang dapat mereka terima pada perangkat mereka. Untuk penentuan radius ini menggunakan metode *haversine formula*. Defaultnya satuan jarak yang digunakan pada algoritma ini adalah mil maka dari itu dilakukalah konversi menjadi kilometer. Untuk antarmuka halaman atur radius oleh driver dapat dilihat pada Gambar 4.18.



Gambar 4. 18 Tampilan Antarmuka Halaman Atur Radius (Driver)

Tampilan Antarmuka Halaman Tawaran Harga (*Driver Apps*)

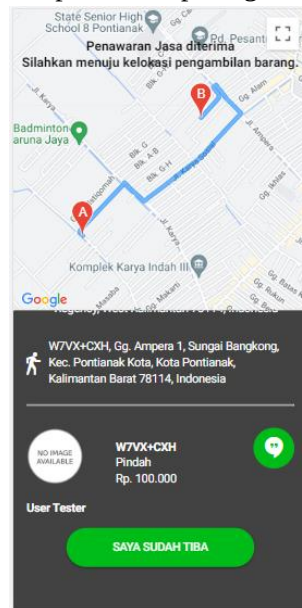
Halaman ini berisi data orderan yang dikirimkan oleh *customer*. Terdapat *button* “terima dengan harga sekian” yang dapat digunakan *driver* jika ingin mengambil orderan dengan cepat, atau *button* “tawarkan harga” yang dapat digunakan *driver* untuk menawarkan harga lain jika dirasa harga yang diminta oleh *customer* terlalu rendah. Selain itu juga terdapat sebuah data *geolocation/geospasial* berupa *maps* dari layanan google maps API yang menampilkan lokasi pengambilan dan pengantaran dari pesanan yang dilakukan oleh *customer* yang bersangkutan, serta sebuah rute yang didapatkan melalui layanan google direction API. Untuk tampilan antarmuka halaman terima / tawarkan harga pada *driver apps* dapat dilihat pada gambar 4.19 dibawah ini.



Gambar 4. 19 Tampilan Antarmuka Halaman Tawaran Harga (*Driver Apps*)

Tampilan Antarmuka Halaman Menuju Lokasi Pengambilan (*Driver Apps*)

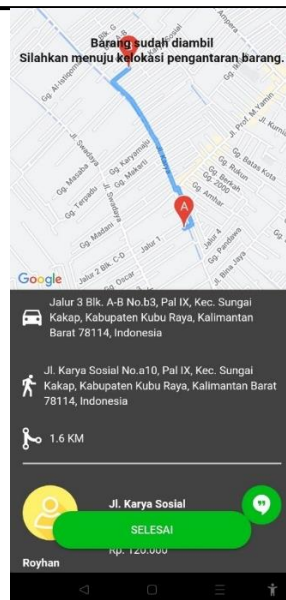
Secara keseluruhan tampilan halaman ini hampir sama seperti yang dimiliki oleh *customer apps*, menampilkan data *geolocation/geospasial* berupa *maps* yang didapatkan dari layanan google maps API beserta rute yang dilewati dari layanan google direction API. Cuma yang membedakan adalah pada *maps* tertulis notifikasi bahwa tawaran yang diajukan diterima oleh *customer* dan *driver* dipersilahkan untuk menjalankan orderan, selain itu juga profil yang ditampilkan adalah milik *customer*. Pada desain awal aplikasi Pickapps akan menggunakan pelacakan posisi *driver* secara *real-time* dalam menjalankan orderan. Namun dalam eksekusinya tidak dapat dilakukan karena terkendala layanan firebase yang dibatasi. Sehingga *direct* ke halaman selanjutnya jika posisi *driver* sudah tiba ke titik pengambilan tidak dalam dilakukan. Lalu dibuatlah sebuah *manual button* untuk mengatasi hal tersebut. *Manual button* digunakan oleh *driver* berpindah ke halaman selanjutnya. Antarmuka halaman ini dapat dilihat pada gambar 4.20.



Gambar 4. 20 Tampilan Antarmuka Halaman Menuju Lokasi Pengambilan Barang (*Driver Apps*)

Tampilan Antarmuka Halaman Menuju Lokasi Pengantaran (*Driver Apps*)

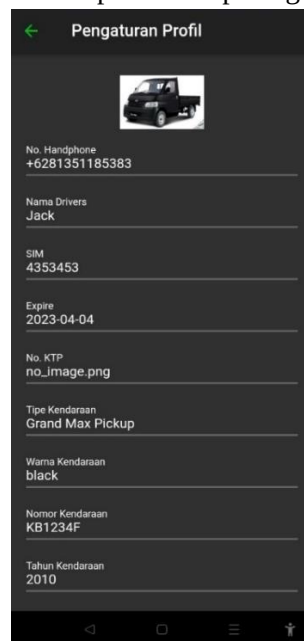
Tampilan antarmuka halaman ini juga hampir sama seperti yang dimiliki *customer apps*. Lagi-lagi yang membedakan adalah notifikasi pada *maps* dan profil yang ditampilkan. Untuk tampilan antarmuka menuju lokasi pengantaran barang pada *driver apps* dapat dilihat pada gambar 4.21 berikut.



Gambar 4. 21 Tampilan Antarmuka Halaman Menuju Lokasi Pengantaran Barang (*Driver Apps*)

Tampilan Antarmuka Halaman Profil (*Driver Apps*)

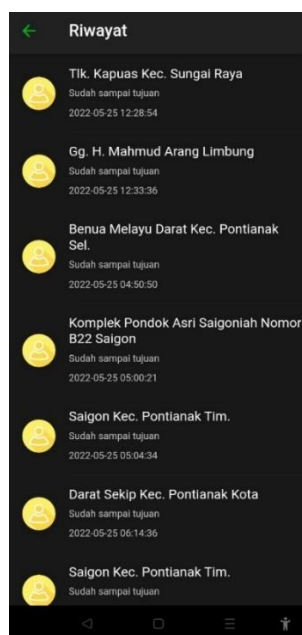
Pada antarmuka halaman profil terdapat *data-data* dari pemilik jasa ojek *pick-up*. Data yang ditampilkan antara lain *avatar*, nomor *handphone driver*, nama pemilik, nomor SIM, tanggal *expired SIM*, nomor KTP, tipe kendaraan, warna kendaraan, nomor kendaraan dan tahun kendaraan. Tampilan antarmuka ini dan dapat dilihat pada gambar 4.22 dibawah ini.



Gambar 4. 22 Tampilan Antarmuka Halaman Profil (*Driver Apps*)

Tampilan Antarmuka Halaman Riwayat (*Driver Apps*)

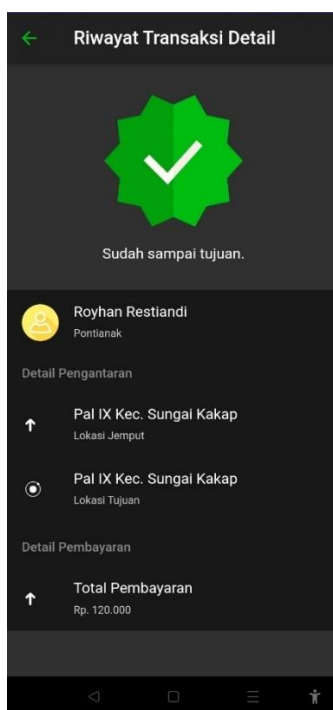
Tampilan antarmuka halaman ini sama dengan yang dimiliki oleh *customer apps*, sehingga dianggap penjelasannya sama. Untuk tampilan antarmuka halaman riwayat pada *driver apps* dapat dilihat pada gambar 4.23 berikut.



Gambar 4. 23 Tampilan Antarmuka Halaman Riwayat (*Driver Apps*)

Tampilan Antarmuka Halaman Riwayat *Detail* (*Driver Apps*)

Tampilan antarmuka halaman ini hampir sama dengan yang dimiliki oleh *customer apps*, cuma yang membedakan adalah profil yang ditampilkan adalah profil milik *customer* yang melakukan *order*. Untuk tampilan antarmuka halaman riwayat *detail* pada *driver apps* dapat dilihat pada gambar 4.24.



Gambar 4. 24 Tampilan Antarmuka Halaman Riwayat *Detail* (*Driver Apps*)

Tampilan Antarmuka Halaman Keselamatan (*Driver Apps*)

Secara garis besar isi dari tampilan halaman keselamatan pada *driver apps* sama seperti yang ada pada *customer apps*. Untuk tampilan antarmuka halaman keselamatan pada *driver apps* dapat dilihat pada gambar 4.25 dibawah ini.



Gambar 4. 25 Tampilan Antarmuka Halaman Keselamatan (*Driver Apps*)

Tampilan Antarmuka Halaman Bantuan (*Driver Apps*)

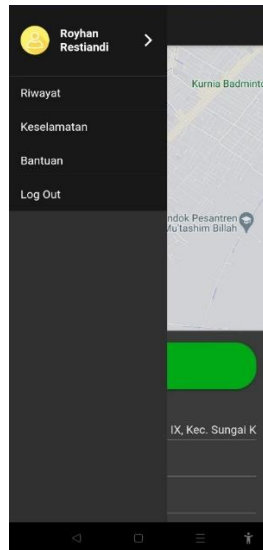
Tampilan antarmuka ini juga sama seperti yang ada pada *customer apps*. Untuk tampilan antarmuka halaman bantuan pada *driver apps* dapat dilihat pada gambar 4.26 berikut.



Gambar 4. 26 Tampilan Antarmuka Halaman Bantuan (*Driver Apps*)

Tampilan Antarmuka Drawer Menu (*Customer & Driver Apps*)

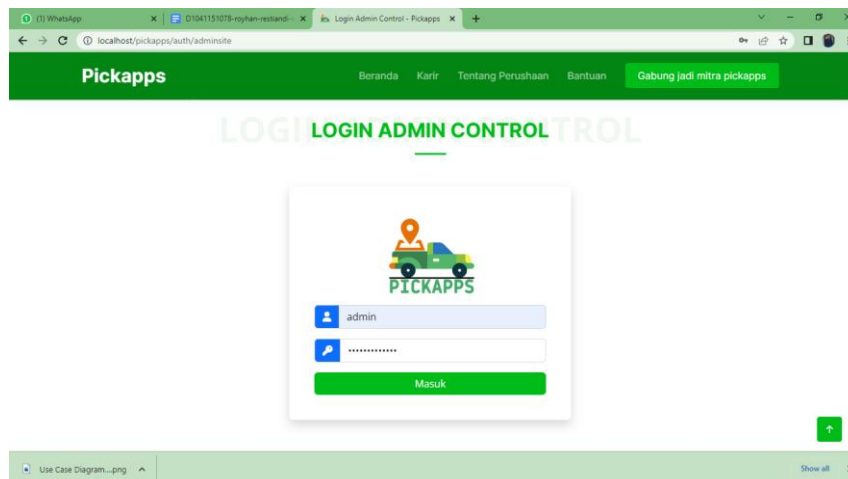
Pada antarmuka *drawer menu* berisi *list menu* yang bisa digunakan oleh *customer* untuk berpindah antar halaman. Isi dari *list menu* tersebut antara lain profile, riwayat, keselamatan, bantuan dan *logout*. Pada dasarnya *drawer menu* milik *customer* dan *driver* memiliki tampilan yang sama. Untuk tampilan antarmuka *drawer menu* dapat dilihat pada gambar 4.27.



Gambar 4. 27 Tampilan Antarmuka Drawer Menu (*Customer & Driver Apps*)

Tampilan Antarmuka Halaman Login (Admin)

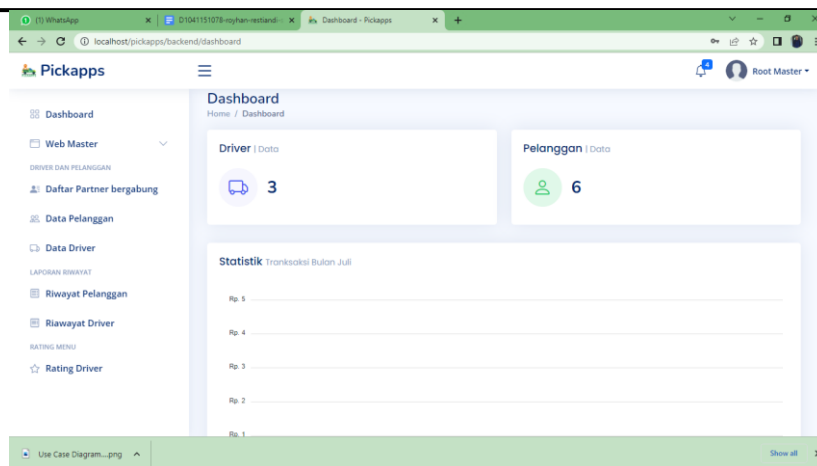
Halaman ini berisi kolom *username* dan *password* yang dapat diisi oleh *admin* sebagai syarat untuk *login* kedalam *dashboard admin*. Untuk antarmuka halaman *login admin* dapat dilihat pada gambar 4.28 berikut.



Gambar 4. 28 Antarmuka Halaman Login (Admin)

Tampilan Antarmuka Halaman Beranda Dashboard (Admin)

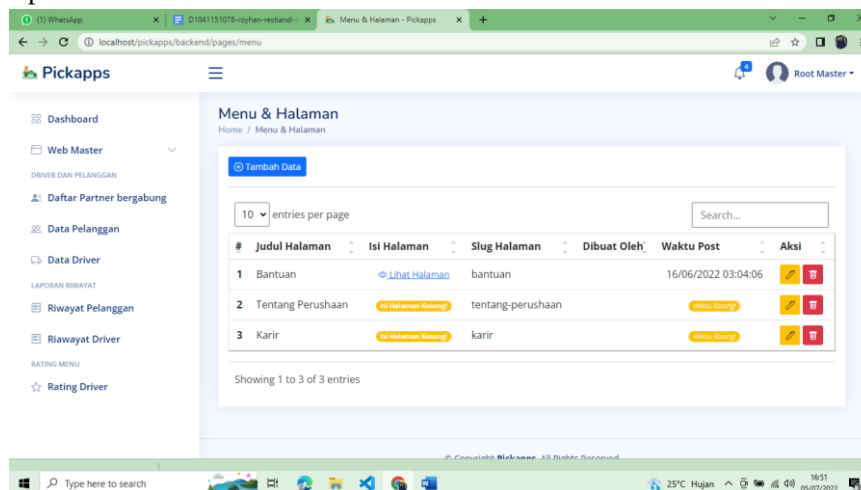
Pada halaman beranda *dashboard admin* berisi *menu-menu* dari halaman yang ada di dalam *dashboard*. Mulai dari halaman kelola *website*, pendaftaran *driver*, data *customer*, data *driver*, riwayat *customer*, riwayat *driver* dan rating *driver*. Disini juga terdapat jumlah data dari *driver* dan *customer* yang ada di aplikasi *Pickapps*. Untuk tampilan antarmuka halaman beranda *dashboard admin* dapat dilihat pada Gambar 4.29 dibawah ini.



Gambar 4. 29 Tampilan Antarmuka Halaman Beranda *Dashboard (Admin)*

Tampilan Antarmuka Halaman Kelola Halaman Website (Admin)

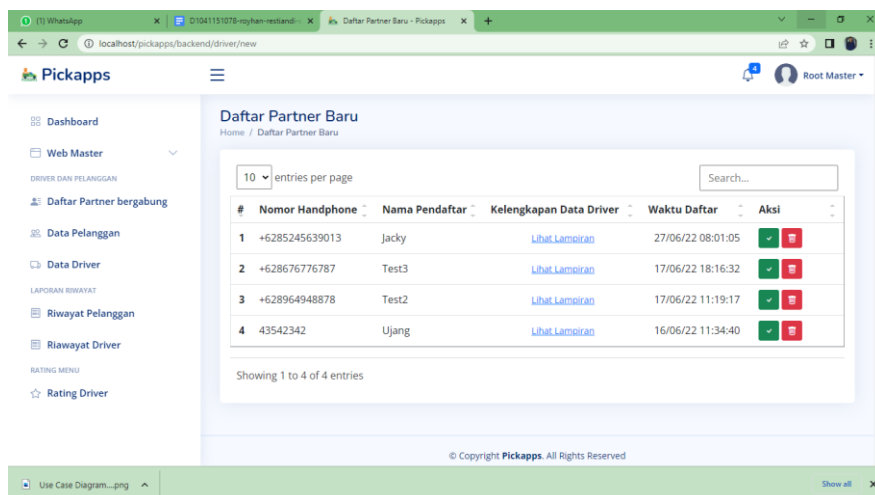
Pada halaman ini *admin* dapat mengatur halaman *menu* yang ada pada *website* Pickapps. *Admin* dapat menambah atau menghapus *menu* tersebut sesuai kebutuhan. Untuk tampilan antarmuka dapat dilihat pada Gambar 4.30.



Gambar 4. 30 Tampilan Antarmuka Halaman Kelola Halaman Website (Admin)

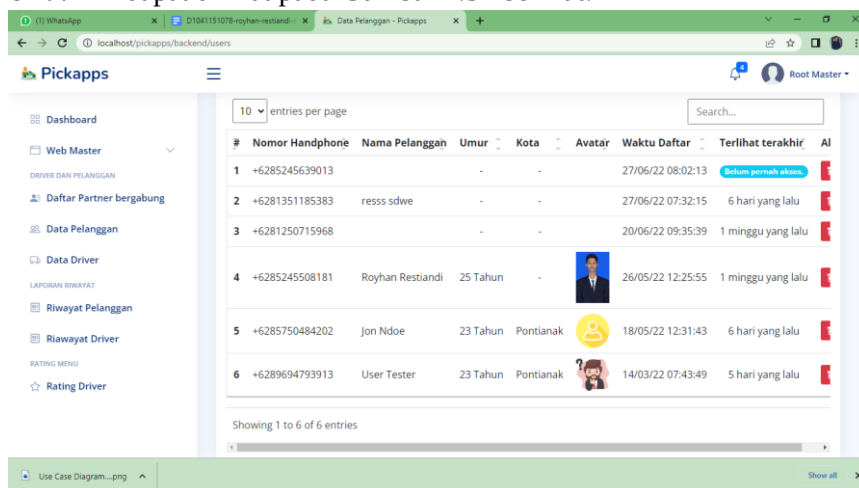
Tampilan Antarmuka Halaman Kelola Pendaftaran Driver (Admin)

Pada halaman ini terdapat *data-data* dari masyarakat yang mendaftar sebagai *driver* pada aplikasi Pickapps. *Admin* dapat memilih untuk menolak atau menerima pendaftaran tersebut sesuai ketentuan yang sudah diatur. Untuk tampilan antarmuka halaman kelola pendaftaran *driver* oleh *admin* dapat dilihat pada gambar 4.31 dibawah ini.



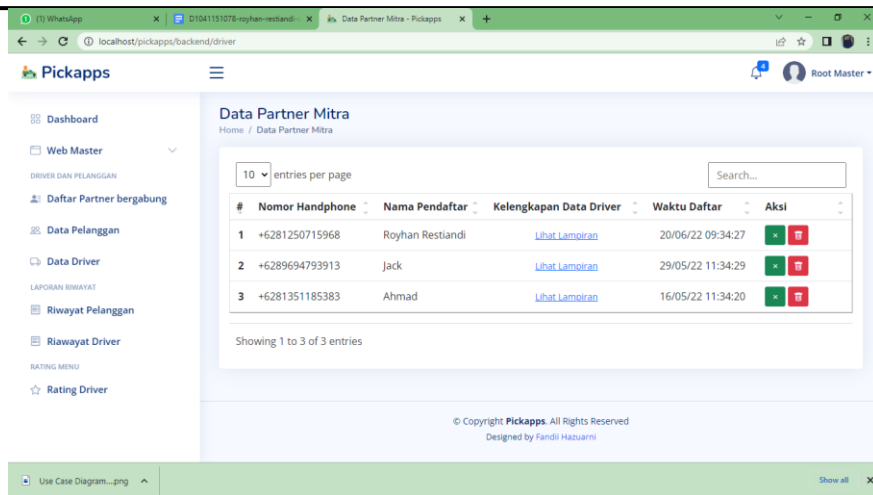
Gambar 4. 31 Tampilan Antarmuka Halaman Kelola Pendaftaran *Driver* (*Admin*)
Tampilan Antarmuka Halaman Kelola Data *Customer* (*Admin*)

Halaman ini berisi data dari semua *customer* yang ada pada aplikasi Pickapps. Pada halaman ini *admin* dapat melihat dan menghapus data *customer* yang ada. Untuk tampilan halaman kelola data *customer* oleh *admin* dapat dilihat pada Gambar 4.32 berikut.



Gambar 4. 32 Tampilan Antarmuka Halaman Kelola Data *Customer* (*Admin*)
Tampilan Antarmuka Halaman Kelola Data *Driver* (*Admin*)

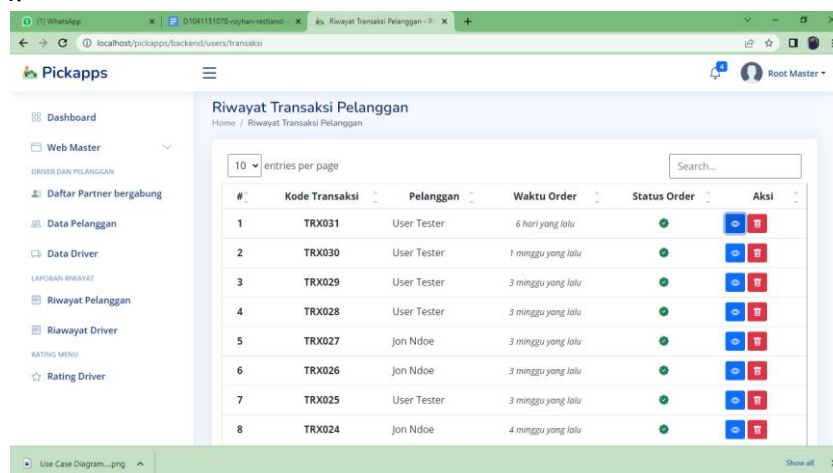
Pada halaman ini berisi data dari para *driver* yang ada pada aplikasi Pickapps. Pada halaman ini *admin* dapat menghapus akun *driver* manapun, selain menghapus *admin* juga dapat menonaktifkan sementara akun *driver* jika pada kasus tertentu hal ini diperlukan. Untuk tampilan antarmuka halaman kelola data *driver* oleh *admin* dapat dilihat pada Gambar 4.33.



Gambar 4. 33 Tampilan Antarmuka Halaman Kelola Data Driver (Admin)

Tampilan Antarmuka Halaman Riwayat Costumer (Admin)

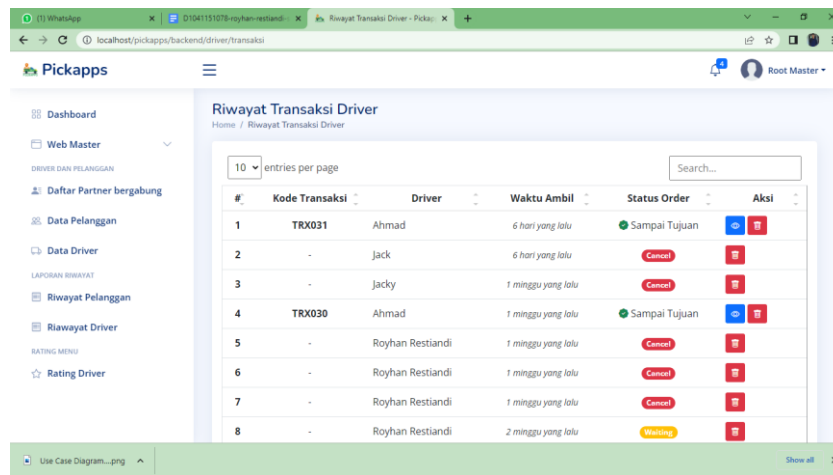
Halaman ini berisi *data-data* dari orderan yang pernah dilakukan oleh para *customer*. Admin dapat melihat data riwayat pesanan mulai dari yang terbaru hingga yang paling lawas. Di halaman ini admin dapat melihat *detail* dari pesanan yang pernah dilakukan *customer* dan juga dapat menghapus riwayat tersebut. Antarmuka halaman riwayat *customer* oleh admin dapat dilihat pada Gambar 4.34.



Gambar 4. 34 Tampilan Antarmuka Halaman Riwayat Customer (Admin)

Tampilan Antarmuka Halaman Riwayat Driver (Admin)

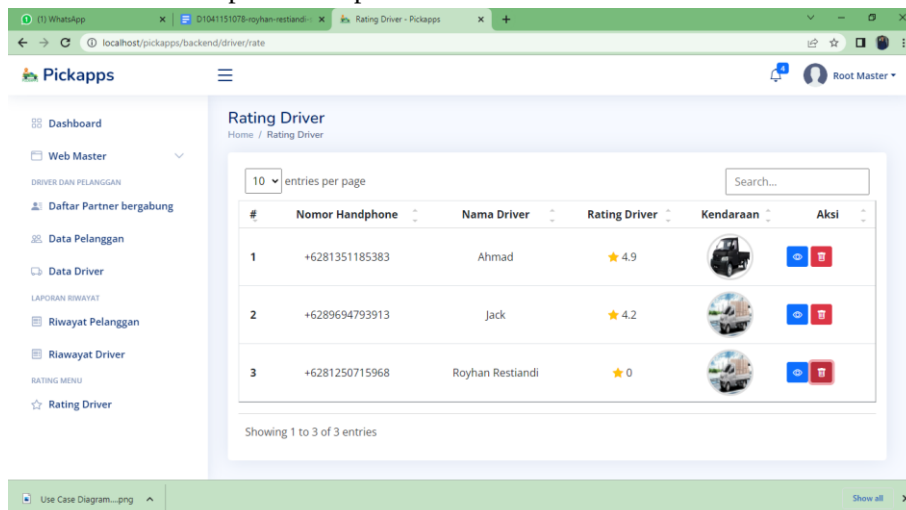
Halaman ini berisi *data* pesanan yang pernah dilakukan oleh *driver*. Pada halaman ini terdapat riwayat pesanan dari yang terbaru hingga yang paling lawas. Admin juga dapat melihat data pesanan dan menghapus data riwayat tersebut. Untuk tampilan antarmukanya dapat dilihat pada Gambar 4.35 berikut.



Gambar 4. 35 Tampilan Antarmuka Halaman Riwayat Driver (Admin)

Tampilan Antarmuka Halaman Data Rating Driver (Admin)

Pada halaman ini berisi *data rating* dari masing-masing *driver* yang terdaftar pada aplikasi Pickapps. *Admin* dapat melihat riwayat penilaian yang diberikan oleh *customer* kepada *driver*, *admin* juga dapat menghapus *driver* pada halaman ini. Untuk tampilan antarmuka halaman data rating *driver* milik *admin* dapat dilihat pada Gambar 4.36 dibawah ini.



Gambar 4. 36 Tampilan Antarmuka Halaman Data Rating Driver (Admin)

Hasil Pengujian

Aplikasi Layanan Jasa Ojek *Pick-up* Berbasis Lokasi Geospasial dan Google Maps dengan Metode *Location Based Service* menggunakan dua metode pengujian. Yaitu metode pengujian *black box* dan metode pengujian *user acceptance testing* (UAT). Adapun hasil dari kedua pengujian tersebut adalah sebagai berikut.

Pengujian Black Box

Pengujian akan dilakukan untuk mengetes fungsionalitas dan performa yang terdapat dalam penelitian yang dilakukan. Teknik pengujian *black box* dipilih karena dinilai sesuai dan cocok dengan logika yang diatur dengan hasil setelah di eksekusi. Pengujian yang dilakukan meliputi pengujian halaman dan fitur yang ada didalam aplikasi Pickapps. Berikut adalah tabel-tabel pengujian yang telah dilakukan.

Tabel 4. 1 Pengujian Fungsi Halaman Pada *Customer Apps*

Pengujian Yang Dilakukan	Hasil Yang Diharapkan	Hasil
Menjalankan aplikasi	Aplikasi berhasil berjalan dan menampilkan halaman <i>splash screen</i>	Berhasil

	selama dua detik lalu beralih ke halaman <i>login</i> .	
Menjalankan fungsi minta kode OTP	Aplikasi berhasil mengirimkan kode OTP ke nomor <i>handphone</i> yang sesuai dan menampilkan kolom <i>input</i> OTP di bawah kolom <i>input</i> nomor <i>handphone</i> .	Berhasil
Menjalankan fungsi <i>login</i>	Aplikasi menampilkan halaman selanjutnya berdasarkan logika yang dirancang yaitu halaman <i>register</i> atau beranda.	Berhasil
Menjalankan fungsi <i>register</i>	Aplikasi berhasil menerima data registrasi dan menampilkan halaman beranda.	Berhasil
Menjalankan fungsi cari kendaraan	Aplikasi berhasil menerima <i>input</i> data cari kendaraan dan melakukan <i>broadcast</i> kepada para <i>driver pick-up</i> . Setelah itu aplikasi membuka halaman tunggu penawaran harga dari <i>driver</i> .	Berhasil
Menjalankan fungsi tawaran harga masuk	Aplikasi menampilkan <i>list</i> tawaran harga yang masuk sesuai dengan <i>driver</i> yang mencoba mengambil orderan tersebut.	Berhasil
Menjalankan fungsi tolak tawaran	Aplikasi menghapus tawaran harga tersebut dari <i>list</i> .	Berhasil
Menjalankan fungsi terima tawaran	Aplikasi membuka halaman tunggu <i>driver</i> ke lokasi pengambilan barang dan memberikan notifikasi orderan berhasil.	Berhasil
Menjalankan fungsi hubungi <i>driver</i>	Aplikasi mendirect <i>customer</i> ke aplikasi whatsapp dan membuka <i>chat</i> baru ke nomor <i>driver</i> yang bersangkutan.	Berhasil
Menjalankan fungsi barang telah diambil	Aplikasi membuka halaman tunggu <i>driver</i> ke lokasi pengantaran barang.	Berhasil
Menjalankan fungsi barang telah sampai	Aplikasi membuka halaman berikan penilaian kepada <i>driver</i> .	Berhasil
Menjalankan fungsi beri penilaian	Aplikasi berhasil menerima input penilaian dan membuka halaman beranda.	Berhasil
Menjalankan fungsi <i>logout</i>	Aplikasi memutuskan <i>session</i> dari akun <i>customer</i> dan membuka halaman <i>login</i> .	Berhasil

Dari hasil pengujian pada tabel 4.1 dapat dilihat bahwa aplikasi *customer* dapat berjalan dengan baik mulai dari fungsi *login* hingga fungsi selesai melakukan *order pick-up* dan memberikan penilaian serta melakukan *logout*. Aplikasi memberikan hasil sesuai dengan yang diharapkan sesuai dengan fungsinya masing-masing.

Tabel 4. 2 Pengujian Fungsi Halaman Pada Driver Apps

Pengujian Yang Dilakukan	Hasil Yang Diharapkan	Hasil
Menjalankan aplikasi	Aplikasi berhasil berjalan dan menampilkan <i>halaman splash screen</i> selama dua detik lalu beralih ke halaman <i>login</i> .	Berhasil
Menjalankan fungsi minta kode OTP	Aplikasi berhasil mengirimkan kode OTP ke nomor <i>handphone</i> yang sesuai dan menampilkan kolom <i>input</i> OTP di bawah kolom <i>input</i> nomor <i>handphone</i> .	Berhasil
Menjalankan fungsi <i>login</i>	Aplikasi berhasil menerima <i>input</i> data <i>login</i> dan menampilkan halaman beranda.	Berhasil
Menjalankan fungsi <i>swipe button</i>	Aplikasi berhasil mendeteksi perubahan pada <i>swipe button</i> dan merubah <i>status driver</i> dari <i>offline</i> menjadi <i>online</i> atau sebaliknya.	Berhasil
Menjalankan fungsi orderan masuk	Aplikasi menampilkan <i>list</i> orderan yang telah di <i>request</i> oleh <i>customer</i> pada halaman beranda.	Berhasil
Menjalankan fungsi pilih orderan	Aplikasi menampilkan halaman terima / tawarkan harga	Berhasil
Menjalankan fungsi terima tawaran dengan harga <i>request</i>	Aplikasi membuka halaman menuju lokasi pengambilan barang dan memberikan notifikasi orderan berhasil diterima.	Berhasil
Menjalankan fungsi tawarkan harga	Aplikasi berhasil menerima <i>input</i> tawarkan harga dan menyampaikan tawaran tersebut kepada <i>customer</i> .	Berhasil
Menjalankan fungsi tawaran ditolak	Aplikasi membawa <i>driver</i> kembali ke halaman beranda	Berhasil
Menjalankan fungsi tawaran diterima	Aplikasi membawa <i>driver</i> ke halaman menuju lokasi pengambilan barang	Berhasil
Menjalankan fungsi saya telah tiba	Aplikasi menampilkan halaman menuju lokasi pengantaran barang	Berhasil
Menjalankan fungsi orderan selesai	Aplikasi membawa <i>driver</i> kembali ke halaman beranda	Berhasil

Menjalankan fungsi <i>logout</i>	Aplikasi memutuskan <i>session</i> akun <i>driver</i> dan menampilkan halaman <i>login</i>	Berhasil
----------------------------------	--	----------

Dari hasil pengujian pada tabel 4.2 dapat dilihat bahwa aplikasi *driver* dapat berjalan baik tanpa adanya *error* dan *bug* pada sistem. Semua fungsi yang dibuat dapat berjalan tanpa adanya hambatan sesuai dengan yang diharapkan.

Tabel 4. 3 Tabel Pengujian Kompatibilitas Jenis-Jenis *Browser*

Nama Browser	Hasil	Keterangan
Google Chrome	Berhasil	Dapat berjalan tanpa kendala.
Microsoft Edge	Berhasil	Dapat berjalan tanpa kendala.
Opera	Berhasil	Dapat berjalan tanpa kendala.
Vivaldi	Berhasil	Dapat berjalan tanpa kendala
Mozilla Firefox	<i>Error</i>	<i>Browser</i> tidak menampilkan captcha pada halaman <i>input</i> nomor <i>handphone</i> untuk keperluan mendapatkan kode OTP.
UC Browser	<i>Error</i>	<i>Browser</i> tidak menampilkan captcha pada halaman <i>input</i> nomor <i>handphone</i> untuk keperluan mendapatkan kode OTP.

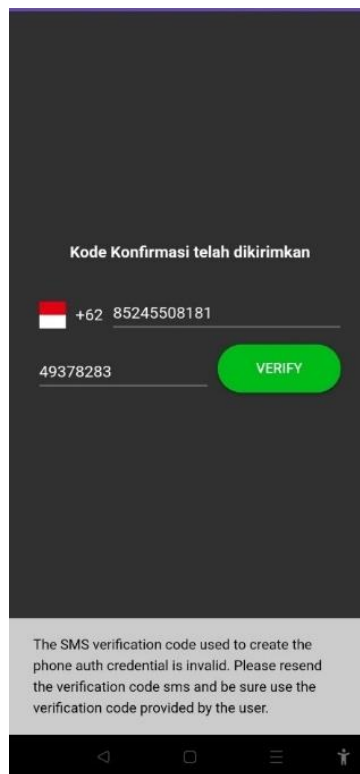
Dilihat dari hasil pengujian yang telah dilakukan pada tabel 4.3 dapat disimpulkan bahwa dari total 6 *browser* yang diujikan 4 diantaranya dapat berjalan tanpa ada kendala. 4 *browser* tersebut yaitu google chrome, microsoft edge, opera dan vivaldi. Sedangkan 2 *browser* lainnya mengalami *error* tidak dapat menampilkan captcha pada halaman *input* nomor *handphone* yang diperlukan untuk mendapatkan kode OTP. 2 *browser* tersebut adalah mozilla firefox dan uc.

Pengujian Fungsi Menggunakan Beberapa Skenario

Dilakukan beberapa skenario pengujian untuk melihat apakah fungsi dari masing-masing *apps* milik *customer* dan *driver* dapat berjalan dengan baik dan lancar serta sesuai dengan perancangan yang dilakukan. Untuk skenario yang dilakukan diantaranya:

A. Customer Apps

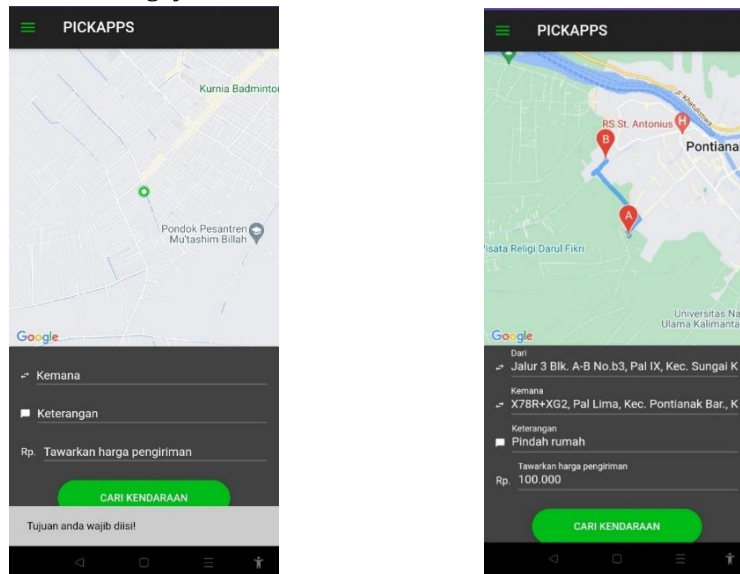
1. Skenario Pengujian Menginputkan Kode OTP Yang Salah



Gambar 4. 37 *Customer* Salah Mengisi Kode OTP

Berdasarkan gambar 4.37 dapat dilihat bahwa jika *customer* mengisi kode OTP dengan salah maka akan mendapat sebuah notifikasi bahwa kode OTP yang dimasukkan *invalid* dan diminta untuk mengisi kode OTP dengan benar agar dapat *login*.

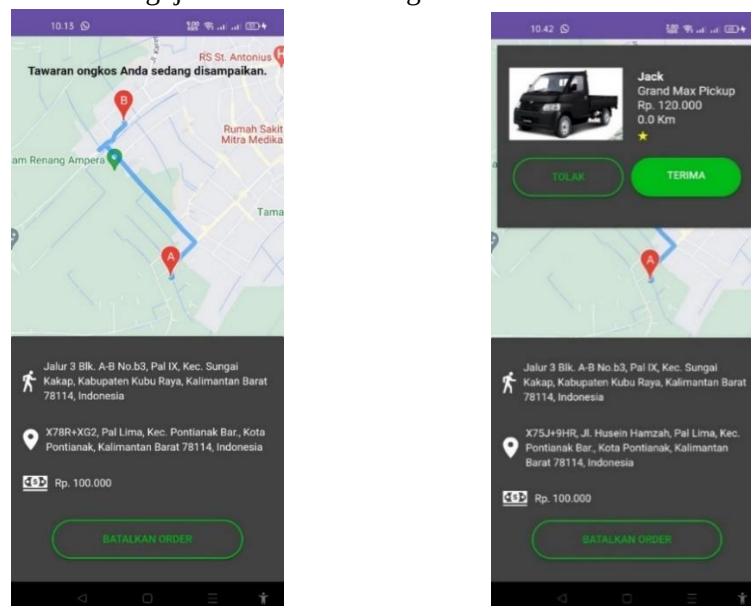
2. Skenario Pengujian Cari Kendaraan



Gambar 4. 38 Customer Melakukan Order Pick-up

Dari Gambar 4.38 dapat dilihat saat ingin mencari kendaraan *customer* harus terlebih dahulu mengisi *data order*. Jika *customer* mencoba melakukan *order* tanpa mengisi data terlebih dahulu maka akan mendapat sebuah notifikasi “tujuan anda wajib diisi!”. Setelah melengkapi *data order* maka *maps* akan menampilkan rute perjalanan dari titik pengambilan ke titik pengantaran barang.

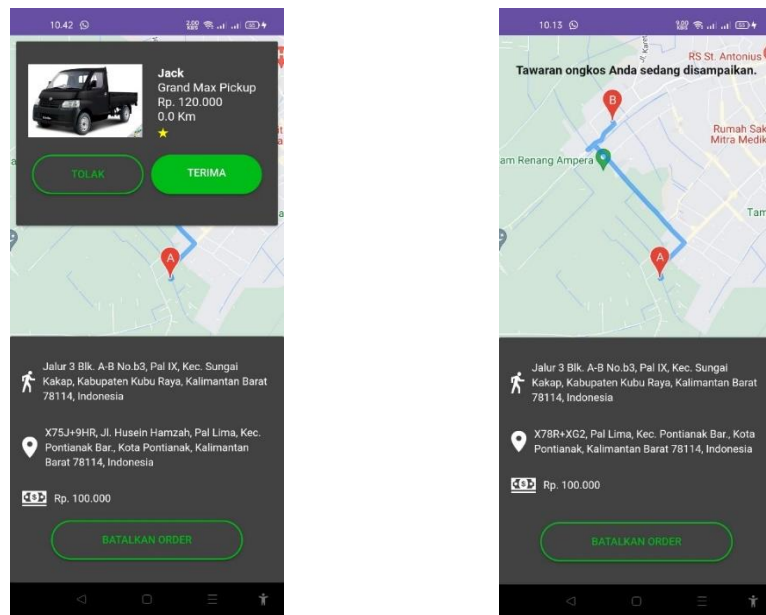
3. Skenario Pengujian Penawaran Harga Masuk Dari Driver



Gambar 4. 39 Customer Menerima Tawaran Harga Masuk Dari Driver

Berdasarkan Gambar 4.39 dapat dilihat saat *customer* berada pada halaman penawaran *order*, *customer* akan menerima tawaran harga masuk dari para *driver* yang sedang *online*. *Customer* dapat menerima lebih dari satu tawaran masuk dari tiap-tiap *driver*.

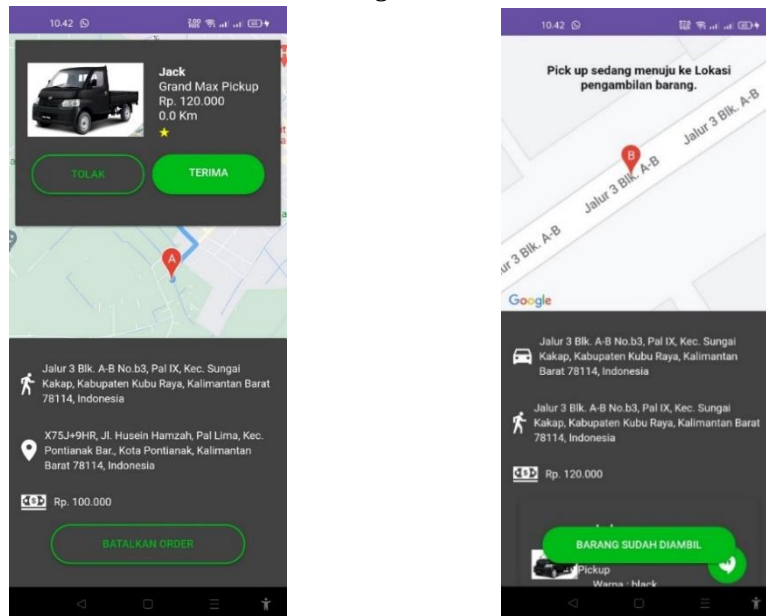
4. Skenario Pengujian Menolak Tawaran Harga Dari Driver



Gambar 4. 40 Customer Menolak Tawaran Harga Dari Driver

Berdasarkan Gambar 4.40 dapat dilihat bahwa jika *customer* menekan *button* tolak pada tawaran harga yang diberikan oleh *driver*, maka tawaran harga dari *driver* tersebut akan dihapus dari *list*.

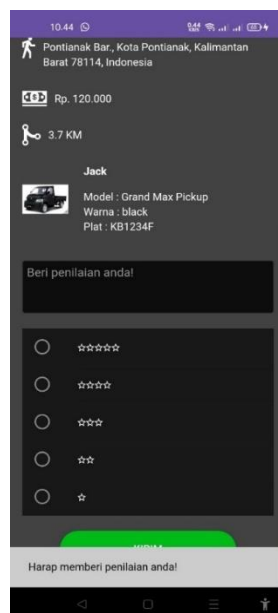
5. Skenario Menerima Tawaran Harga Dari *Driver*



Gambar 4. 41 *Customer* Menerima Tawaran Harga Dari *Driver*

Dari Gambar 4.6 dapat dijelaskan saat *customer* menekan *button* terima pada tawaran harga yang disampaikan oleh *driver*, maka *customer* akan dialihkan ke halaman tunggu *driver* ke lokasi pengambilan barang dan menerima notifikasi bahwa orderan berhasil.

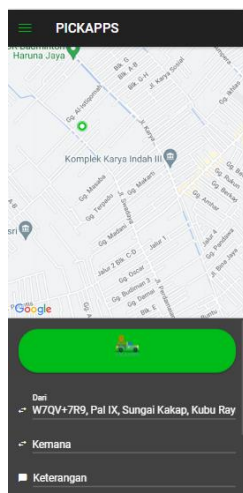
6. Skenario Pengujian Tidak Memberikan Penilaian



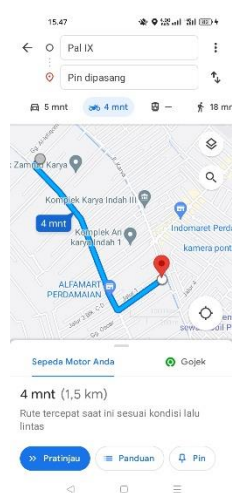
Gambar 4. 42 *Customer* Tidak Memberikan Penilaian Kepada *Driver*

Berdasarkan Gambar 4.2 dapat dilihat bahwa *customer* tidak bisa tidak memberikan penilaian kepada *driver* yang telah menyelesaikan orderan. *Customer* diwajibkan mengisi kolom penilaian dan *rating* yang ingin diberikan kepada *driver*.

7. Skenario Pengujian Tingkat Akurasi Lokasi



Gambar 4. 43 Lokasi Customer Pada Aplikasi Pickapps

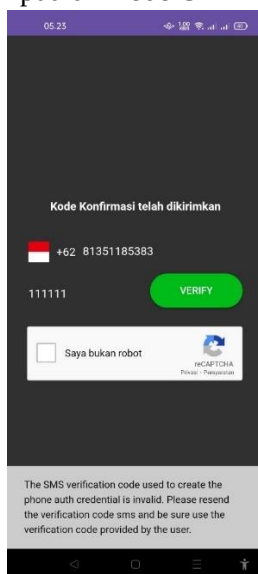


Gambar 4. 44 Posisi Customer Sebenarnya dan Selisih Jarak

Pada Gambar 4.43 dapat dilihat bahwa posisi *customer* saat ini pada aplikasi Pickapps adalah pada jalan Karya Gang Al-Istiqamah. Sedangkan jika dilihat pada Gambar 4.44 dapat dilihat pada kenyataan seharusnya posisi *customer* saat ini adalah pada komplek Ari Karya Indah 4, Jalur 3, Blok A-B nomor B5. Maka diketahui bahwa tingkat akurasi pembacaan lokasi *customer* pada aplikasi Pickapps kurang tepat atau sangat rendah. Selisih jarak antara titik lokasi pada aplikasi dan lokasi sebenarnya adalah kurang lebih 1,5 km.

Driver Apps

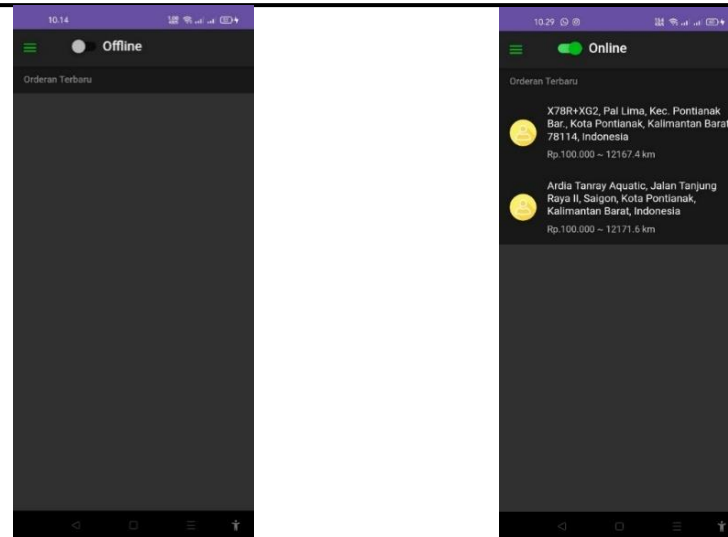
1. Skenario Pengujian Menginputkan Kode OTP Yang Salah



Gambar 4. 45 Driver Memasukkan Kode OTP Yang Salah

Dari Gambar 4.45 dapat dijelaskan bahwa saat *driver* mengisikan kode OTP secara salah maka *driver* akan mendapat notifikasi berupa kode OTP yang dimasukkan *invalid* dan diminta untuk mengisikan kode OTP secara benar.

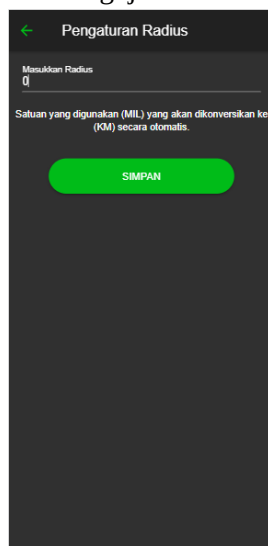
2. Skenario Pengujian *Swipe Button*



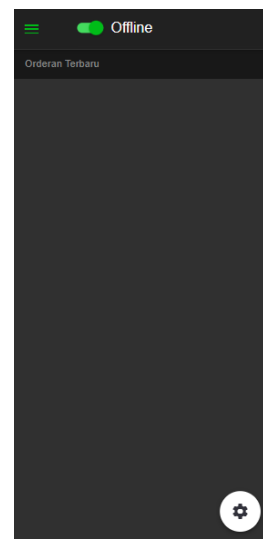
Gambar 4. 46 Driver Mengaktifkan *Swipe Button*

Berdasarkan Gambar 4.46 diatas dapat kita lihat bahwa saat *swipe button driver* sedang dalam keadaan *offline* maka *driver* tidak akan menerima orderan masuk dan setelah *driver* menggeser *swipe button* menjadi *online* maka *driver* akan mulai menerima orderan masuk dari para *customer*.

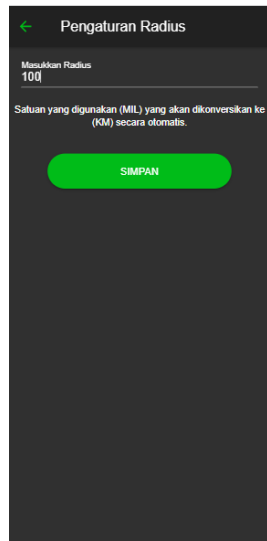
3. Skenario Pengujian Atur Radius



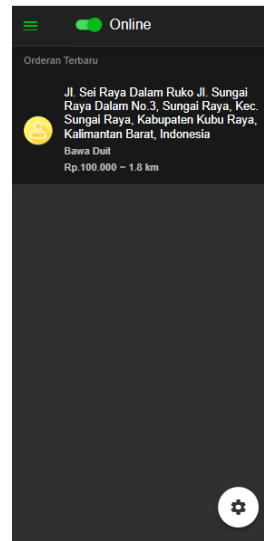
Gambar 4. 47 Jarak Radius Diatur Menjadi 0 KM



Gambar 4. 48 Tidak Ada Orderan Masuk Pada Radius Jarak 0 KM



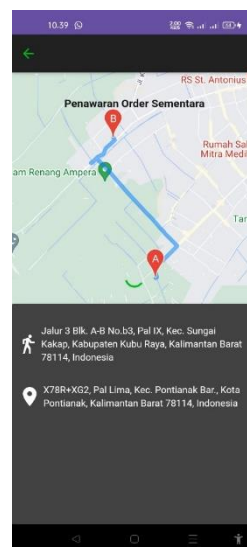
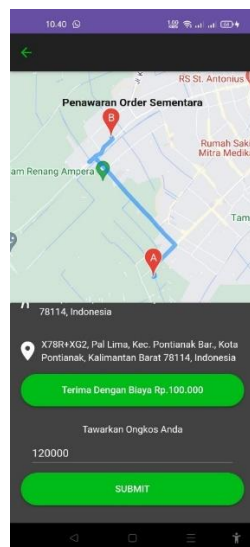
Gambar 4. 49 Jarak Radius Diatur Menjadi 100 KM



Gambar 4. 50 Terdapat Orderan Masuk Pada Radius Jarak 100 KM

Dari Gambar 4.47 terlihat bahwa radius jarak yang diatur adalah 0 KM. Itu artinya driver hanya akan menerima orderan masuk dari customer sejauh 0 KM dari posisi driver saat ini. Lalu pada Gambar 4.48 terlihat bahwa pada jarak 0 KM driver tidak menerima orderan masuk satupun dari customer, hal ini dikarenakan pada jarak 0 KM dari posisi driver saat ini tidak ada satupun customer yang melakukan orderan. Selanjutnya pada Gambar 4.49 terlihat driver memperluas jangkauan radius miliknya menjadi 100 KM dan pada Gambar 4.50 dapat dilihat bahwa pada jarak radius 100 KM terdapat seorang customer yang melakukan orderan.

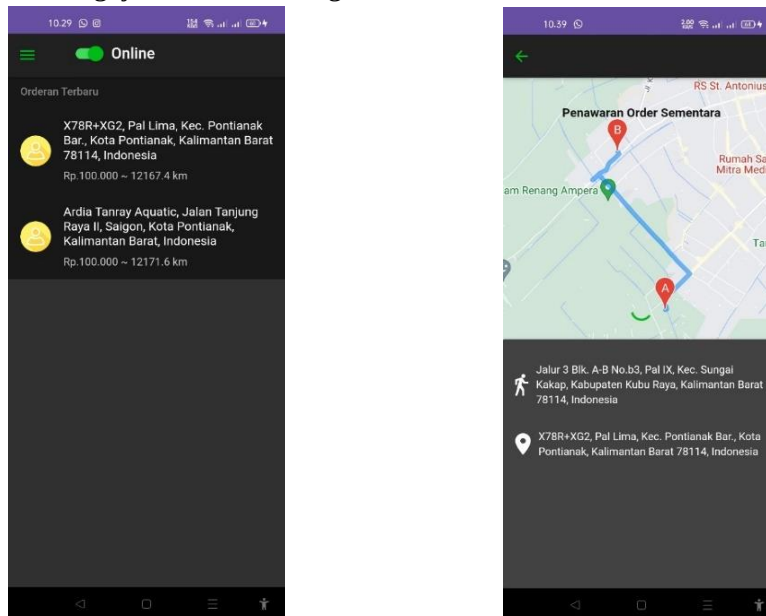
4. Skenario Pengujian Menawarkan Harga Kepada *Customer*



Gambar 4. 51 Driver Menawarkan Harga Kepada *Customer*

Dilihat dari Gambar 4.51 dijelaskan bahwa untuk bisa memberikan tawaran kepada *customer*, *driver* harus terlebih dahulu mengisi nominal harga yang ingin ditawarkan. Setelah menekan *button submit* maka *driver* akan menunggu tawaran tersebut disampaikan kepada *customer*.

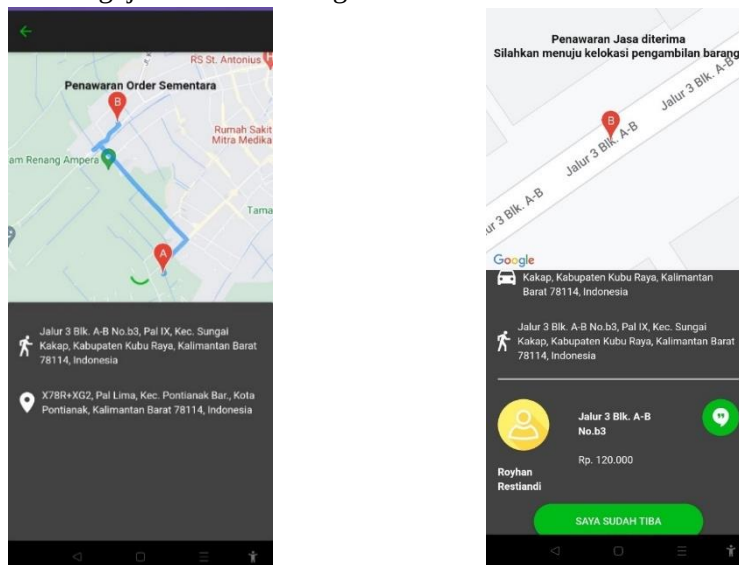
5. Skenario Pengujian Tawaran Harga Ditolak Oleh Customer



Gambar 4. 52 Tawaran Harga Driver Ditolak Oleh Customer

Dari Gambar 4.52 dapat dilihat saat sedang menunggu tawaran disampaikan kepada *customer*, *driver* akan tetap berada di halaman penawaran order sementara. Jika *customer* menolak tawaran yang diberikan oleh *driver*, maka *driver* akan dibawa kembali ke halaman beranda untuk dapat memilih orderan kembali.

6. Skenario Pengujian Tawaran Harga Diterima Oleh Customer



Gambar 4. 53 Tawaran Harga Driver Diterima Oleh Customer

Berdasarkan Gambar 4.53 dapat dijelaskan bahwa jika tawaran harga yang diberikan oleh *driver* diterima oleh *customer* maka *driver* akan dialihkan dari halaman penawaran order sementara ke halaman menuju lokasi pengambilan barang dan mendapat notifikasi bahwa orderan berhasil.

Pengujian User Acceptance Testing (UAT)

Tabel 4. 4 Hasil Pengujian User Acceptance Testing (UAT)

No.	Pertanyaan	Sangat Buruk	Buruk	Cukup	Baik	Sangat Baik

1	Apakah aplikasi yang dibuat mudah dimengerti untuk dipakai?	-	-	2	8	5
2	Apakah aplikasi dapat berjalan dengan lancar?	-	-	3	7	5
3	Apakah aplikasi nyaman untuk digunakan?	-	-	4	5	6
4	Apakah anda dapat dengan mudah menemukan tombol menu pada aplikasi pickapps?	-	-	3	4	8
5	Apakah informasi yang ditampilkan untuk keperluan memesan ojek <i>pick-up</i> mudah dimengerti?	-	-	2	8	5
6	Apakah tombol direct whatsapp yang disediakan dapat berjalan dengan sesuai?	-	-	4	5	6
7	Apakah aplikasi memudahkan anda untuk dapat memesan jasa ojek <i>pick-up</i> ?	-	-	4	4	7
8	Apakah informasi mengenai pesanan yang ditampilkan mudah dipahami?	-	-	5	6	4
9	Bagaimana penilaian tampilan antarmuka dari aplikasi pickapps?	-	-	7	3	5
10	Dari segala aspek bagaimana penilaian anda tentang aplikasi pickapps?	-	-	2	10	3

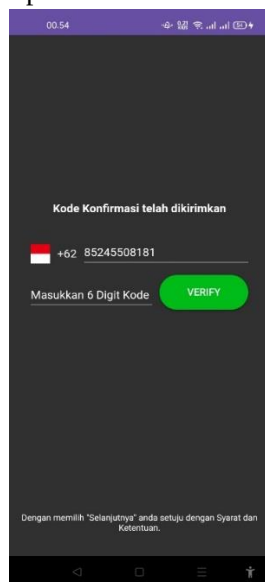
Dari tabel 4.4 dapat dilihat bahwa hasil penilaian dari survei yang telah dilakukan menggunakan *user acceptance testing* (UAT) didapatkan kesimpulan sebagai berikut:

1. 13,3% menyatakan bahwa aplikasi yang dibuat cukup mudah dimengerti untuk digunakan, 53,3% menyatakan mudah dimengerti dan 33,3% mengatakan aplikasi yang dibuat sangat mudah dimengerti untuk digunakan.
2. Aplikasi dapat berjalan dengan cukup lancar pada 20% perangkat responden, 46,6% mengatakan berjalan dengan lancar dan 33,3% menyatakan aplikasi dapat berjalan dengan sangat lancar.
3. Lalu sebesar 26,6% mengatakan bahwa aplikasi yang dibuat cukup nyaman untuk digunakan, 33,3% merasa nyaman menggunakan aplikasi dan 40% menyatakan sangat nyaman dalam menggunakan aplikasi yang dibuat.
4. Kemudian tombol menu yang didesain cukup mudah ditemukan oleh 20% responden, mudah ditemukan oleh 26,6% pengguna dan 53,3% mengatakan mudah untuk menemukannya.
5. 13,3% menyatakan bahwa informasi yang ditampilkan untuk keperluan memesan ojek *pick-up* cukup mudah difahami, 53,3% merasa mudah dipahami dan 33,3% mengatakan informasi yang ditampilkan sangat mudah untuk difahami.

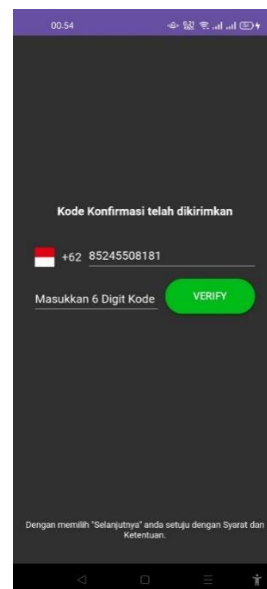
6. Tombol *direct* whatsapp yang disediakan dapat berjalan dengan cukup sesuai menurut 26,6% responden, 33,3% mengatakan berjalan dengan sesuai dan sisanya 40% mengatakan menyatakan dapat berjalan dengan sangat sesuai.
7. Lalu 26,6% menyatakan bahwa aplikasi yang dibuat cukup memudahkan mereka dalam memesan jasa ojek *pick-up*, 26,6% merasa mudah dalam memesan dan 46,6% mengatakan sangat memudahkan bagi responden.
8. Kemudian informasi mengenai pesanan yang ditampilkan menurut 33,3% responden cukup mudah dimengerti, 40% merasa mudah dimengerti dan sisanya 26,6% mengatakan informasi yang diberikan sangat mudah dipahami.
9. 46,6% menilai tampilan keseluruhan antarmuka aplikasi Pickapps cukup bagus, 20% mengatakan bagus dan 33,3% menyatakan sangat bagus.
10. Penilaian keseluruhan dari segala aspek aplikasi Pickapps mendapatkan penilaian 13,3% cukup baik, 66,6% menilai baik dan 20% mengatakan sangat baik.

Demo Pengujian Aplikasi

Berikut adalah tampilan aplikasi saat melakukan *demo* melakukan pesanan jasa ojek *pick-up* dari pihak *customer* dan *driver*, mulai dari *login* hingga memberikan penilaian dan *rating*. Untuk *demo* pengujian dapat dilihat dibawah ini.

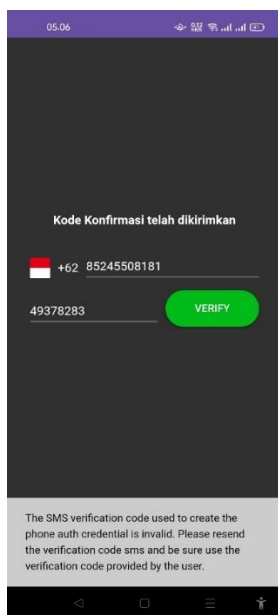


Gambar 4. 54 *Customer* mengisi nomor *handphone* pada halaman *login*

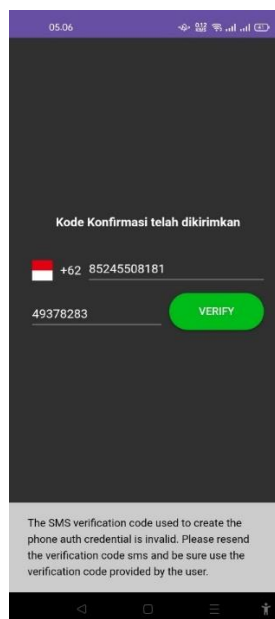


Gambar 4. 55 *Driver* Mengisi nomor *handphone* pada halaman *login*

Pada Gambar 4.54 dan 4.55 untuk tahap awal mulai melakukan pesanan ojek *pick-up* pihak *customer* dan *driver* harus terlebih dahulu mengisi nomor *handphone* mereka untuk mendapatkan kode OTP sebagai syarat *login* kedalam aplikasi.

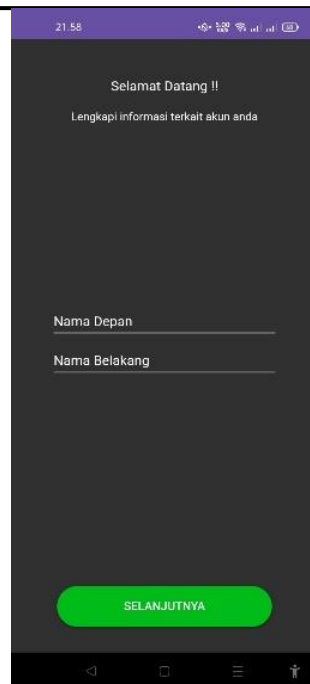


Gambar 4. 56 *Customer* mengisi kode OTP yang didapat melalui sms



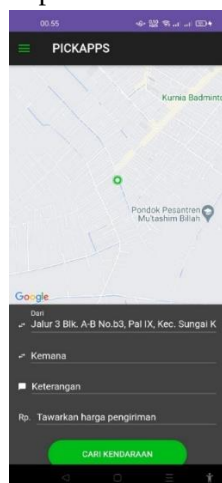
Gambar 4. 57 *Driver* mengisi kode OTP yang didapat melalui sms

Selanjutnya pada Gambar 4.56 dan 4.57 setelah *customer* dan *driver* mendapatkan kode OTP yang dikirim *via* sms maka *customer* dan *driver* mengisi kode OTP tersebut pada kolom yang telah disediakan dan menekan *button* "verify". Pada tahap ini setelah *customer* dan *driver* menekan *button* "verify" maka sistem akan menjalankan sebuah fungsi pengecekan kode OTP. Jika kode OTP yang dimasukkan salah maka *customer* dan *driver* akan terus berada dihalaman ini dan diminta untuk mengisi kode OTP dengan benar. Jika kode OTP yang dimasukkan sudah benar maka pada aplikasi *customer* akan menjalankan sebuah fungsi lagi yaitu pengecekan apakah nomor *handphone* yang dimasukkan sebelumnya sudah pernah *login* ke dalam aplikasi Pickapps atau belum. Jika belum maka *customer* akan diarahkan ke halaman register terlebih dahulu untuk melakukan pengisian data diri. Dan jika sudah pernah *login* sebelumnya, *customer* akan langsung diarahkan ke halaman beranda. Namun pada aplikasi *driver* fungsi ini tidak ada karena sebelum menjadi *driver* mereka sudah terlebih dahulu melakukan *register* pada halaman *web* yang telah disediakan.

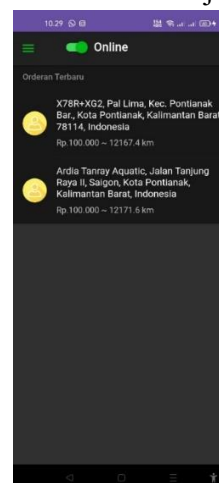


Gambar 4. 58 Customer Mengisikan Data Diri Pada Halaman Register

Gambar 4.58 adalah halaman *register*, disini *customer* diminta untuk mengisikan data diri mereka berupa nama depan dan nama belakang. Setelah ini menekan *button* “selanjutnya”.

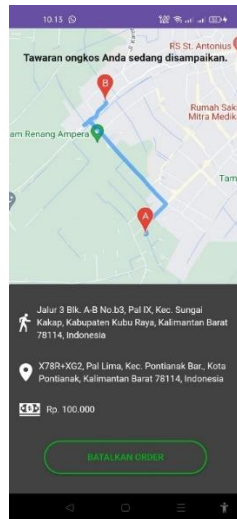


Gambar 4. 59 Customer Mengisikan Data Pesanan Pada Halaman Beranda

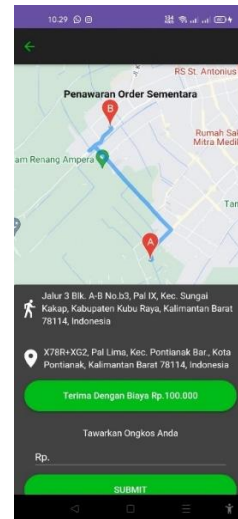


Gambar 4. 60 Driver Menunggu Pesanan Masuk Pada Halaman Beranda

Selanjutnya pada Gambar 4.59 setelah *customer* berhasil *login* kedalam aplikasi, *customer* mengisikan *data* pesanan pada kolom yang telah disediakan dan menekan *button* “cari kendaraan” maka sistem akan menerima *data* pesanan tersebut dan melakukan broadcast kepada semua *driver* yang sedang online. Lalu pada Gambar 4.60 *driver* yang sedang *online* menerima *data* pesanan dari *customer* tadi. *Driver* memilih salah satu pesanan yang mereka inginkan dengan pertimbangan apakah mereka bisa menjalankan pesanan tersebut atau tidak melalui *data* pesanan yang telah diisikan oleh *customer*.

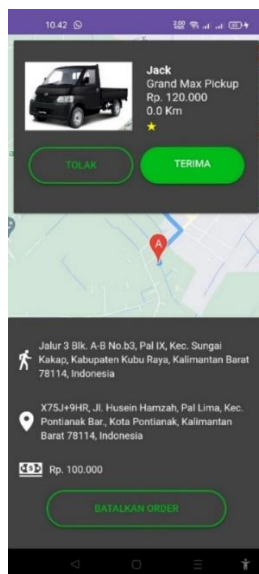


Gambar 4. 61 Customer Menunggu Tawaran Harga Masuk Pada Halaman Tunggu Tawaran Order

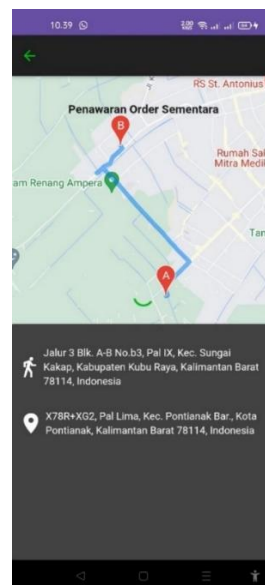


Gambar 4. 62 Driver Menawarkan Harga Jasa Kepada Salah Satu Pesanan Customer

Pada Gambar 4.61 dapat dilihat bahwa setelah *customer* menekan *button* “cari kendaraan” *customer* akan dibawa ke halaman tunggu tawaran *order*. Pada halaman ini *customer* akan menunggu tawaran harga yang akan dikirimkan oleh para *driver*. Gambar 4.62 adalah halaman berikan penawaran harga yang akan dilakukan oleh *driver*. Setelah sebelumnya *driver* memilih salah satu pesanan yang ada pada *list* di halaman beranda. Disini *driver* dapat memasukkan nominal biaya yang dia inginkan lalu menekan *button* “submit” maka sistem akan mengirimkan data tawaran tersebut kepada *customer* yang bersangkutan.



Gambar 4. 63 Tawaran Harga Dari Driver Masuk Kepada Customer

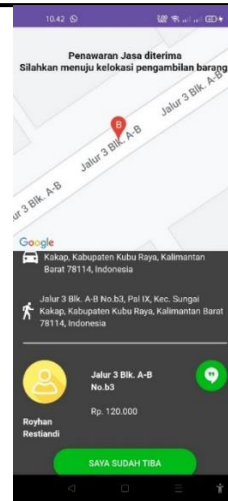


Gambar 4. 64 Driver Menunggu Tawaran Diterima / Ditolak

Dilihat pada Gambar 4.63 bahwa *customer* telah menerima tawaran harga yang dikirimkan oleh *driver*. *Customer* dapat memilih untuk menolak atau menerima tawaran harga tersebut. Namun karena disini tujuan utama adalah *demo* melakukan pesanan maka *customer* akan menerima tawaran tersebut. Lalu selanjutnya pada Gambar 4.64 *driver* akan menunggu pada halaman yang sama apakah tawaran harga yang diberikan ditolak atau diterima oleh *customer*.

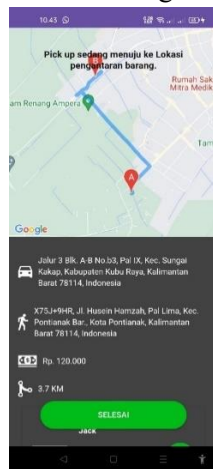


Gambar 4. 65 Customer Menunggu Driver Menuju Lokasi Pengambilan Barang

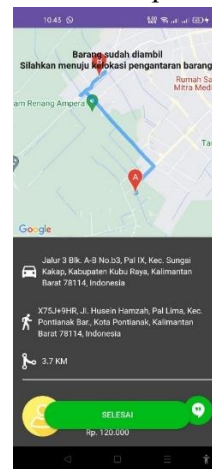


Gambar 4. 66 Driver Menuju Lokasi Pengambilan Barang

Gambar 4.65 dapat dilihat setelah *customer* menekan *button* “terima” maka pesanan dianggap berhasil dan *customer* dibawa ke halaman tunggu *driver* menuju lokasi pengambilan barang. Gambar 4.66 menjelaskan bahwa setelah *customer* menerima penawaran harga maka *driver* akan mendapat notifikasi bahwa pesanan berhasil dan *driver* akan langsung dibawa ke halaman menuju lokasi pengambilan barang untuk segera memulai menjalankan pesanan. Saat sudah sampai di lokasi pengambilan barang, *driver* menekan *button* “saya telah tiba” dan *customer* akan diminta untuk menekan *button* “barang sudah diambil” untuk melanjutkan ke tahap selanjutnya.

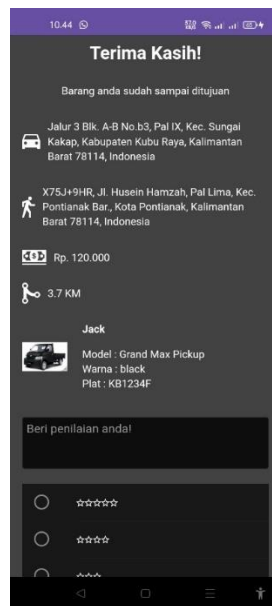


Gambar 4. 67 Customer Menunggu Driver Menuju Lokasi Pengantaran



Gambar 4. 68 Driver Menuju Lokasi Pengantaran

Pada Gambar 4.67 dan 4.68 dijelaskan setelah *customer* menekan *button* “barang sudah diambil” pada halaman sebelumnya dan *driver* menekan *button* “saya telah tiba” maka *customer* dan *driver* akan dibawa ke halaman menuju lokasi pengantaran barang. Disini *driver* akan menuju lokasi pengantaran dan *customer* akan menunggu hingga *driver* tersebut sampai pada tujuan. Setelah sampai dan barang telah dibongkar/diturunkan, *customer* dan *driver* menekan *button* “selesai”.



Gambar 4. 69 *Customer* Memberikan Rating Kepada *Driver*

Pada Gambar 4.69 dapat dilihat setelah *customer* menekan *button* “selesai” selanjutnya *customer* akan diarahkan ke halaman berikan penilaian dan *rating*. Disini *customer* memberikan penilaian dan *rating* atas kinerja *driver* yang bersangkutan. Setelah mengisi penilaian dan *rating* *customer* menekan *button* “selesai” dan pesanan dinyatakan benar-benar telah selesai.

Analisis Hasil Perancangan dan Pengujian

Berikut ini adalah hasil analisis hasil perancangan dan pengujian perangkat lunak aplikasi permainan dalam penelitian:

1. Hasil pengujian fungsional yang dilakukan oleh sistem kepada *customer apps* menggunakan metode *black box* menunjukkan semua fungsi yang ada dalam aplikasi dapat berjalan dengan lancar tanpa adanya *bug* dan *error*.
2. Hasil pengujian fungsional yang dilakukan oleh sistem kepada *driver apps* menggunakan metode *black box* menunjukkan semua fungsi yang ada dalam aplikasi dapat berjalan dengan lancar tanpa adanya *bug* dan *error*.
3. Aplikasi yang dibuat dapat berjalan lancar pada 3 dari 5 *browser* yang diujikan. Diantaranya google chrome, microsoft edge dan opera. Sedangkan 2 *browser* lain yaitu mozilla firefox dan UC *browser* mengalami kendala tidak dapat menampilkan captcha pada halaman *input* nomor *handphone*.
4. Berdasarkan hasil pengujian *user acceptance testing* (UAT) dapat dikatakan aplikasi yang dibuat dapat membantu *customer* dalam memesan jasa ojek *pick-up* dengan lebih mudah.
5. Berdasarkan hasil *user acceptance testing* (UAT) aplikasi Pickapps dapat menampilkan informasi yang berguna saat *customer* memesan jasa ojek *pick-up*.
6. Skema tawar menawar harga dapat digunakan *customer* untuk menyesuaikan biaya jasa sesuai keinginan mereka.

Aplikasi yang dibuat dapat berjalan dengan baik sehingga dapat menjadi salah satu alternatif pilihan bagi penyedia jasa dalam memasarkan jasa ojek *pick-up* milik mereka.

Kesimpulan

Berdasarkan hasil analisis dan pengujian yang telah dilakukan menggunakan metode *black box* dan metode *user acceptance testing* (UAT) maka dapat disimpulkan bahwa:

1. Aplikasi yang dibuat dapat menampilkan informasi penyedia jasa yang bersedia mengambil pesanan dari *customer* beserta data kendaraan, harga yang ditawarkan dan rating yang dapat dijadikan pertimbangan bagi *customer* dalam memesan jasa ojek *pick-up*. Hal ini sudah sesuai dengan tujuan penelitian point pertama.
2. Tahapan dan alur memesan yang dirancang dalam aplikasi Pickapps lebih sedikit dibandingkan cara memesan lewat cara pemasaran yang dilakukan oleh penyedia jasa pada saat sekarang ini, sehingga waktu yang diperlukan dalam memesan jasa ojek *pick-up* lebih singkat. Hal ini juga sudah sesuai dengan tujuan penelitian point pertama.
3. Skema tawar menawar harga yang dirancang saat memesan jasa ojek *pick-up* dapat digunakan oleh *customer* dalam menyesuaikan biaya jasa yang sesuai dengan keinginan mereka. Hal ini sesuai dengan tujuan penelitian point kedua.

Aplikasi dapat menerima pesanan masuk dari para *customer* sehingga dapat menjadi alternatif baru bagi penyedia jasa dalam memasarkan jasa ojek *pick-up* milik mereka. Ini sudah sesuai dengan tujuan penelitian point ketiga.

Daftar Pustaka

- Adil, A. (2017). Sistem Informasi Geografis. Penerbit Andi.
- Almunawaroh, Siti Nur Fadlilah. "Penentuan Area Pick Up Point Ojek Online untuk Mengurangi Kemacetan Lalu Lintas di Sekitar Stasiun Kereta Api Jabodetabek." Jurnal Penelitian Transportasi Darat 21.2 (2019): 145-154.
- Bongkar muat. 2016. Pada KBBi Daring. Diambil 5 Juli 2022 dari <https://kbbi.kemdikbud.go.id/entri/bongkar%20muat>
- Connolly, T & Begg, C. (2002). Database System. UK : Addison Wesley.
- Dicoding Intern. (2020). Apa itu Firebase? Pengertian, Jenis-Jenis, dan Fungsi Kegunaannya. Dicoding. <https://www.dicoding.com/blog/apa-itu-firebase-pengertian-jenis-jenis-dan-fungsi-kegunaannya/>
- Doshi, Pankti, Pooja Jain, and Abhishek Shakwala. "Location Based Services and integration of google maps in android." International Journal of Engineering and Computer Science 3.03 (2014).
- Dvorski, Dalibor D. "Installing, configuring, and developing with Xampp." Skills Canada (2007).
- Ipo, Igo Suparius. (2021). Rancang Bangun Sistem Informasi Geografis Persebaran Bengkel, Tambal Ban dan Minyak Eceran di Kota Pontianak Berbasis Web, Skripsi, Teknik Informatika UNTAN, Pontianak.
- Jasa. 2016. Pada KBBi Daring. Diambil 5 Juli 2022, dari <https://kbbi.kemdikbud.go.id/entri/jasa>.
- Kusuma, Risnu Arya. (2020). Aplikasi Peringatan Rambu Lalu Lintas Dengan Metode Location Based Service Berbasis Mobile, Skripsi, Teknik Informatika UNTAN, Pontianak.
- Mehta, Heeket, Pratik Kanani, and Priya Lande. "Google maps." Int. J. Comput. Appl 178.8 (2019): 41-46.
- Ojek. 2016. Pada KBBi Daring. Diambil 5 Juli 2022, dari <https://kbbi.kemdikbud.go.id/entri/ojek>.
- Otaduy, Itziar, and Oscar Díaz. "User acceptance testing for Agile-developed web-based applications: Empowering customers through wikis and mind maps." Journal of Systems and Software 133 (2017): 212-229.
- Pikap(pikap). 2016. Pada KBBi Daring. Diambil 5 Juli 2022, dari <https://kbbi.kemdikbud.go.id/entri/pikap>.
- Richardson, Leonard, and Sam Ruby. RESTful web services. " O'Reilly Media, Inc.", 2008.
- Roby, Yamto. (2016). Manajemen Basis Data Menggunakan MySQL. Yogyakarta: deepublisher.
- Saparianto, Muhammad Fajri. (2019). Aplikasi Pemesanan Tukang Berbasis Mobile, Skripsi, Teknik Informatika UNTAN, Pontianak.
- Satzinger, John W. 2011. Systems Analysis And Design In A Changing World. Bookbarn International.
- Siswanto. (2013). Sistem Informasi Geografis Objek Wisata Menggunakan Google Maps API Studi Kasus Kabupaten Mojokerto. 1–5.

- Stefan Steiniger, Moritz Neun, & Alistair Edwards. (2006). Foundations of Location Based Services. 1–28.
- Sukamto, R. A., (2009). Black box Testing. <http://rosa-as.id/download/8-Langkah-langkahPengujianEvaluasiPirantiLunak.pdf> akses tanggal 18 September 2021.
- Sukamto, R. A., dan Salahudin, M. (2013). "Rekayasa Perangkat Lunak, Terstruktur dan berorientasi objek." Penerbit Informatika Bandung.
- Tandel, S., and Abhishek Jamadar. "Impact of progressive web apps on web app development." International Journal of Innovative Research in Science, Engineering and Technology 7.9 (2018): 9439-9444.
- Tim, E. M. S. Pemrograman Android dalam Sehari. Elex Media Komputindo, 2015.
- Vladimir Kharlampidi. (2022). Framework7. <https://framework7.io/>. akses tanggal 6 Juli 2022.
- Xia, Nan, et al. "Accessibility based on Gravity-Radiation model and Google Maps API: A case study in Australia." Journal of Transport Geography 72 (2018): 178-190.