

APLIKASI TRAVEL AGENT BERBASIS ANDROID DENGAN IMPLEMENTASI REST API MENGGUNAKAN RETROFIT

Dewa Asmara Putra*¹⁾, Supriyadi²⁾

1. Universitas Kristen Satya Wacana, Indonesia
2. Universitas Kristen Satya Wacana, Indonesia

Article Info

Kata Kunci: Travel Agent; Java; Android; Fitur Custom; API

Keywords: *Travel Agent; Java; Android; Custom Features; API*

Article history:

Received 21 May 2023

Revised 4 June 2023

Accepted 18 June 2023

Available online 1 December 2023

DOI :

<https://doi.org/10.29100/jipi.v8i4.4271>

* Corresponding author.

Dewa Asmara Putra

E-mail address:

dewaasmara589@gmail.com

ABSTRAK

Setelah beberapa tahun Indonesia terkena dampak dari pandemi Covid 19 yang membatasi segala kegiatan orang-orang untuk berpergian. Membuat masyarakat menjadi lebih terbiasa menggunakan *smartphone*. Sementara itu, pasca pandemi bermunculan juga fenomena yang dikenal sebagai *revenge travel* atau perjalanan wisata balas dendam. Pada penelitian ini terdapat studi kasus untuk bisnis *travel agent* yang ada di kota Salatiga yang bernama Tenta Tour. Berdasarkan diskusi yang dilakukan, Tenta Tour membutuhkan sebuah aplikasi untuk membantu mempersingkat proses pertemuan dengan pelanggan saat pemesanan paket wisata. Metode penelitian yang digunakan pada penelitian ini adalah model *prototype* dan untuk analisis pengumpulan data menggunakan *System Usability Scale* (SUS). Berdasarkan hal tersebut, hasil penelitian adalah aplikasi Travel Agent berbasis Android yang mengimplementasikan REST API (Retrofit). Dengan tampilan-tampilan lengkap yang ada pada aplikasi yakni *Splash Screen, Home, Berangkat, Detail Paket Wisata, Detail Paket Custom, Pulang, Detail Pemesan, Detail Pesanan, Login, Daftar dan Riwayat Pesanan*. Fitur unggulan yang diberikan aplikasi Travel Agent ini sendiri yaitu kemudahan custom paket wisata seperti transportasi, tempat wisata, tempat makan, dan hotel. Selain itu, aplikasi Travel agent memberikan *output pdf rundown* atau jadwal perjalanan dan *auto send email* ke *customer* dan *admin*. Selain itu, berdasarkan data *black box testing* dan penilaian dengan metode SUS menunjukkan bahwa fitur-fitur pada Aplikasi Travel Agent dapat berjalan sebagaimana mestinya serta layak digunakan dan diterima dengan baik oleh pihak *client* yakni Tenta Tour dan beberapa *responden* yang dapat artikan sebagai *customer*.

ABSTRACT

After several years, Indonesia has been affected by the COVID-19 pandemic, which has limited all people's activities to travel. Making people more accustomed to using smartphones. Meanwhile, after the pandemic, a phenomenon known as *revenge travel* has emerged. In this research, there is a case study for a travel agent business in the city of Salatiga called Tenta Tour. Based on the discussions that were conducted, Tenta Tour needed an application to help shorten the process of meeting with customers when ordering tour packages. The research method used in this study is a prototype model and data collection analysis using the System Usability Scale (SUS). Based on this, the results of the study are Android-based travel agent applications that implement a REST API (Retrofit). With complete views in the application, namely *Splash Screen, Home, Departure, Tour Package Details, Custom Package Details, Return, Order Details, Order Details, Login, Register, and Order History*. The excellent features provided by the Travel Agent application itself are the convenience of custom tour packages such as transportation, tourist attractions, places to eat, and hotels. In addition, the travel agent application provides a pdf rundown output or travel itinerary and automatically sends emails to customers and admins. In addition, based on black box testing data and assessment using the SUS method, it shows that the features of the Travel Agent Application can work properly, are suitable for use, and are well received by the client, namely Tenta Tour and several respondents who can be interpreted as customers.

I. PENDAHULUAN

SETELAH beberapa tahun sebelumnya Indonesia mengalami pandemi Covid 19. Dimana Indonesia pertama kali muncul virus corona pada Maret 2020 [1]. Pandemi Covid 19 tersebut membatasi segala kegiatan orang-orang untuk berpergian atau beraktivitas di luar rumah sehingga memberikan dampak yang cukup besar juga pada sektor pariwisata Indonesia serta bisnis-bisnis seperti *travel agent*. Namun dibalik itu semua juga membuat orang-orang Indonesia menjadi lebih terbiasa menggunakan berbagai teknologi seperti *smartphone* untuk melakukan segala sesuatu serta menunjang aktivitasnya agar dapat dilakukan dari rumah saja [2]. Pasca pandemi Covid 19 sekarang ini, dengan dilakukannya pembukaan kembali pariwisata di Indonesia sehingga masyarakat dan wisatawan dapat melakukan perjalanan wisata. Mulai bermunculan juga fenomena yang dikenal sebagai *revenge travel* atau perjalanan wisata balas dendam. *Revenge travel* merujuk pada keinginan banyak orang untuk melepas kepenatan dan kebosanan yang dialami selama pandemi Covid 19 dengan melakukan perjalanan wisata [3].

Bisnis-bisnis yang bergerak pada sektor pariwisata seperti *travel agent* dapat memanfaatkan peluang yang ada sekarang. Salah satunya membuat sebuah aplikasi *mobile* untuk menunjang proses pemesanan paket perjalanannya. Serta dengan adanya pula fitur *custom* paket perjalanan sesuai keinginan atau kebutuhan *customer*. Berdasarkan hal tersebut, membuat *customer* tidak hanya terpaku pada paket yang diberikan saja, yang dimana belum tentu juga *customer* membutuhkan atau menginginkan tempat wisata atau keperluan wisata yang diberikan pada paket yang sudah ada [4]. Pada penelitian ini terdapat studi kasus untuk bisnis *travel agent* yang ada di kota Salatiga yang bernama Tenta Tour. Menurut hasil diskusi dengan pemilik Tenta Tour menerangkan bahwa bisnis tersebut ingin membuat dan membutuhkan sebuah aplikasi yang dapat membantu menjembatani antara *customer* dan Tenta Tour dalam proses pemesanan paket perjalanan. Oleh sebab itu, pada kasus yang sering dialami yakni proses pertemuan tidak berulang antara Tenta Tour dan customer atau dapat mempersingkat proses pesanan *customer* dengan hanya mengakses aplikasi *mobile* yang sudah menyediakan paket *traveling* tersebut.

Berdasarkan uraian sebelumnya maka akan dilakukan perancangan dan implementasi sebuah aplikasi *mobile* yang dapat digunakan dan diakses *customer* dengan hanya menggunakan *smartphone* android. Untuk pembuatan aplikasi berbasis android tersebut akan digunakannya bahasa pemrograman Java [5]. Untuk data-data akan disimpan dan diakses dengan bantuan internet saat pengimplementasian metode REST API yang memanfaatkan salah satu *library* API pada Java yakni Retrofit. Lebih tepatnya menggunakan Retrofit2 sebagai REST *Client* karena Retrofit memang sesuai dengan aplikasi yang dibuat dengan Java dan dinilai lebih aman untuk aplikasi android dan Java. Pemanfaatan Retrofit2 juga dapat mempermudah dalam pertukaran data pada saat dilakukan pengubahan data objek Java menjadi format JSON dan begitu juga dari JSON ke objek Java [6].

Hasil yang didapatkan yakni aplikasi Travel Agent yang mudah digunakan serta memiliki fitur unggulan untuk *customer* dapat mengcustom paket perjalanannya seperti kendaraan, lokasi wisata, tempat makan dan hotel sesuai keinginannya dan dapat diimplementasikan pada aplikasi dengan berbasis Android. Oleh karena itu, nantinya aplikasi tersebut dapat bermanfaat, tepatguna, dan membantu bisnis *travel agent* Tenta Tour agar berjalan dengan efektif serta memperoleh banyak *customer* dengan paket menarik yang disediakan serta adanya proses *custom* paket perjalanan juga yang pastinya lebih fleksibel dan sesuai keinginan *customer*.

Dilihat dari latar belakang penelitian ini, maka masalah yang muncul adalah bagaimana dapat terciptanya sebuah sistem kerja aplikasi Travel Agent yang berguna untuk pemesanan paket *traveling* dengan adanya fitur *custom* sesuai keinginan *customer* dan dapat dimanfaatkannya API untuk menyimpan serta mengakses data-datanya ?.

Tujuan yang ingin dicapai dari penelitian ini yakni untuk dapat merancang aplikasi Travel Agent berbasis android untuk menjembatani proses pemesanan paket wisata antara Tenta Tour dengan *customer* yang pastinya mudah digunakan dan diakses dengan internet dari *smartphone* android serta dapat diterapkannya REST API dengan menggunakan *library* pendukungnya adalah Retrofit.

Adapun manfaat yang ingin dicapai dari penelitian ini adalah dengan terciptanya aplikasi tersebut mudah saat digunakan customer dan fitur unggulan *custom* paket perjalanan dapat sesuai dengan keinginan atau kebutuhan. Supaya bisnis *travel agent* terlebih untuk Tenta Tour dapat menjalankan bisnisnya dengan lebih baik, praktis dan dapat menariknya banyak *customer*. Untuk proses pengolahan datanya juga menjadi lebih baik dengan penerapan REST API sehingga untuk data-data nantinya juga dapat digunakan untuk pengembangan seperti *website* atau aplikasi-aplikasi lainnya.

Pada penelitian sebelumnya disebutkan bahwa aplikasi Tour Agent di Indonesia masih memiliki kekurangan pada sisi pemesanan. Dimana paket yang ditawarkan oleh *tour agent* bersifat mengikat dari segi jadwal, makanan, dan tempat yang dikunjungi. Oleh karena itu, *traveller* menjadi tidak leluasa dalam menjelajahi objek wisatanya. Berdasarkan hal tersebut, dibuatnya sebuah aplikasi yakni *vaithme* dengan berbasis Android menggunakan bahasa pemrograman Java dan untuk pengolahan datanya menggunakan *library* Retrofit. Agar dengan adanya fitur-fitur

yang diberikan seperti *custom* trip untuk rekomendasi paket wisata yang sesuai dengan kriteria dan minat *traveller* dalam menentukan pilihan berwisatanya [7].

Berdasarkan penelitian sebelumnya terdapat kasus yang serupa dengan yang diteliti penulis. Serta tujuan yang serupa untuk memunculkan aplikasi pemesanan pada *Travel Agent* yang bersifat fleksibel dengan fitur *custom* sehingga *customer* atau *traveller* dapat melakukan penyesuaian kriteria dan minatnya dalam berwisata. Pada penelitian sebelumnya juga digunakan bahasa pemrograman *Java* untuk membuat aplikasi Android dan Retrofit untuk REST API web *service* sehingga dapat terjadinya komunikasi antar aplikasi serta pengolahan datanya. Dengan adanya aplikasi berbasis android tersebut dapat memberikan keleluasaan dalam melakukan pemesanan dan *customize* perjalanan wisata. Selain itu juga, dengan implementasi retrofit sebagai teknologi REST API dapat berlangsung dengan baik dalam pembuatan sistem berbasis Android tersebut.

Selanjutnya pada penelitian sebelumnya dijelaskan bahwa Android adalah sistem operasi yang cerdas dan *open source* pada *smartphone* yang juga menyediakan *platform* terbuka untuk *developer* membuat sebuah aplikasi. Sekarang ini *handphone* tidak hanya sebagai alat komunikasi saja, dengan bantuan koneksi internet fungsi *handphone* menjadi semakin bervariasi bahkan dapat melakukan transaksi dengan media *handphone* tersebut. Sekarang ini *handphone* sendiri merupakan salah satu perangkat atau kebutuhan wajib dimiliki setiap orang [8].

Sesuai dengan yang sudah dijelaskan sebelumnya oleh penulis, perangkat *handphone* sudah banyak dimiliki oleh setiap orang. Dengan *handphone* yang berbasis Android juga dapat membantu *developer* dalam membuat aplikasi. Selain itu, adanya koneksi internet juga dapat membuat aplikasi melakukan berbagai hal dan juga mengakses segala media. Dengan banyaknya orang yang memiliki *handphone* berbasis android maka dalam penelitian ini ditentukan pembuatan aplikasi yang berbasis android dengan bantuan akses internet juga. Supaya dapat diaksesnya juga data-data dengan mudah untuk mendukung pembuatan fitur-fitur yang ada nantinya.

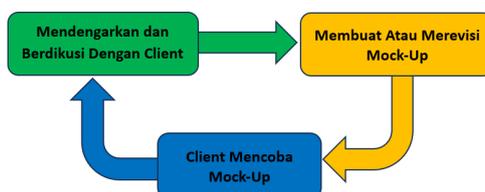
Kemudian pada penelitian yang berjudul Pertukaran Data Pada Aplikasi Android Menggunakan *Java Script Object Notation* (JSON) dan REST API Dengan retrofit2 [9] disebutkan bahwa retrofit2 berperan sebagai REST *Client* serta *library* HTTP *Client* yang aman untuk aplikasi Android dan *Java*. Dengan menggunakan retrofit, tidak perlu lagi membuat metode-metode sendiri untuk terhubung ke layanan web REST. GSON sebagai bagian dari retrofit merupakan pustaka untuk mengubah objek *Java* ke dalam format JSON dan begitu juga sebaliknya. Dengan retrofit2 dapat mempermudah proses pertukaran data dari format JSON, yang kemudian diurai menjadi *Plain Old Java Objects* (POJOs). Proses permintaan yang dapat dijalankan pun lengkap sesuai dengan arsitektur REST sehingga dapat dieksekusi hanya dengan melalui *library* retrofit saja.

Memang dengan cukup terkenalnya Retrofit serta sering diterapkan dan digunakannya retrofit sebagai *library* REST *Client* yang juga cukup mudah dan sangat sesuai dengan bahasa pemrograman *java* terlebih lagi cocok dalam mengembangkan aplikasi berbasis android. Berdasarkan uraian terkait Retrofit tersebut, kami menerapkan serta menggunakan Retrofit karena berbagai kelebihan yang sangat sesuai dan mendukung dalam pembuatan aplikasi yang diinginkan.

II. METODE PENELITIAN

A. Metode Penelitian

Pada penelitian ini menggunakan metode penelitian *prototype model*. *Prototype model* adalah salah satu metode yang banyak digunakan dalam pengembangan perangkat lunak yang melibatkan interaksi antara pengembang dan *client* dalam proses pembuatan sistem/aplikasi [10], [11]. Metode penelitian pada penelitian ini dibuat menjadi 3 tahapan seperti Gambar 1 berikut ini:

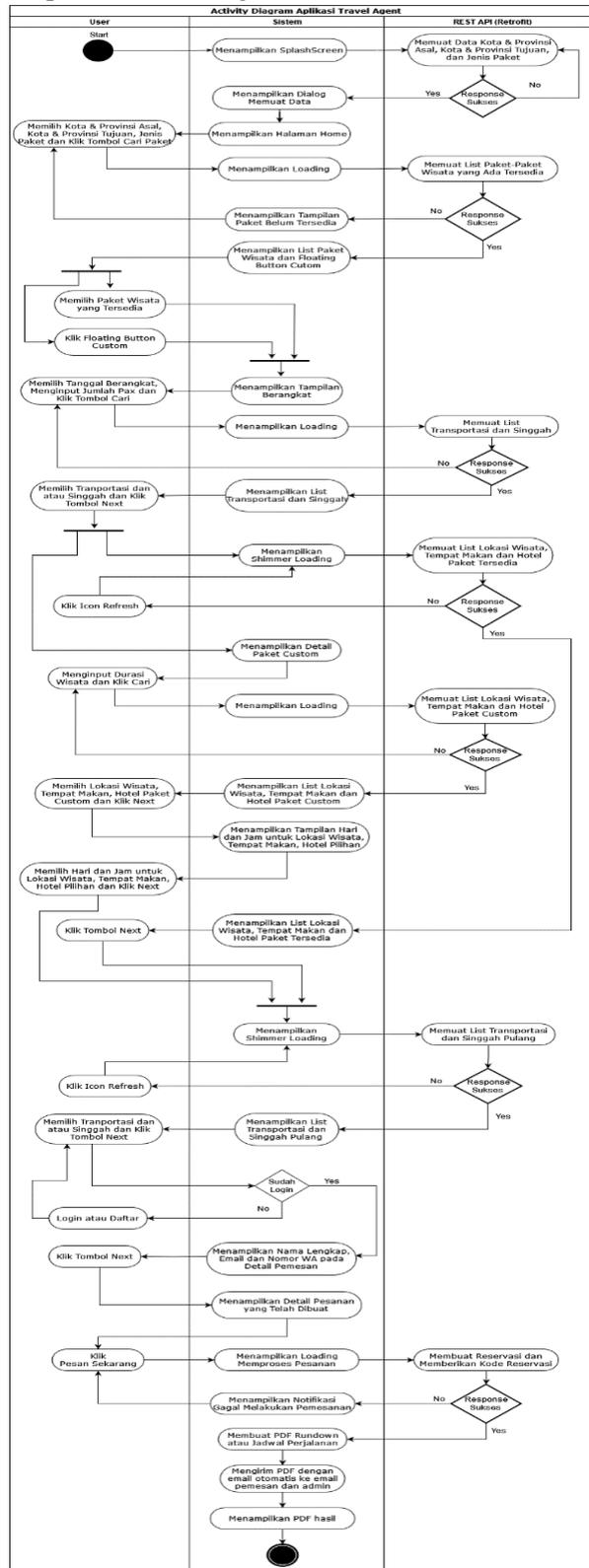


Gambar 1. Metode Penelitian *Prototype model*

Prototype Model pada penelitian ini diawali dengan mendengarkan dan berdiskusi dengan *client*. *Client* pada penelitian ini adalah *travel agent* Tenta Tour. Dilakukannya diskusi untuk membahas permasalahan, kebutuhan dan keinginan dari pihak *client*. Selanjutnya barulah dilakukan perancangan *mock-up* dari sistem atau aplikasi hasil representasi pada diskusi sebelumnya. Lalu *client* dapat mencoba *mock-up* yang sudah dibuat tersebut dan menilai apakah sudah sesuai atau perlu dilakukannya evaluasi kembali. Jika saat mencoba *Mock-Up client* merasa perlu dilakukannya evaluasi, maka dilakukan kembali diskusi dan revisi *Mock-Up* hingga hasilnya sesuai dengan kebutuhan dan keinginan dari pihak *client*.

B. Perancangan Sistem

Metode yang digunakan dalam perancangan proses aplikasi Travel Agent menggunakan metode *Activity Diagram*. *Activity Diagram* adalah semantik semi formal spesifikasi yang intuitif dan fleksibel. *Activity Diagram* digunakan untuk mendeskripsikan perilaku *system* dan logika internal kompleks operasi [12]. Berikut ini merupakan desain *Activity Diagram* dari aplikasi Travel Agent.



Gambar. 2. Activity Diagram Aplikasi Travel Agent

C. Analisis Hasil Pengumpulan Data

Untuk analisis hasil pengumpulan data yang digunakan pada penelitian ini adalah System Usability Scale (SUS). Analisis dengan SUS berfungsi untuk salah satunya mengukur kegunaan dari sebuah sistem dan produk perangkat lunak seperti *website* dan aplikasi *mobile*. SUS dapat digunakan sebagai kuesioner *post-test* untuk penilaian kegunaan yang dirasakan. SUS terdiri dari sepuluh item kuesioner skala *Likert* dengan lima tanggapan untuk setiap pertanyaan yaitu "sangat setuju" dengan skor 5, "setuju" dengan skor 4, "netral" dengan skor 3, "tidak setuju" dengan skor 2, dan "sangat tidak setuju" dengan skor 1. Tujuan utama SUS adalah membantu untuk mengumpulkan data kuantitatif tentang pendapat *user* dan mengukur "persepsi subyektif orang" dengan cepat dan mudah. Untuk pertanyaan-pertanyaan pada kuesioner terhadap aplikasi Travel Agent adalah sebagai berikut[13], [14].

TABEL 1
 Tabel Pertanyaan Kuesioner terhadap Aplikasi Travel Agent

No	Pertanyaan
1	Saya akan lebih sering lagi untuk menggunakan aplikasi ini.
2	Saya merasa aplikasi ini tidak seharusnya dibuat untuk serumit ini.
3	Saya merasa aplikasi ini mudah untuk digunakan.
4	Saya membutuhkan para teknisi atau orang lain untuk membantu menggunakan aplikasi ini.
5	Saya merasa fitur – fitur yang ada di dalam aplikasi ini telah terintegrasi dengan baik.
6	Saya telah menemukan adanya ketidaksesuaian di dalam antar aplikasi ini.
7	Saya merasa orang lain akan mudah dan cepat untuk mempelajari aplikasi ini dengan baik.
8	Saya merasa aplikasi ini sangat rumit untuk digunakan.
9	Saya merasa percaya diri untuk menggunakan aplikasi ini dengan baik.
10	Saya merasa perlu mempelajari aplikasi ini sebelum saya akan menggunakannya nanti.

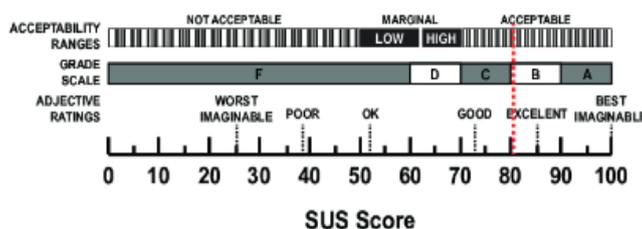
Selanjutnya akan dilakukannya penghitungan skor dari responden kuesioner. Dengan rumus perhitungannya seperti gambar 3 berikut ini.

$$\bar{x} = \frac{\sum x}{n}$$

\bar{x} = skor rata-rata
 $\sum x$ = jumlah skor SUS
 n = jumlah responden

Gambar. 3. Rumus Perhitungan SUS

Kemudian berdasarkan perhitungan tersebut akan diperoleh rata-rata skor SUS. Dengan demikian, dapat dilihat indikator tingkatan metode SUS dapat dilihat pada gambar 4 berikut ini [15].



Gambar. 4. Indikator Hasil Perhitungan Rata-Rata Skor SUS

III. HASIL DAN PEMBAHASAN

A. Hasil Diskusi dengan Client

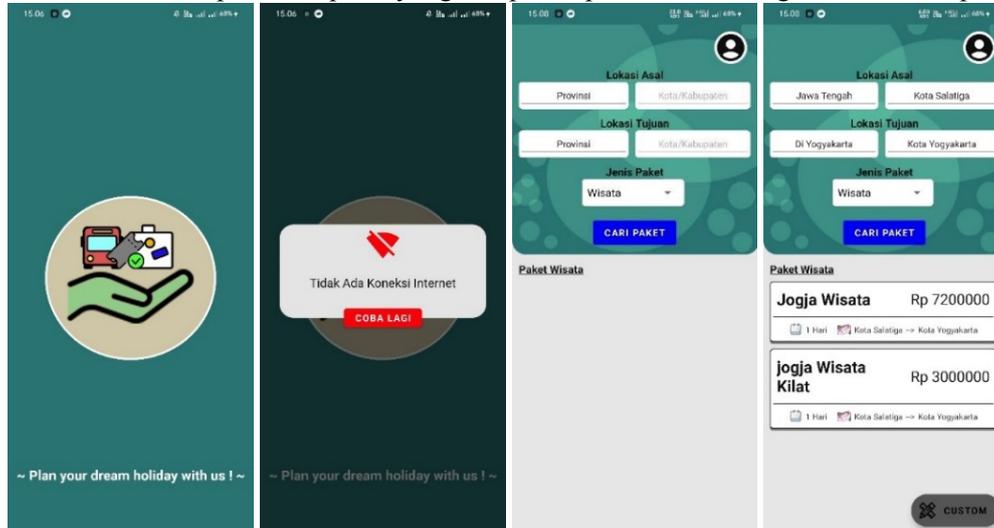
Hasil diskusi dengan pihak *client* atau Tenta Tour adalah aplikasi android yang dapat menggambarkan atau menginformasikan paket wisatanya. Berdasarkan hal tersebut terdapat beberapa poin utama yaitu

1. Terdapat lokasi awal dan lokasi tujuan dari pemesan
2. Memuat 3 tahapan penting yaitu Bagian Pertama yaitu Berangkat, Bagian Utama yaitu Detail Paket Wisata dan Bagian Terakhir yaitu Pulang.
3. Memberikan opsi *custom* paket wisata kepada *customer* atau *traveler*

4. Menghasilkan output akhirnya adalah PDF *rundown* atau jadwal perjalanan yang dipesan

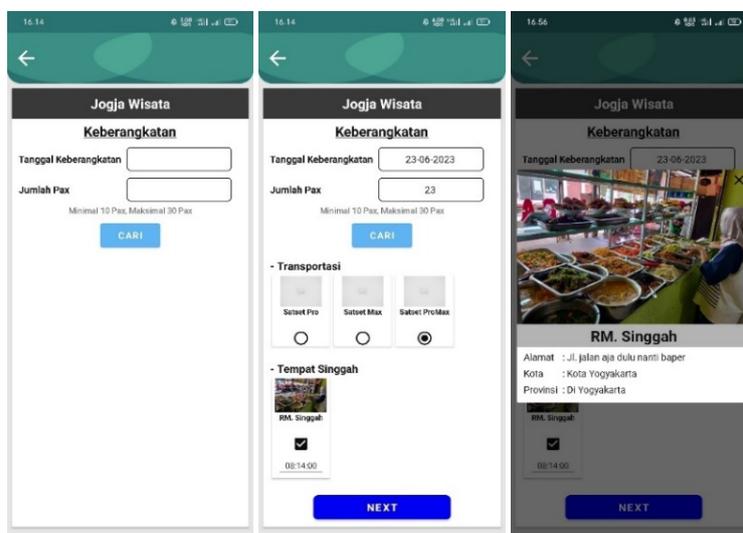
B. User Interface Aplikasi Travel Agent

User Interface atau antarmuka pengguna adalah interaksi antara pengguna dan sistem atau perangkat lunak yang mencakup elemen-elemen visual dan interaktif seperti tombol dan menu [16]. Oleh sebab itu, pada bagian ini akan dibahas terlebih dahulu tampilan-tampilan yang ada pada aplikasi Travel Agent dari sudut pandang *user*.



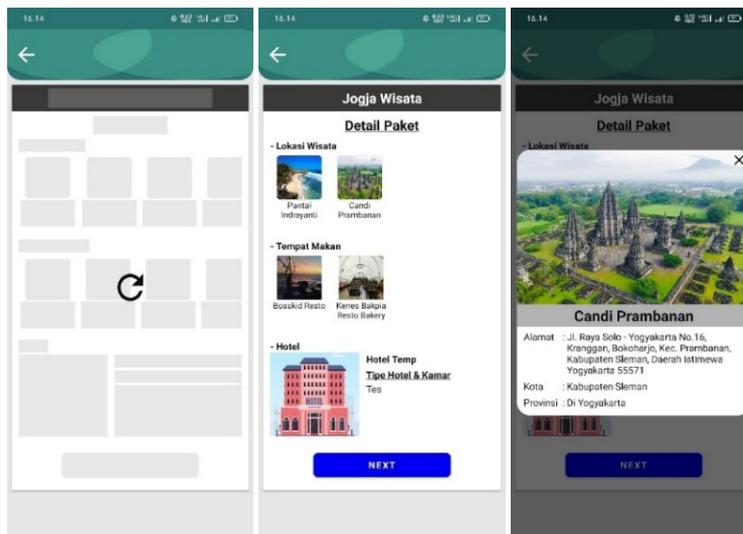
Gambar. 5. Tampilan *Splash Screen* dan *Home*

Tampilan *Splash Screen* yang ditunjukkan pada Gambar 5 menampilkan Logo dan *Tagline* dari aplikasi ini dan akan dilakukan pengecekan, apakah perangkat android sudah terhubung atau memiliki jaringan *internet* dikarenakan untuk memproses data-data yang diperlukan tampilan-tampilan selanjutnya akan menggunakan API. Jika tidak ada koneksi *internet* pada perangkat android tersebut maka akan ditampilkan dialog peringatan, apabila jika ada koneksi *internet* maka akan ditampilkannya halaman *Home*. Untuk tampilan *Home* berisikan inputan provinsi dan kota untuk lokasi asal, inputan provinsi dan kota untuk lokasi tujuan, *dropdown* jenis paket, dan tombol untuk mencari paket yang ada. *User* diharuskan untuk memilih semua inputan lokasi asal, lokasi tujuan dan jenis paket untuk dapat memperoleh *list* paket-paket yang ada. Jika paket tersedia atau sesuai dengan lokasi asal dan lokasi tujuan maka akan ditampilkan *list* paket-paketnya. *List* paket-paket yang ada akan berisi nama paket, harga paket, durasi wisata, dan lokasi asal-lokasi tujuan paket. *User* dapat memilih atau mengklik paket yang sesuai dengan keinginannya dan kemudian akan ditampilkan halaman Berangkat seperti pada Gambar 6 berikut ini. Selain itu, terdapat *icon* akun yang jika diklik akan menuju ke halaman *account* seperti pada Gambar 12.



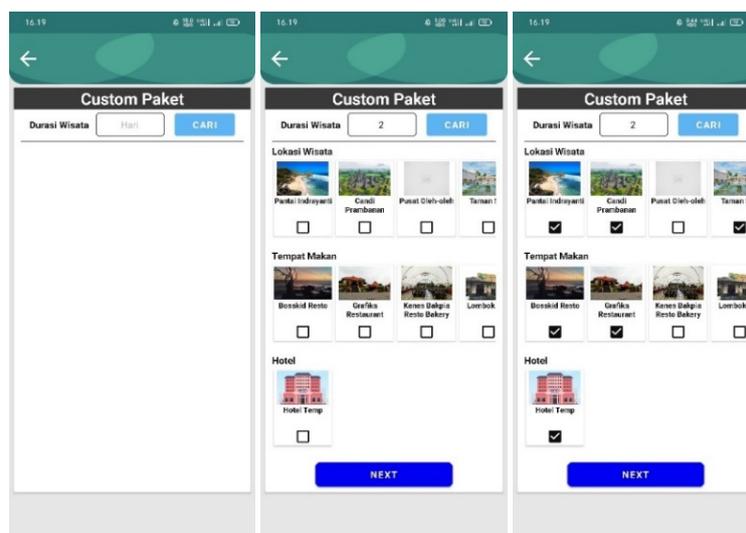
Gambar. 6. Tampilan Berangkat

Untuk tampilan Berangkat yang ditunjukkan pada Gambar 6 berisikan inputan Tanggal Keberangkatan dan Jumlah Pax. Dimana *user* diminta untuk memilih Tanggal Keberangkatan dari tampilan *popup Calendar* dan menginputkan Jumlah Pax yang diperlukan. Lalu dapat mengklik tombol cari untuk memperoleh *list* data Transportasi dan Tempat Singgah. Untuk transportasi *user* diharuskan untuk memilih satu kendaraan, namun untuk tempat singgah *user* tidak harus memilihnya atau bersifat opsional saja. Jika *user* memilih singgah maka akan muncul inputan jam yang dimana *user* juga harus memilih jam dari *popup Time*. *User* juga dapat mengklik *card list* transportasi atau singgah untuk *popup* detailnya. Jika sudah sesuai maka *user* dapat mengklik *Next* jika sebelumnya *user* memilih paket yang tersedia akan ditampilkan Detail Paket Wisata Tersedia seperti pada Gambar 7, namun jika *user* memilih paket *custom* akan ditampilkan Tampilan Detail Paket *Custom* seperti pada Gambar 8.



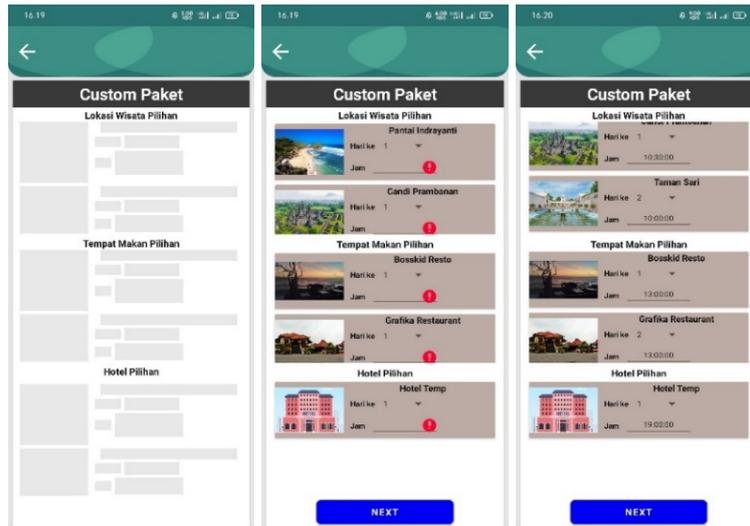
Gambar. 7. Tampilan Detail Paket Wisata Tersedia

Untuk tampilan Detail Paket Wisata Tersedia yang ditunjukkan pada Gambar 7 berisikan list lokasi wisata, list tempat makan dan hotel. *User* dapat mengetahui rincian dari lokasi wisata atau tempat makan tersebut dari detail *popup*. Kemudian jika paket sudah sesuai atau cocok, *user* dapat mengklik tombol *Next*. Setelah itu akan ditampilkan halaman pulang seperti pada Gambar 10.



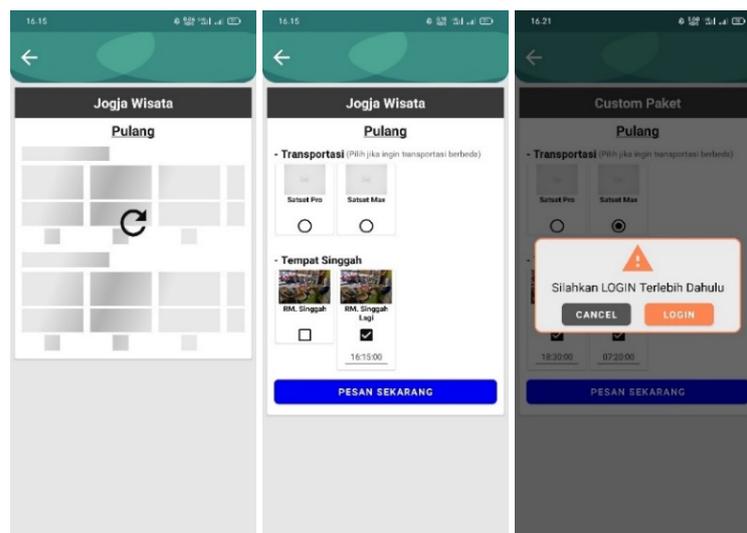
Gambar. 8. Tampilan Detail Paket Custom

Untuk tampilan Detail Paket *Custom* yang ditunjukkan pada Gambar 8 *user* diminta untuk menginputkan durasi wisata atau lama wisata dalam satuan Hari. Setelah itu *user* dapat mengklik tombol cari yang akan menampilkan *list* lokasi-lokasi wisata, *list* tempat-tempat makan, dan *list* hotel. *User* dapat memilih sesuai keinginan dengan batasan berdasarkan jumlah hari yang diinginkan dengan diatur maksimal per hari hanya dapat dua lokasi wisata, satu tempat makan dan satu hotel. Setelah sesuai keinginan, *user* dapat mengklik tombol *Next*. Setelah itu akan ditampilkan halaman Hari dan Jam Paket Custom Pilihan seperti pada Gambar 9.



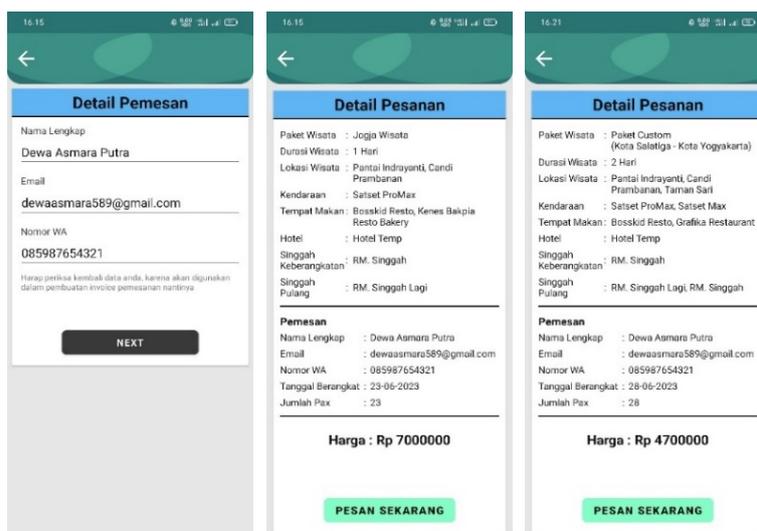
Gambar. 9. Tampilan Hari dan Jam Paket *Custom* Pilihan

Untuk tampilan Detail Paket *Custom* yang ditunjukkan pada Gambar 9 *user* harus memilih hari dari *popup Calendar* dan jam dari *popup Time* untuk Lokasi Wisata, Tempat Makan dan Hotel yang sudah dipilih sebelumnya. Hari dan Jam tersebut akan digunakan untuk membuat PDF jadwal atau *rundown* perjalanan. Kemudian jika Hari dan Jam sudah dipilih atau diisikan semua sesuai keinginan *user* maka dapat mengklik tombol Next. Setelah itu akan ditampilkan halaman pulang seperti pada Gambar 10.



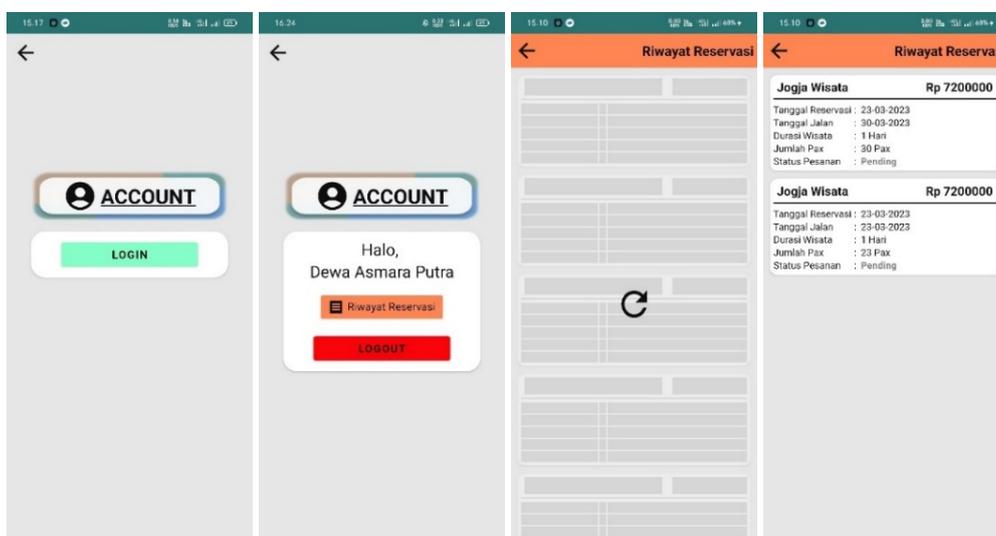
Gambar. 10. Tampilan Pulang

Untuk tampilan Pulang yang ditunjukkan pada Gambar 10 berisikan *list* data Transportasi dan Tempat Singgah. *User* tidak diharuskan memilih atau saat *user* menginginkan berganti kendaraan dan melakukan singgah pulang saja. Jika *user* memilih singgah maka harus memilih jam dari *popup Time*. *User* juga dapat mengetahui lebih detail dari transportasi atau singgah yang diinginkan dengan mengkliknya kemudian akan ditampilkan *popup* detail. Kemudian jika sudah maka *user* dapat mengklik *Next* dan sistem akan mengecek terlebih dahulu apakah *user* sudah melakukan *Login*. Jika *user* belum melakukan *Login* maka akan muncul Dialog untuk dapat melakukan *Login* atau Daftar seperti Gambar 13, namun jika *user* sudah melakukan *Login* maka akan ditampilkan Detail Pemesan seperti pada Gambar 11.



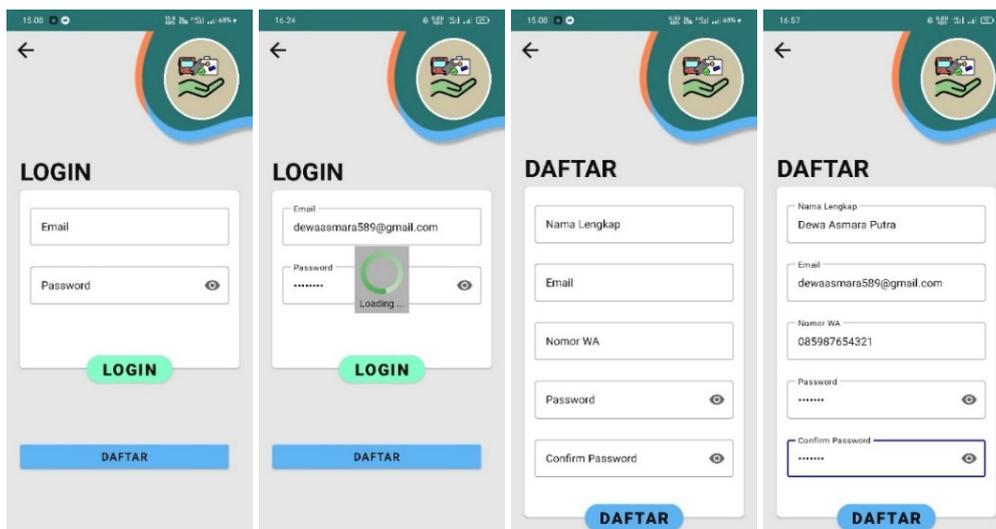
Gambar. 11. Tampilan Detail Pesanan

Untuk tampilan Detail Pemesanan yang ditunjukkan pada Gambar 11, data-data seperti nama lengkap, *email* dan nomor wa sudah terisi secara otomatis berdasarkan data-data akun yang digunakan *user* untuk *Login*. Disini *user* dapat melakukan perubahan nama lengkap, *email* dan nomor wa jika masih belum sesuai. Namun jika sudah sesuai maka *user* dapat mengklik tombol *Next*, kemudian akan menuju ke tampilan Detail Pesanan. Untuk tampilan Detail Pesanan, *user* akan diberikan rekap atau detail semua data-data paket dan pemesanan. Kemudian jika sudah sesuai, *user* dapat mengklik tombol *Pesan Sekarang*. Sistem akan membuat reservasi, menyimpan data-data ke dalam API, membuat PDF jadwal perjalanan sesuai pesanan, dan mengirimkan PDF tersebut ke Email Pemesan dan Email Admin seperti Gambar 14 untuk nantinya digunakan juga menjadi notifikasi dan rekap oleh *Admin* Travel Agent.



Gambar. 12. Tampilan Account

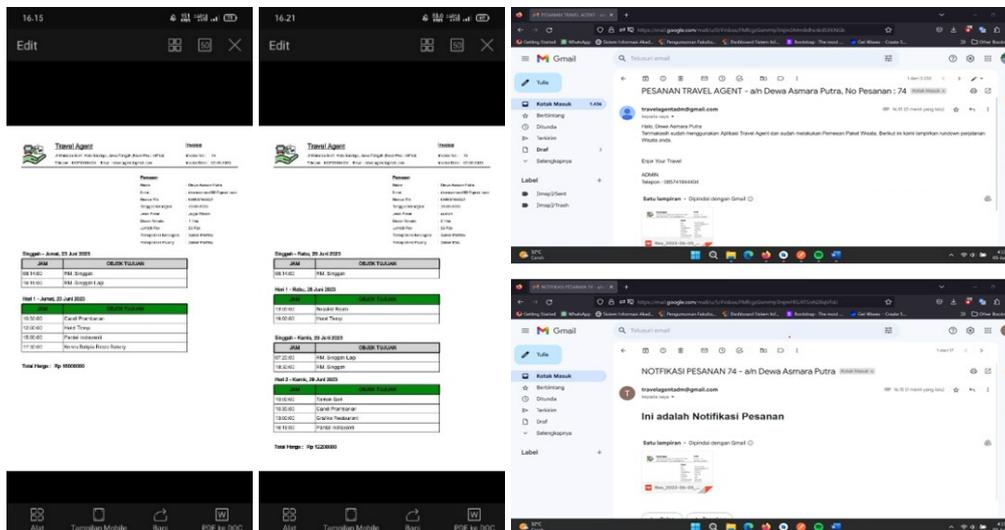
Untuk Tampilan *Account* seperti yang ditunjukkan pada Gambar 12, memiliki perbedaan dimana jika *user* belum melakukan *login* maka akan menampilkan tombol *login* saja yang jika diklik akan menuju ke halaman *login* seperti pada Gambar 13. Namun jika *user* sudah melakukan *login* akan muncul nama dari akun dan riwayat reservasi. Dimana jika *user* mengklik riwayat reservasi akan menuju ke tampilan Riwayat Reservasi. Sementara itu, Tampilan Riwayat Reservasi akan ditampilkan *list* data pemesanan yang telah dilakukan atau dibuat oleh *user* berdasarkan akun yang sudah digunakan untuk *login* sebelumnya. Dengan demikian, *list* data-data pemesanan yang ada akan ditampilkan berisikan informasi Nama Paket, Harga, Tanggal Reservasi, Tanggal Jalan, Durasi Wisata, Jumlah Pax dan Status Pesanan.



Gambar. 13. Tampilan Login dan Daftar

Untuk tampilan *Login* yang ditunjukkan pada Gambar 13, *user* dapat melakukan login dengan mengisi *email* serta *password* dari akun yang dimiliki lalu mengklik tombol *Login*. Namun jika belum memiliki akun, maka *user* dapat mendaftar dengan mengklik tombol *Daftar* yang nantinya akan menampilkan halaman *Daftar*. Sementara itu, pada tampilan *Daftar* *user* diminta untuk mengisi beberapa hal yang dibutuhkan seperti nama lengkap, *email*, nomor wa, *password* dan *confirm password*. Apabila sudah sesuai maka dapat mengklik tombol *Daftar*. Jika berhasil akan ditampilkan kembali halaman *Login* untuk *user* dapat melakukan *login*, Namun jika gagal akan tetap berada pada tampilan *daftar* dan dapat melihat beberapa kesalahan atau *error* yang terjadi.

Berikut ini adalah beberapa contoh hasil pesanan yang dilakukan *user* yakni Tampilan PDF Jadwal Hasil Pesanan dan otomatis send *Email* ke Pemesan & *Admin* yang ditunjukkan pada Gambar 14.



Gambar. 14. Tampilan PDF Jadwal Hasil Pesanan dan otomatis send Email ke Pemesan & Admin

C. Pengujian Sistem Aplikasi Travel Agent

Untuk pengujian sistem yang dilakukan pada penelitian ini akan menggunakan metode *user acceptance testing* (UAT) sebagai pengujian perangkat lunak atau sistem dapat berfungsi sesuai dengan fungsionalitasnya dan sesuai dengan apa yang diharapkan [17], [18]. Dengan tipe yang dipilih adalah *Black Box Testing*. Berikut ini adalah hasil *Black Box Testing* pengujian sistem pada setiap *activity* yang ada pada aplikasi *Travel Agent*.

TABEL 2
 Tabel *Black Box Testing* Pengujian Sistem aplikasi Travel Agent

Deskripsi Pengujian	Yang Diharapkan	Kesimpulan
<i>Splash Screen</i> dan <i>Pop up Dialog</i> Tidak Ada Koneksi Internet pada <i>Splash Screen</i>	Menampilkan animasi Logo dan <i>Tagline</i> dari aplikasi Travel Agent serta mengecek koneksi internet.	Berhasil
<i>Dialog Loading</i> memuat data pada Halaman <i>Home</i>	Menampilkan <i>Dialog Loading</i> memuat data untuk proses mengakses data API kota & provinsi asal, kota & provinsi tujuan, dan jenis paket.	Berhasil
Pemilihan <i>list</i> kota & provinsi asal, <i>list</i> kota & provinsi tujuan, jenis paket dan mengklik tombol cari pada Halaman <i>Home</i>	Menampilkan <i>list dropdown</i> provinsi asal dan provinsi tujuan saat mengklik atau menginputkan kata pada inputan provinsi asal. Memilih provinsi asal dan provinsi tujuan akan mengaktifkan inputan kota asal dan kota tujuan. Menampilkan pilihan <i>list dropdown spinner</i> untuk jenis paket. Lalu saat mengklik cari akan menampilkan <i>loading</i> untuk proses mencari data <i>list</i> paket wisata tersedia pada API. Jika <i>response</i> sukses akan ditampilkan <i>recyclerview</i> list paket wisata tersedia dan <i>floating button custom</i> , namun jika tidak sukses atau data tidak ada maka menampilkan <i>imageview</i> paket wisata belum tersedia.	Berhasil
Memilih <i>list</i> paket wisata tersedia atau <i>Floating button Custom</i> pada Halaman <i>Home</i>	Memilih <i>list</i> paket akan menyimpan data dan kode paket wisata tersedia untuk detail paket wisata yang tersedia. Sedangkan <i>Floating button</i> akan menyimpan jenis data <i>custom</i> yang dipilih untuk detail paket wisata <i>custom</i> . Lalu menampilkan Halaman Berangkat.	Berhasil
Menginputkan tanggal berangkat, jumlah pax dan Mengklik Tombol Cari pada Tampilan Berangkat pada Tampilan Berangkat	Menampilkan <i>pop up calendar</i> saat mengklik inputan tanggal berangkat dan hasil pilihan tanggal akan tampil pada inputan tanggal berangkat. Sedangkan untuk jumlah pax dapat diinputkan dengan jenis angka saja. Saat mengklik tombol cari akan mengecek inputan tanggal dan jumlah pax apakah sudah diisi atau kosong. Jika kosong akan muncul <i>warning</i> pada inputan tanggal atau jumlah pax, akan tetapi jika sudah diisi akan menampilkan <i>loading</i> untuk mencari data API kendaraan dan singgah.	Berhasil
Memilih Transportasi (wajib memilih) dan Memilih Singgah (opsional atau tidak wajib memilih) dan Mengklik Tombol <i>Next</i> pada Tampilan Berangkat	Mengklik <i>radiobutton</i> untuk memilih transportasi dan <i>checkbox</i> akan aktif sesuai pilihan. Mengklik <i>checkbox</i> untuk memilih singgah dan <i>checkbox</i> singgah akan aktif sesuai pilihan dan menampilkan inputan jam. <i>Pop up time</i> muncul saat mengklik inputan jam singgah dan dapat dipilih jam singgah yang kemudian akan tampil hasil pilihan pada inputan jam singgah. Jika mengklik <i>recyclerview</i> transportasi atau singgah maka akan tampil <i>pop up</i> detail dari transportasi atau singgah. Mengklik <i>Next</i> akan cek sudah melakukan pemilihan transportasi atau belum dan cek sudah mengisi jam saat memilih paket singgah. Jika belum makan akan muncul notifikasi peringatan, sedangkan jika sudah maka akan mengecek sebelumnya memilih paket tersedia atau <i>custom</i> kemudian menampilkan Halaman Detail Paket Tersedia atau Detail Paket <i>Custom</i> .	Berhasil
Menampilkan <i>loading shimmer</i> , Mengklik <i>recyclerview list</i> , dan Mengklik Tombol <i>Next</i> pada Tampilan Detail Paket Wisata Tersedia	Menampilkan <i>loading shimmer</i> saat mencari data API lokasi wisata, tempat makan dan hotel sesuai dengan pilihan paket wisata. Jika <i>response</i> sukses maka akan menampilkan <i>list recyclerview</i> lokasi wisata dan tempat makan, dan menampilkan <i>imageview</i> dan detail dari hotel. Mengklik <i>recyclerview list</i> akan menampilkan <i>pop up</i> detail lebih rinci dari lokasi wisata atau tempat makan yang diklik. Mengklik Tombol <i>Next</i> akan menampilkan halaman Pulang.	Berhasil
Menginputkan durasi wisata Tampilan dan Mengklik tombol Cari Detail Paket Wisata <i>Custom</i>	Menginputkan durasi wisata dalam satuan hari dengan tipe inputan angka saja. Mengklik tombol cari akan menampilkan <i>Loading</i> untuk memproses data API list lokasi wisata, tempat makan dan hotel. Jika <i>response</i> sukses maka akan tampil <i>recyclerview</i> lokasi wisata, tempat makan dan hotel.	Berhasil
Memilih, Mengklik <i>recyclerview list</i> lokasi wisata, tempat makan dan hotel, dan Mengklik Tombol <i>Next</i> Tampilan Detail Paket Wisata <i>Custom</i>	Mengklik <i>checkbox</i> untuk memilih lokasi wisata, tempat makan dan hotel singgah. <i>Checkbox</i> akan aktif sesuai pilihan, selama belum melebihi batas pilihan berdasarkan hari. Mengklik <i>recyclerview</i> akan menampilkan <i>pop up</i> detail lebih rinci dari lokasi wisata, tempat makan, dan hotel yang diklik. Mengklik tombol <i>Next</i> akan mengecek apakah sudah memilih lokasi wisata, tempat makan dan hotel. Jika belum akan muncul notifikasi, sedangkan jika sudah akan menampilkan halaman Pulang.	Berhasil
Menampilkan <i>loading shimmer</i> dan Mengklik Tombol <i>Next</i> pada Tampilan Hari dan Jam Paket <i>Custom</i> Pilihan	Menampilkan <i>loading shimmer</i> saat memproses lokasi wisata, tempat makan dan hotel sesuai dengan pilihan pada detail lokasi wisata <i>custom</i> . Akan ditampilkan <i>recyclerview</i> data lokasi wisata, tempat makan dan hotel sesuai pilihan dengan adanya <i>spinner</i> hari dan inputan jam. Mengklik tombol <i>Next</i> akan mengecek apakah sudah memilih hari dan menginputkan jam pada semua data lokasi wisata, tempat makan dan hotel pilihan. Jika belum akan muncul notifikasi <i>warning</i> , sedangkan jika sudah akan menampilkan halaman Pulang.	Berhasil
Memilih Transportasi, Memilih Singgah (opsional atau tidak wajib memilih) dan Mengklik Tombol <i>Next</i> pada Tampilan Pulang pada Tampilan Pulang	Mengklik <i>radiobutton</i> untuk memilih transportasi dan <i>checkbox</i> akan aktif sesuai pilihan. Mengklik <i>checkbox</i> untuk memilih singgah dan <i>checkbox</i> singgah akan aktif sesuai pilihan dan menampilkan inputan jam. <i>Pop up time</i> muncul saat mengklik inputan jam singgah dan dapat dipilih jam singgah yang kemudian akan tampil hasil pilihan pada inputan jam singgah. Jika mengklik <i>recyclerview</i> singgah atau transportasi maka akan tampil <i>pop up</i> detail dari singgah atau transportasi. Saat mengklik tombol <i>Next</i> akan mengecek jika memilih singgah dan belum mengisikan jam maka akan muncul notifikasi peringatan. Serta mengecek apakah sudah melakukan <i>login</i> atau belum, jika belum maka akan muncul dialog untuk melakukan <i>Login</i> . Jika sudah maka akan menampilkan halaman Detail Pesanan.	Berhasil
Inputan Data Pemesan (Nama Lengkap, Email dan Nomor WA) dan Mengklik Tombol <i>Next</i> pada Tampilan Detail Pemesan	Menampilkan inputan Nama Lengkap, inputan <i>Email</i> dan inputan Nomor WA yang sudah otomatis terisi dengan data pemesan yang digunakan untuk <i>Login</i> . Dapat dilakukan perubahan jika terdapat data yang salah atau tidak sesuai. Mengklik tombol <i>Next</i> akan menampilkan halaman Detail Pesanan.	Berhasil
Menampilkan <i>loading shimmer</i> dan Mengklik Tombol <i>Next</i> pada Tampilan Detail Pesanan	Menampilkan <i>loading shimmer</i> saat memproses rekap atau detail semua data-data sesuai paket wisata pilihanda atau paket wisata <i>custom</i> yang dibuat dan data-data pemesan sesuai. Mengklik tombol <i>Next</i> akan menampilkan <i>Loading</i> membuat pesanan. Menampilkan PDF rundown atau jadwal perjalanan dan <i>auto send email</i> ke pemesan dan <i>admin</i> .	Berhasil

D. Hasil Pengumpulan Data dengan metode SUS

Hasil analisis pengumpulan data dengan metode SUS menggunakan cara pengisian kuisioner dengan penilaian dan pertanyaan-pertanyaan sesuai dengan yang sudah dijelaskan pada metode penelitian. Target dari kuisioner tersebut adalah pihak *client* (Tenta Tour) dan beberapa *user* yang dapat diartikan sebagai *customer* yang melakukan pemesanan paket wisata dari aplikasi Travel Agent. Berikut ini adalah jawaban para *responden* dan rata-rata nilai yang diperoleh berdasarkan pertanyaan-pertanyaan pada metode SUS.

TABEL 3
 Tabel *Black Box Testing* Pengujian Sistem aplikasi Travel Agent

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Jumlah	Nilai
Responden 1	3	4	4	4	4	4	3	4	4	3	37	93
Responden 2	3	3	3	1	3	2	4	3	4	2	28	70
Responden 3	4	2	4	2	4	3	4	3	4	2	32	80
Responden 4	4	2	4	3	4	3	3	3	3	2	31	78
Responden 5	3	3	4	2	4	4	3	3	4	1	31	78
Responden 6	3	2	3	2	4	3	2	2	3	2	26	65
Responden 7	4	2	4	2	4	2	4	2	4	1	29	73
Responden 8	2	0	4	4	3	2	4	4	3	4	30	75
Responden 9	3	2	3	2	4	2	3	2	4	3	28	70
Responden 10	3	3	3	2	3	3	3	3	3	1	27	68
Responden 11	2	2	3	3	3	3	3	2	2	3	26	65
Responden 12	3	4	3	2	3	3	4	3	4	1	30	75
Responden 13	4	4	4	0	4	4	4	4	4	2	34	85
Responden 14	2	2	3	3	3	3	3	3	2	1	25	63
Responden 15	4	3	4	2	4	2	3	3	3	1	29	73
Skor Rata-rata (Hasil Akhir)											74	

Hasil akhir yang diperoleh berdasarkan metode SUS tersebut mendapatkan skor sebesar 74. Oleh sebab itu, aplikasi Travel Agent memperoleh kategori penilaian *Good* dengan grade B. Dengan demikian berdasarkan penilaian yang diperoleh dengan metode SUS tersebut sudah layak digunakan dan telah dapat diterima dengan baik.

IV. KESIMPULAN

Berdasarkan pembahasan yang ada, telah dapat dihasilkannya sebuah aplikasi *Travel Agent* berbasis android dengan diterapkannya juga metode REST API. Dengan sistem pemesanan yang dapat dilakukan oleh *customer* dan Tenta Tour secara praktis, mudah dan *online* hanya dengan menggunakan *smartphone* android saja. Serta aplikasi tersebut memberikan fitur unggulan untuk dapat melakukan *custom* transportasi, lokasi wisata, tempat makan, dan hotel sesuai keinginan dan kebutuhan atau *customer* dapat juga memesan sebuah paket wisata baik yang sudah ada. Aplikasi Travel Agent akan menyimpan data *reservasi* atau pemesanan *customer* secara langsung melalui API bahkan mengirimkan pesan otomatis ke *customer* dan *admin*. Dengan demikian, segala data pesanan dan pengolahan datanya dapat dikelola dengan baik dan dapat digunakan juga dalam pengembangan *website* atau aplikasi lain dengan hanya mengakses API yang sama tersebut. Serta berdasarkan data *black box testing* dan penilaian dengan metode SUS menunjukkan bahwa fitur-fitur pada Aplikasi Travel Agent dapat berjalan sebagaimana mestinya serta layak digunakan dan diterima dengan baik oleh pihak *client* yakni Tenta Tour dan beberapa *responden* yang dapat artikan sebagai *customer*.

DAFTAR PUSTAKA

- [1] S. Syaharuddin, "Menimbang Peran Teknologi dan Guru dalam Pembelajaran di Era COVID-19," *Menimbang Peran Teknologi dan Guru dalam Pembelajaran di Era COVID-19*, 2020.
- [2] C. K. Putri, D. Meidelfi, and A. Erianda, "Perancangan Aplikasi Pemesanan Travel Antar Kota Di Sumatera Barat Berbasis Android," *Journal of Applied Computer Science and Technology*, vol. 1, no. 1, pp. 32–37, 2020.
- [3] M. G. Gunagama, Y. R. Naurah, and A. E. P. Prabono, "Pariwisata pascapandemi: Pelajaran penting dan prospek pengembangan," *LOSARI: Jurnal Arsitektur Kota dan Pemukiman*, pp. 56–68, 2020.
- [4] I. Fakhruddin, N. Rismawati, and R. Sriyanti, "Perancangan Sistem Aplikasi Penjualan Paket Wisata Pulau Seribu Berbasis Java Dengan Metode Scrum (Studi Kasus PT. Abarter Global Indonesia)," *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 5, no. 1, 2022.

- [5] F. A. Hendraswara, T. Informatika, F. T. Informasi, U. Kristen, and S. Wacana, "Perancangan Pemesanan Paket Wisata dengan Pembayaran Online Menggunakan Payment Gateway pada aplikasi Android (Studi Kasus: CV Ambarawa Raya Travel Agent)," *no. April*, 2016.
- [6] N. AZIZAH, "RANCANG BANGUN APLIKASI PANDUAN WISATA MAGELANG DI ERA MILLENNIAL BERBASIS ANDROID MOBILE APP".
- [7] T. W. Harjanti and L. Lipindi, "Rancang Bangun Sistem Rekomendasi Blitrotravel pada Fitur Custom Trip untuk Aplikasi Travel Vaithme Menggunakan Algoritma Weighted Product Berbasis Android," in *Seminar Nasional Teknologi*, 2019, p. 704.
- [8] M. Rizki and L. Slamet, "Rancang Bangun Aplikasi Penjualan Tiket Travel di Lima Puluah Kota (Mudiak) Berbasis Android," *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, vol. 8, no. 1, pp. 82–89, 2020.
- [9] E. Lumba, "PERTUKARAN DATA PADA APLIKASI ANDROID MENGGUNAKAN JAVA SCRIPT OBJECT NOTATION (JSON) DAN REST API DENGAN RETROFIT 2," *PROSIDING SNAST*, pp. 118–127, 2021.
- [10] S. Syafril, Z. Asril, E. Engkizar, A. Zafirah, F. A. Agusti, and I. Sugiharta, "Designing prototype model of virtual geometry in mathematics learning using augmented reality," in *Journal of Physics: Conference Series*, IOP Publishing, 2021, p. 012035.
- [11] A. U. Zailani, A. Perdananto, and M. Ardiansyah, "Penggunaan Model Prototype dalam Membuat Library System di SMPIT AL Mustopa," *SMARTICS Journal*, vol. 6, no. 2, pp. 89–96, 2020.
- [12] S. Al-Fedaghi, "Validation: Conceptual versus Activity Diagram Approaches," *arXiv preprint arXiv:2106.16160*, 2021.
- [13] H. Badioze Zaman *et al.*, Eds., *Advances in Visual Informatics*, vol. 13051. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-90235-3.
- [14] A. M. Nur Fauzi, A. Triayudi, and I. D. Sholihati, "MENGUKUR TINGKAT KEPUASAN PENGGUNA APLIKASI KEARSIPAN MENGGUNAKAN SYSTEM USABILITY SCALE DAN PIECES FRAMEWORK," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 1, pp. 231–239, Feb. 2022, doi: 10.29100/jupi.v7i1.2452.
- [15] Z. Sharfina and H. B. Santoso, "An Indonesian adaptation of the System Usability Scale (SUS)," in *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, IEEE, Oct. 2016, pp. 145–148. doi: 10.1109/ICACSIS.2016.7872776.
- [16] M. S. Hartawan, "PENERAPAN USER CENTERED DESIGN (UCD) PADA WIREFRAME DESAIN USER INTERFACE DAN USER EXPERIENCE APLIKASI SINOPSIS FILM," *JEIS: JURNAL ELEKTRO DAN INFORMATIKA SWADHARMA*, vol. 2, no. 1, pp. 43–47, Jan. 2022, doi: 10.56486/jeis.vol2no1.161.
- [17] I. Ripai, "IMPLEMENTASI DAN USER ACCEPTANCE TEST (UAT) APLIKASI INTEGRATED LIBRARY SYSTEM (INLIS Lite) DI MTs NEGERI 7 KUNINGAN," *ICT Learning*, vol. 7, no. 1, 2023.
- [18] R. Supriatna, "IMPLEMENTASI DAN USER ACCEPTANCE TEST (UAT) TERHADAP APLIKASI E-LEARNING PADA MADRASAH ALIYAH NEGERI (MAN) 3 KOTA BANDA ACEH," UIN Ar-Raniry Banda Aceh, Banda Aceh, 2019.