

OAJIS

Open Access
Journal of
Information
Systems

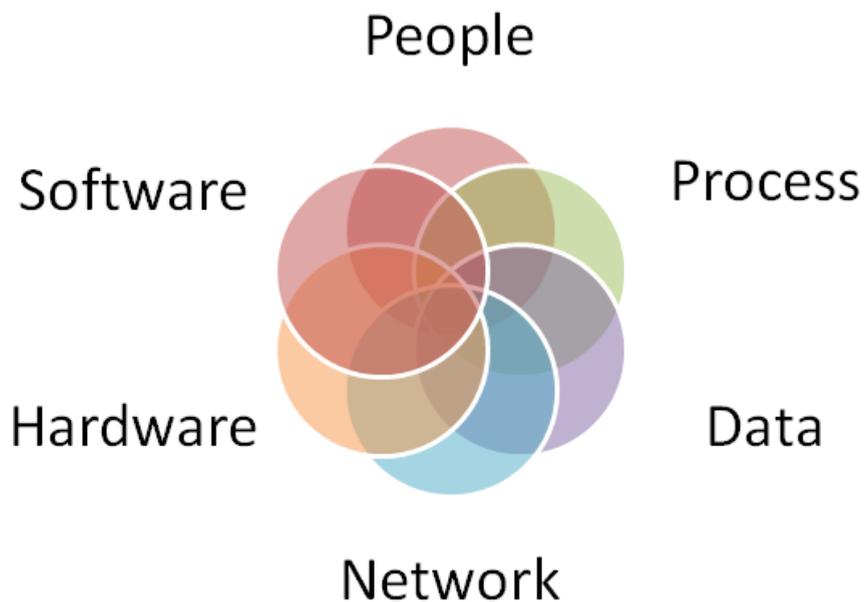
is.its.ac.id/pubs/oajis/

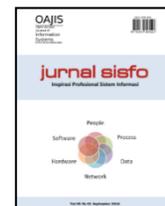
ISSN 1979-3979



jurnal sisfo

Inspirasi Profesional Sistem Informasi





Pimpinan Redaksi

Eko Wahyu Tyas Darmaningrat

Dewan Redaksi

Amna Shifia Nisafani

Arif Wibisono

Faizal Mahananto

Tata Pelaksana Usaha

Achmad Syaiful Susanto

Ricky Asrul Sani

Rini Ekowati

Sekretariat

Jurusan Sistem Informasi – Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember (ITS) – Surabaya

Telp. 031-5999944 Fax. 031-5964965

Email: editor@jurnalsisfo.org

Website: <http://jurnalsisfo.org>

Jurnal SISFO juga dipublikasikan di *Open Access Journal of Information Systems* (OAJIS)

Website: <http://is.its.ac.id/pubs/oajis/index.php>



Mitra Bestari

Aditya Rachmadi, S.ST, M.TI (Universitas Brawijaya)

Ahmad Mukhlason, S.Kom, M.Sc, Ph.D (Institut Teknologi Sepuluh Nopember)

Amalia Utamima, S.Kom, MBA (Institut Teknologi Sepuluh Nopember)

Anisah Herdiyanti, S.Kom, M.Sc, ITILF (Institut Teknologi Sepuluh Nopember)

Ari Widyanti, S.T, M.T, Ph.D (Institut Teknologi Bandung)

Dany Primanita Kartikasari, S.T, M.Kom (Universitas Brawijaya)

Dewi Yanti Liliana, S.Kom, M.Kom (Politeknik Negeri Jakarta)

Erma Suryani, S.T, M.T, Ph.D (Institut Teknologi Sepuluh Nopember)

Faizal Johan Atletiko, S.Kom, M.T (Institut Teknologi Sepuluh Nopember)

Feby Artwodini Muqtadiroh, S.Kom, M.T (Institut Teknologi Sepuluh Nopember)

Hatma Suryotrisongko, S.Kom, M.Eng (Institut Teknologi Sepuluh Nopember)

Irmasari Hafidz, S.Kom, M.Sc (Institut Teknologi Sepuluh Nopember)

Mahendrawathi ER., S.T, M.Sc, Ph.D (Institut Teknologi Sepuluh Nopember)

Mudjahidin, S.T, M.T (Institut Teknologi Sepuluh Nopember)

Nur Aini R., S.Kom, M.Sc.Eng, Ph.D (Institut Teknologi Sepuluh Nopember)

Radityo Prasetianto W., S.Kom, M.Kom (Institut Teknologi Sepuluh Nopember)

Rahadian Bisma, S.Kom, M.Kom, ITILF (Universitas Negeri Surabaya)

OAJIS

Open Access
Journal of
Information
Systems
is.its.ac.id/pubs/oajis/

jurnal sisfo

Jurnal Sisfo Vol. 6 No. 1 (2017) i–iii



Mitra Bestari

Raras Tyasnurita, S.Kom, MBA (Institut Teknologi Sepuluh Nopember)

Renny Pradina Kusumawardani, S.T, M.T (Institut Teknologi Sepuluh Nopember)

Retno Aulia Vinarti, S.Kom, M.Kom (Institut Teknologi Sepuluh Nopember)

Sholiq, S.T, M.Kom, M.SA (Institut Teknologi Sepuluh Nopember)

Wiwik Anggraeni, S.Si, M.Kom (Institut Teknologi Sepuluh Nopember)



Daftar Isi

Pembuatan Perangkat Lunak Berbasis Lokasi untuk Berbagi Kendaraan

Arif Wibisono, Amril Azhar..... 265

Pemetaan Proses Bisnis dengan Pendekatan Klasifikasi Proses CIMOSA: Studi Kasus Perusahaan Pengelola Kawasan Industri

Effi Latiffianti, Stefanus Eko Wiratno, Dewanti Anggrahini, Muhammad Saiful Hakim..... 283

Sistem Penginderaan Berbasis UAV untuk Membantu Operasi Pencarian dan Penyelamatan Korban Kecelakaan di Wilayah Pegunungan

Ketut Bayu Yogha, Rajalida Lipikorn..... 293

Analisis dan Perancangan Sistem Informasi Data Siswa Pada Sekolah Menengah Kejuruan (SMK) PGRI 8 Medan dengan *Zachman Framework*

Safrian Aswati, Ada Udi Firmansyah, William Ramdhan, Suhendra..... 309

Desain dan Evaluasi Prototipe Jaringan Sensor Nirkabel untuk Monitoring Lahan Persawahan di Kabupaten Gowa

Mohammad Fajar, Agus Halid, Syaiful Rahman 319

Evaluasi Kebergunaan (*Usability*) pada Aplikasi Daftar Online Rumah Sakit Umum Daerah Gambiran Kediri

Fithrotu Khoirina, Anisah Herdiyanti, Tony Dwi Susanto..... 331

Sistem Pakar untuk Menentukan Penyakit Hernia dengan Menggunakan Metode *Dempster-Shafer*

Tumingan, Yessy Yanitasari, Dedih..... 347

Pengelompokan Peminatan Program Menggunakan *K-Means Clustering* Berdasarkan Asal Sekolah

C. Purnama Yanti..... 383



Pembuatan Perangkat Lunak Berbasis Lokasi untuk Berbagi Kendaraan

Arif Wibisono^{*}, Amril Azhar

^aDepartemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember

Abstract

The level of congestion in major cities in Indonesia is quite high and is very challenging to overcome. This causes various negative impacts such as increased air pollution, increased travel time, wasted fuel, and others. Car sharing is an effective solution to solve traffic congestion in urban areas. The reason is that it can reduce the use of private vehicles by increasing vehicle occupancy. Media liaison between the provider and the passenger elevator is required to share the vehicle can be implemented for all citizens. In this study, we develop a prototype of a web application to share vehicle (ridesharing) as a medium to connect between the giver and the passenger. We utilize a traditional software development lifecycle (SDLC) named Waterfall model. The SDLC produces a software application prototype. The prototype developed using Google Maps to assist in determining the appropriate travel and suitable routes. The salient feature of this prototype is that it can propose a suitable travel route is identical, partial, and inclusive and seek a partner to share a ride vehicle to do the giver and the passenger.

Keywords: Ridesharing Application, Carpooling, Ridesharing, Google Maps

Abstrak

Tingkat kemacetan di kota-kota besar di Indonesia cukup tinggi dan sangat menantang untuk mengatasi. Hal ini menyebabkan berbagai dampak negatif seperti polusi meningkat udara, peningkatan waktu tempuh, bahan bakar terbuang, dan lain-lain. Mobil berbagi adalah solusi yang efektif untuk mengatasi kemacetan lalu lintas di perkotaan karena dapat mengurangi penggunaan kendaraan pribadi dengan meningkatkan kendaraan hunian. Media penghubung antara penyedia dan lift penumpang diperlukan untuk berbagi kendaraan dapat diimplementasikan untuk semua orang. Studi ini mengembangkan prototipe dari sebuah aplikasi web untuk berbagi kendaraan sebagai media untuk menghubungkan antara pemberi dan penumpang. Metode pengembangan yang digunakan adalah Model Waterfall. Penelitian ini menghasilkan prototipe aplikasi perangkat lunak. Prototipe yang dikembangkan menggunakan Google Maps untuk membantu dalam menentukan perjalanan dan cocok rute yang tepat. Fitur yang menonjol dari prototipe ini adalah bahwa hal itu dapat mengusulkan rute perjalanan yang cocok identik, parsial, dan inklusif dan mencari mitra untuk berbagi tumpangan kendaraan untuk melakukan pemberi dan penumpang.

Kata kunci: Ridesharing Application, Carpooling, Ridesharing, Google Maps

© 2017 Jurnal SISFO.

Histori Artikel : Disubmit 2 Januari 2017; Diterima 31 Januari 2017; Tersedia online 12 April 2017

^{*}Corresponding Author
Email wibisono@is.its.ac.id (Arif Wibisono)

1. Pendahuluan

Tingkat kemacetan di kota besar seperti DKI Jakarta, Bogor, Bandung dan Surabaya sudah cukup tinggi. Satuan Lalu Lintas Polrestabes Surabaya memprediksi terjadinya peningkatan kemacetan sebesar 30 % pada tahun 2013 jika dibandingkan dengan tahun 2012 [1]. DKI Jakarta, Bogor dan Bandung saat ini memiliki perbandingan antara jumlah kendaraan dengan daya tampung jalan (VCR) 0,85 – 0,86 dan laju kendaraan berkisar sepuluh kilometer hingga dua puluh kilometer [2]. VCR 0,85 menandakan kapasitas jalan untuk sepuluh kendaraan digunakan oleh delapan setengah kendaraan. Kendaraan yang melintas di Surabaya sekitar 80% merupakan kendaraan pribadi [3], hal ini juga berlaku pada kota-kota besar lainnya. Pada tahun 2009 jumlah pengguna kendaraan pribadi sebesar tujuh juta untuk mobil penumpang dan lima puluh dua juta untuk sepeda motor. Jumlah kendaraan bermotor terus meningkat tiap tahunnya. Pada tahun 2013 jumlah pengguna kendaraan pribadi meningkat menjadi sebelas juta untuk mobil penumpang dan delapan puluh empat juta untuk sepeda motor. Banyaknya penggunaan kendaraan pribadi khususnya mobil menyebabkan lalu lintas semakin padat. Penggunaan mobil pribadi biasanya juga hanya di isi oleh satu atau dua orang. Apabila setiap dua pengguna mobil pribadi dapat disatukan dalam satu mobil saja, hal itu akan dapat mengurangi jumlah mobil yang melintas sebesar 50 %.

Peningkatan jumlah kendaraan juga berdampak pada meningkatnya polusi di perkotaan. Polusi yang banyak dihasilkan dari kendaraan adalah gas karbon monooksida. Data dari Ecoton pada tahun 2007 menunjukkan beberapa titik di kota Surabaya memiliki tingkat polusi yang tinggi, dan polusi dengan jenis karbon monooksida sebesar 96 % dihasilkan oleh moda transportasi. Berdasarkan data dari *World Health Organization* (WHO) kota Surabaya, Bandung, dan DKI Jakarta memiliki tingkat polusi diatas batas maksimal PM 10 yang ditetapkan WHO yaitu 20 mikrogram/ m³. Kota Surabaya memiliki tingkat polusi sekitar 69 mikrogram/m³ sedangkan Bandung dan Jakarta sebesar 43 mikrogram/m³.

Penggunaan kendaraan pribadi yang ada di perkotaan berpusat pada waktu-waktu tertentu, yaitu pada pagi, siang dan sore hari tepatnya pada saat warga kota berangkat kerja, istirahat kerja dan pada saat pulang kerja. Tingginya penggunaan kendaraan pribadi yang menyebabkan tingginya polusi dan kemacetan, membuat pemerintah menggalakkan penggunaan angkutan massal sebagai alternatif untuk mengurangi penggunaan kendaraan pribadi. Beberapa angkutan massal yang tersedia sebagai alternatif adalah bus dan kereta. Bus dan kereta memiliki kelebihan pada sisi jumlah penumpang yang bisa diangkut dan harga yang murah. Bus dapat mengangkut hingga puluhan orang sedangkan kereta bisa mencapai ratusan orang, tetapi keduanya memiliki kelemahan. Bus dan kereta dianggap kurang nyaman dan aman jika dibandingkan dengan mobil pribadi selain itu rute yang dilalui bus dan kereta juga sudah pasti atau tidak fleksibel. Alternatif lainnya untuk angkutan massal adalah angkutan kota (lyn, bemo, dan bison) dan mobil antar jemput, keduanya merupakan angkutan massal yang hanya dapat mengangkut sekitar lima sampai sepuluh orang. Angkutan kota memiliki kelemahan dari sisi kenyamanan dan keamanan, serta memiliki rute perjalanan yang tetap seperti pada bus dan kereta tetapi angkutan kota memiliki kelebihan yaitu harga yang murah. Berbeda dengan angkutan kota mobil antar jemput memiliki tingkat kenyamanan dan keamanan yang hampir sama dengan mobil pribadi serta memiliki rute yang fleksibel, tetapi biaya sewa yang dikenakan cukup mahal. Alternatif lain yang cukup efektif adalah penggunaan kendaraan pribadi secara bersama-sama atau bisa disebut juga berbagi kendaraan (*ridesharing*). Dari beberapa pendapat peneliti dalam bidang transportasi [4] [5] [6] [7] [8], berbagi kendaraan (*ridesharing*) dapat mengurangi jumlah polusi, kemacetan dan konsumsi bahan bakar minyak (BBM). Selain itu berbagi kendaraan (*ridesharing*) memiliki tingkat kenyamanan dan keamanan seperti mobil pribadi, dan biaya yang dikeluarkan lebih murah jika dibandingkan dengan mobil antar jemput. Di Indonesia konsep berbagi kendaraan sudah ada sejak lama, dan pada tahun 2005 Rudyanto Linggar mempublikasikan website *nebeng.com* sebagai media untuk mempertemukan calon pemberi tumpangan dengan calon penumpang. Website *nebeng.com* sudah diakui oleh google sebagai aplikasi inovatif dalam bidang transportasi, akan tetapi website ini memiliki kelemahan pada sistem pencarian pemberi tumpangan oleh penumpang. Sistem pencariannya dilakukan berdasarkan data lokasi kota dan wilayah (kecamatan, kelurahan atau perumahan), hal ini membuat hasil pencarian tidak efektif karena calon penumpang harus membaca satu persatu rute perjalanan yang sesuai dengan kebutuhannya. Untuk

menyelesaikan masalah ini kami mengusulkan layanan berbagi kendaraan (*ridesharing*) dengan memanfaatkan Google Maps dalam pencarian rute perjalanan.

Penelitian ini mengusulkan sebuah prototipe aplikasi perangkat lunak. Prototipe ini dikembangkan dengan menggunakan Siklus Pengembangan Perangkat Lunak bertipe Waterfall. Prototipe yang dikembangkan menggunakan Google Maps untuk membantu dalam menentukan perjalanan dan cocok rute yang tepat. Fitur yang menonjol dari prototipe ini adalah bahwa hal itu dapat mengusulkan rute perjalanan yang cocok identik, parsial, dan inklusif dan mencari mitra untuk berbagi tumpangan kendaraan untuk melakukan pemberi dan penumpang. Prototipe ini diharapkan dapat memudahkan pemberi tumpangan potensial dan penumpang potensial dalam menemukan satu sama yang lain untuk berbagi kendaraan

2. Tinjauan Pustaka/ Penelitian Sebelumnya

2.1 *Ridesharing*

Ridesharing adalah konsep berbagi kendaraan untuk perjalanan sehingga kendaraan digunakan lebih dari satu orang. *Ridesharing* bertujuan untuk meningkatkan okupansi kendaraan yang akan berdampak pada pengurangan penggunaan kendaraan pribadi di jalan [9]. Saat ini proses *ridesharing* sudah difasilitasi oleh penyedia layanan *ridesharing*. Ada dua tipe penyedia layanan *ridesharing* [10]: 1) *Service Operators* yaitu mereka yang menjalankan layanan *ridesharing* menggunakan kendaraan dan pengemudi mereka sendiri 2) *Matching Agencies* yaitu mereka yang memfasilitasi layanan *ridesharing* dengan melakukan pencocokan rute antara pemberi tumpangan dan penumpang. Sementara itu pola rute perjalanan *ridesharing* bisa di bedakan menjadi empat : 1) *Identical Ridesharing* yaitu Rute perjalanan antara pemberi tumpangan dan penumpang sama persis. Pemberi tumpangan memiliki titik asal keberangkatan dan tujuan yang sama dengan penumpang 2) *Inclusive Ridesharing* yaitu rute perjalanan penumpang berada didalam rute perjalanan pemberi tumpangan. Rute perjalanan dari pemberi tumpangan melewati titik keberangkatan dan tujuan dari penumpang 3) *Partial Ridesharing* yaitu rute perjalanan penumpang hanya sebagian yang sama dengan pemberi tumpangan. Penumpang memiliki titik asal keberangkatan dan penjemputan yang berbeda serta titik penurunan dan tujuan yang berbeda pula. Sehingga perjalanan pemberi tumpangan hanya mencakup sebagian dari perjalanan penumpang dan 4) *Detour Ridesharing* yaitu rute perjalanan penumpang dan pemberi tumpangan tidak sama, tetapi pemberi tumpangan menyesuaikan rute dengan penumpang sehingga terjadi perubahan rute perjalanan dari pemberi tumpangan.

2.2 Penelitian Sebelumnya

Dari beberapa penelitian yang ada saat ini layanan berbagi kendaraan dapat dibagi menjadi dua tipe proses bisnis, pertama *ridesharing* konvensional seperti pada “*nebeng.com*”, dan yang kedua *dynamic ridesharing* atau *real-time ridesharing* [9] [11] [12]. Saat ini penelitian banyak yang memfokuskan pada pengembangan *dynamic ridesharing* atau *real-time ridesharing*, tetapi konsep *real-time ridesharing* masih susah di terapkan di Indonesia karena faktor keamanannya yang masih belum terjamin [12]. *Real-time ridesharing* bertujuan untuk menyediakan sarana transportasi satu arah, dan biasanya digunakan oleh orang yang ingin liburan atau pergi ke suatu tempat. *Real-time ridesharing* tidak akan berdampak banyak pada penurunan kemacetan di jam-jam padat, yaitu pada jam berangkat dan pulang kerja.

Pada penelitian lainnya [10] [13] [14] banyak peneliti yang membuat model dan simulasi tetapi belum di uji penerapannya dalam dunia nyata. Dalam penelitian lainnya [15] dikembangkan model *push service* untuk *ridesharing* namun penelitian ini memiliki kelemahan pada pencocokan rute antara pemberi tumpangan dan penumpang. Pencocokan rute hanya dilakukan berdasarkan titik-titik lokasi padahal pada kenyataannya rute berbentuk jalur dengan kemungkinan penumpang yang berada didekat jalur untuk ikut menumpang. Penelitian *Cho et al* [16] dan *Dessouky et al* [10] meneliti mengenai kemungkinan titik penjemputan yang fleksibel tersebut, hasilnya adalah sebuah model. Model yang dikembangkan oleh *Cho et al* dan *Dessouky*

et al dapat dijadikan sebagai acuan dalam pembuatan proses bisnis aplikasi berbagi kendaraan (*ridesharing*) dengan mengambil beberapa aspek seperti pencocokan rute (*identical*, *partial* dan *inclusive*), dan bentuk layanan berbagi kendaraan (*ridesharing*).

Aplikasi web yang ada saat ini (seperti Nebeng.com dan Blablacar) juga masih banyak yang belum memanfaatkan open GIS seperti Google Maps, Bing Maps, dan OpenLayers. Padahal dengan adanya GIS penumpang maupun pemberi tumpangan bisa menentukan lokasi keberangkatan dan lokasi tujuan dengan lebih tepat. *Lin et al* mengajukan sistem *real-time ridesharing* yang memanfaatkan cloud service dan open GIS untuk pencocokan rute antar penumpang dan pemberi tumpangan. *Lin et al* [9] mengajukan algoritma yang dapat memproses pencocokan rute yang lebih cepat dari algoritma biasa. Algoritma yang diajukan memang lebih cocok digunakan untuk *real-time ridesharing* karena lebih mengedepankan kecepatan dari pada kelengkapan. Algoritma yang diajukan *Lin et al* mengesampingkan beberapa data untuk tidak di cek agar proses *query* lebih cepat. Algoritma ini dapat digunakan dalam pengembangan aplikasi berbagi kendaraan (*ridesharing*) untuk mengantisipasi data rute perjalanan yang cukup besar.

3. Metodologi

Berikut ini adalah tahapan dalam pengembangan prototipe aplikasi sebagaimana tergambar pada Gambar 1.



Gambar 1 Metodologi Penelitian

3.1 Studi Literatur

Studi literatur yang dilakukan adalah pembelajaran dan pemahaman literatur mengenai *ridesharing*, atau *carpooling*. Beberapa yang akan dipelajari adalah model-model proses bisnis berbagi kendaraan (*ridesharing*) dari *Dessouky et al* dan *Cho et al*, algoritma untuk pencocokan rute antar pemberi tumpangan dan penumpang, serta pemahaman mengenai penggunaan *Google Maps API V3*.

3.2 Identifikasi Kebutuhan Perangkat Lunak

Pada tahap ini dilakukan analisa kebutuhan perangkat lunak aplikasi berbagi kendaraan (*ridesharing*). Kebutuhan perangkat lunak didapatkan dari pemahaman studi literatur dan perbandingan dengan aplikasi sejenis. Dari kebutuhan perangkat lunak yang ada lalu dipetakan menjadi kebutuhan fungsional dan kebutuhan non-fungsional.

3.3 Perancangan Aplikasi

Pada tahap ini dilakukan desain aplikasi berbagi kendaraan (*ridesharing*). Data yang didapat dari analisa kebutuhan diubah kedalam bentuk *use cases*. Pada tahap ini juga dilakukan desain struktur data dan arsitektur perangkat lunak. Pada penelitian ini database yang digunakan adalah MySQL. Bahasa pemrograman yang digunakan dalam pembuatan aplikasi adalah *PHP hypertext Processor* (PHP). *Geographical Information System* (GIS) yang di gunakan adalah *Google Maps*.

3.4 Pengkodean Aplikasi

Pada tahap ini dilakukan pengkodean yang didasarkan pada desain aplikasi. Pengkodean *web* dilakukan dengan menggunakan bahasa pemrograman PHP dan framework *Codeigniter*, sedangkan untuk menghubungkan aplikasi web dengan *Google Maps API V3* digunakan bahasa *javascript*.

3.5 Pengujian Aplikasi

Pada tahap ini dilakukan pengujian aplikasi berbagi kendaraan. Pengujian dilakukan untuk menguji fungsional aplikasi. Pengujian dilakukan dengan menggunakan *Requirements Traceability Matrix* (RTM). Apabila terdapat fungsi aplikasi yang kurang atau tidak bekerja sesuai kebutuhan maka harus dilakukan pengkodean untuk menambah atau memperbaiki fungsi aplikasi.

4. Hasil dan Pembahasan

Berikut ini adalah hasil dan pembahasan dari penelitian ini.

4.1 Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak terdiri dari dua hal, yaitu kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional diperoleh melalui studi literatur dan *benchmarking* aplikasi sejenis. Berdasarkan hasil identifikasi kebutuhan aplikasi *ridesharing* didapatkan daftar kebutuhan fungsional sebagai seperti pada Tabel 1.

Tabel 1. Kebutuhan Fungsional

ID	Deskripsi Kebutuhan Fungsional
FR01	Sistem dapat mempertemukan antara calon penumpang dan pemberi tumpangan

ID	Deskripsi Kebutuhan Fungsional
FR02	Sistem dapat memberikan rute terbaik untuk bertemu
FR03	Sistem dapat memastikan bahwa tidak terjadi <i>overcapacity</i> pada mobil
FR04	Sistem dapat mengelola rating bagi penumpang dan pemberi tumpangan
FR05	Sistem dapat mengelola jadwal penjemputan
FR06	Sistem dapat mengelola pengguna

Berdasarkan dari hasil identifikasi kebutuhan fungsional dapat dipetakan aktor-aktor yang terlibat dalam aplikasi *ridesharing* yang dikembangkan. Rincian informasi mengenai aktor-aktor yang terlibat akan dijabarkan pada Tabel 2.

Tabel 2. Daftar Aktor

Aktor	Deskripsi
Pengguna	Orang yang terdaftar dalam aplikasi dan nantinya menggunakan aplikasi sesuai dengan peran yang dipilih
Pemberi Tumpangan	Pengguna yang berperan sebagai penyedia tumpangan untuk penumpang
Penumpang	Pengguna yang berperan sebagai penumpang dari pemberi tumpangan
Admin	Orang yang melakukan verifikasi dan validasi pengguna, serta mengelola aplikasi

Adapun alur penggunaan sistem berbagi kendaraan adalah sebagai berikut.

- 1) Pemberi tumpangan membuat rute perjalanan.
- 2) Pemberi tumpangan merencanakan *ridesharing*
- 3) Penumpang membuat rute perjalanan
- 4) Penumpang mencari *ridesharing* berdasarkan peta perjalanan
- 5) Setelah hasil pencarian di temukan penumpang menumpang pada salah satu *ridesharing*
- 6) Pemberi tumpangan mengkonfirmasi permintaan tumpangan (menyetujui).
- 7) Penumpang menerima pesan konfirmasi.
- 8) Pemberi tumpangan dan penumpang melakukan *ridesharing* sesuai rencana
- 9) Pemberi tumpangan memberi review dan rating ke penumpang
- 10) Penumpang memberi review dan rating ke pemberi tumpangan

Selain kebutuhan fungsional, sistem *ridesharing* ini memiliki kebutuhan non fungsional sebagaimana terlihat dalam Tabel 3. Kebutuhan non fungsional ini diperoleh dengan melakukan komparasi subyektif dengan aplikasi sejenis di luar negeri seperti Lyft, Uber, Getaround, Sidecar, Carma Carpooling, Rideshare, dan Zimride. Secara umum ada empat kebutuhan non fungsional sebagaimana bisa dilihat di Tabel 3.

Tabel 3. Kebutuhan Non Fungsional

ID	Deskripsi Kebutuhan Non Fungsional
NFR01	Sistem mampu berjalan di browser umum masyarakat Indonesia
NFR02	Sistem menggunakan bahasa Indonesia dengan tampilan yang dapat dimengerti pengguna umum
NFR03	Sistem berjalan dengan sistem open source
NFR04	Sistem dapat memastikan bahwa pengguna memiliki identitas yang sama dengan yang diklaimkan

4.2 Perancangan Aplikasi

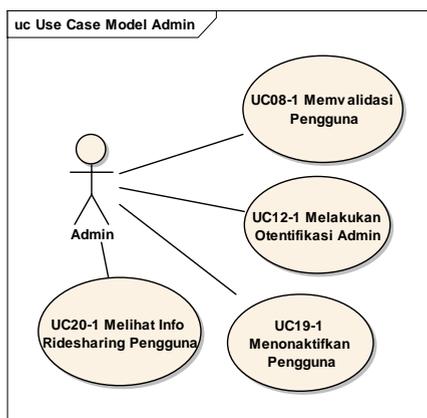
4.2.1 Use Case

Use case dikembangkan dengan menurunkan *functional requirement* yang telah didefinisikan sebelumnya. Ada 20 use case yang memetakan diri ke 6 *functional requirement* (lihat Tabel 4).

Tabel 4. Mapping Functional Requirement terhadap use case

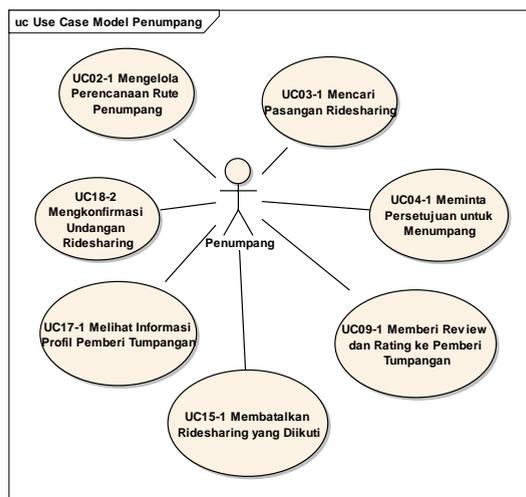
Functional Requirement ID	Use Case ID	Use Case
FR01, FR02	UC01-1	Mengelola Pembuatan Perencanaan Rute Pemberi Tumpangan
FR02	UC01-2	Mengelola Rute Perjalanan
FR01, FR02	UC02-1	Mengelola Perencanaan Rute Penumpang
FR01	UC03-1	Mencari Pasangan <i>Ridesharing</i>
FR01	UC04-1	Meminta Persetujuan Untuk Menumpang
FR01	UC05-1	Mengkonfirmasi Permintaan Tumpangan
FR02	UC06-1	Memperkirakan Titik dan Waktu Penjemputan
FR01	UC07-1	Meregistrasi Pengguna
FR04	UC08-1	Memvalidasi Pengguna
FR04	UC09-1	Memberi <i>Review</i> dan <i>Rating</i> ke Pemberi Tumpangan
FR04	UC10-1	Memberi <i>Review</i> dan <i>Rating</i> ke Penumpang
FR06	UC11-1	Melakukan Otentifikasi Pengguna
FR06	UC12-1	Melakukan Otentifikasi Admin
FR01	UC13-1	Melihat Detail Informasi <i>Ridesharing</i>
FR01,FR02,FR05	UC14-1	Membatalkan Perencanaan <i>Ridesharing</i>
FR01,FR02,FR05	UC15-1	Membatalkan <i>Ridesharing</i> yang Diikuti
FR01,FR02,FR05	UC16-1	Melihat Informasi Profil Penumpang

Functional Requirement ID	Use Case ID	Use Case
FR01,FR02,FR05	UC17-1	Melihat Informasi Profil Pemberi Tumpangan
FR06	UC18-1	Mengelola Akun
FR06	UC19-1	Menonaktifkan Pengguna
FR04	UC20-1	Melihat Informasi <i>Ridesharing</i> yang Dilakukan Pengguna



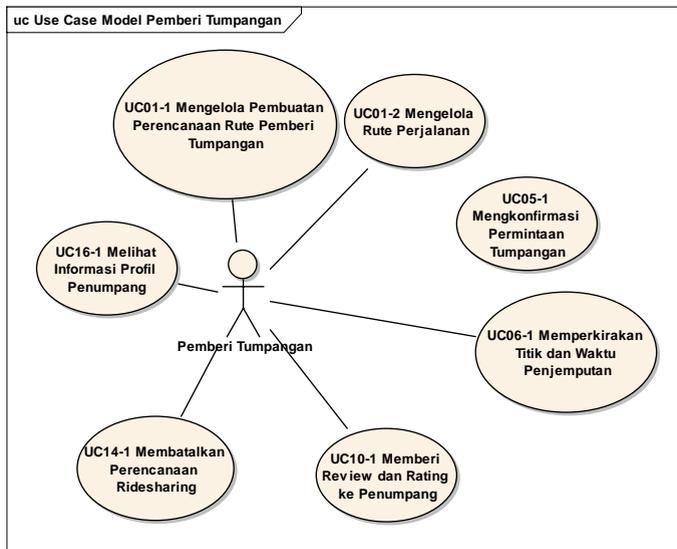
Gambar 2 Diagram *use case* Admin

Lebih jauh, Admin akan bertanggung jawab terhadap empat *use case* yaitu : memvalidasi pengguna, melakukan otentifikasi admin, menonaktifkan pengguna dan melihat info ridesharing pengguna (lihat Gambar 2). Sementara itu, tanggung jawab penumpang ada tujuh yaitu mencari pasangan *ridesharing*, meminta persetujuan untuk menumpang, memberikan *review* dan rating kepada pemberi tumpangan, membatalkan *ridesharing* yang diikuti, melihat informasi profil pemberi tumpangan, mengonfirmasi undangan *ridesharing* dan mengelola rute pribadi (lihat Gambar 3).



Gambar 3 Diagram *use case* Penumpang

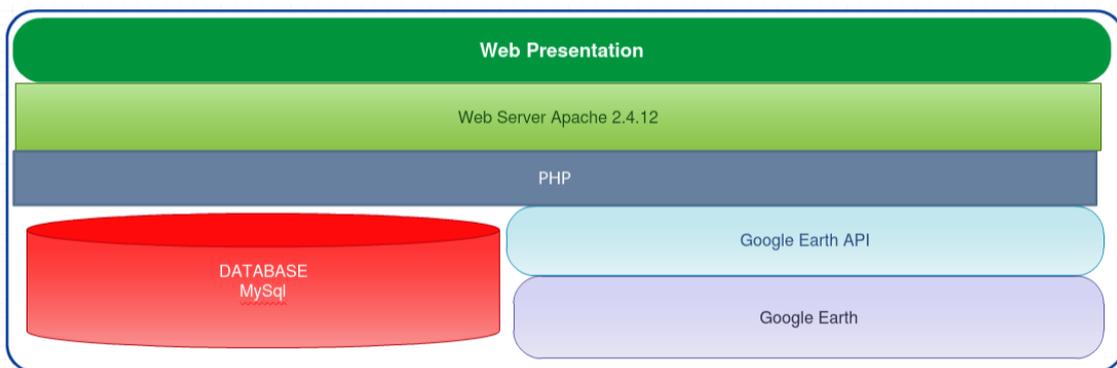
Pemberi tumpangan adalah jenis aktor yang lain daripada admin dan penumpang. Pemberi tumpangan adalah pemilik kendaraan yang mengizinkan seseorang untuk bergabung dalam kelompok *ridesharing*-nya atau tidak. Di sini, pemberi tumpangan memiliki enam fungsi yaitu: mengelola rute perjalanan, mengonfirmasi permintaan tumpangan, memperkirakan titik dan waktu penjemputan, memberi *review* dan rating kepada penumpang, membatalkan perencanaan *ridesharing*, melihat informasi profil penumpang, dan mengelola pembuatan perencanaan rute pemberi tumpangan (lihat Gambar 4).



Gambar 4 Diagram *use case* pemberi tumpangan

4.2.2 Arsitektur Sistem

Aplikasi ini dibuat berbasis web. Pemberi tumpangan atau penumpang mengakses aplikasi melalui koneksi internet dengan *user interface* yang telah disiapkan oleh *web server*. Dalam hal ini, kami menggunakan webserver Apache akan mengolah data dalam *database* MySQL (lihat Gambar 5). Untuk peta, aplikasi ini menggunakan API yang disediakan oleh Google Earth. Di sini peta sudah memiliki algoritma untuk mencari rute terpendek.

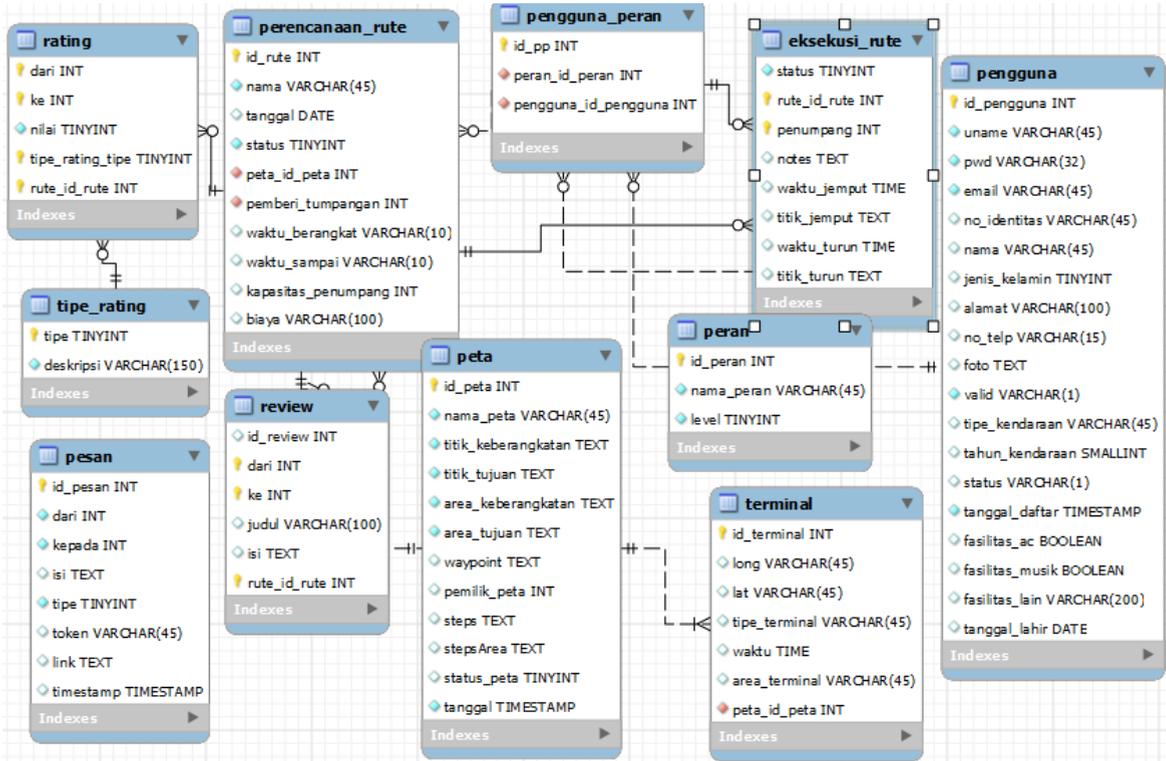


Gambar 5 Arsitektur Sistem

4.2.3 Rancangan *Database*

Selain mengembangkan arsitektur sistem, dalam penelitian ini juga dirancang *Entity Relationship Diagram* (ERD) dengan 11 Tabel sebagaimana yang terlihat pada Gambar 6. Tabel-tabel tersebut antara lain: rating,

tipe rating, pesan, *review*, perencanaan rute, peta, pengguna_peran, eksekusi_rute, terminal, dan pengguna. Tabel peta adalah tabel yang menggunakan data yang didapatkan dari Google Earth. Data yang disimpan adalah point koordinat untuk menjelaskan lokasi secara fisik.



Gambar 6 Entity Relationship Diagram

Tabel lain yang menggunakan Google Earth adalah eksekusi_rute. Tabel ini menyimpan bagaimana rute betul betul dieksekusi oleh pemberi tumpangan. Tentu saja hasil eksekusi bisa jadi sama dengan yang direncanakan dan bisa jadi berbeda. Tabel yang lain adalah tabel yang terkait dengan pengelolaan pengguna misalnya tabel rating, tipe rating, *review*, dan peran_pengguna.

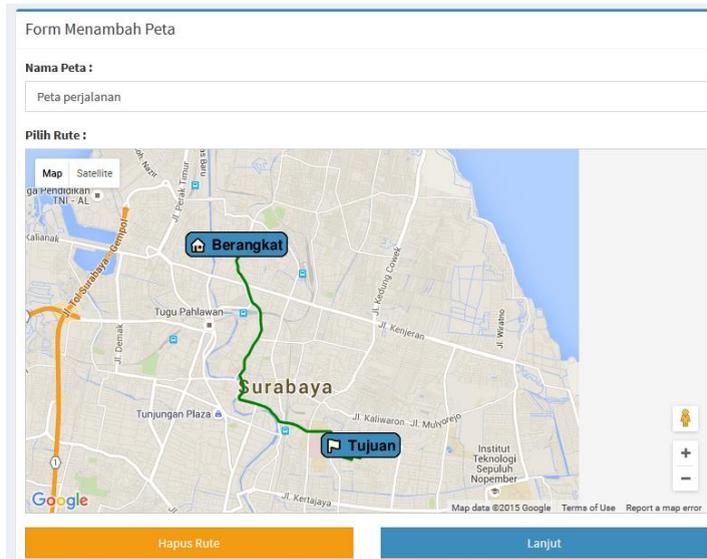
4.3 Pengkodean Aplikasi

Dalam pengembangan perangkat lunak ini, digunakan database MySQL dengan Web Server Apache. Selain itu, dalam pengembangannya juga digunakan bahasa Pemrograman PHP dengan Framework Pemrograman Code Igniter (lihat Tabel 5). Gambar 7, 8, 9 dan 10 menunjukkan tampilan dari perangkat lunak *ridesharing*. Gambar 7 menunjukkan tampilan ketika pengguna mencari rute yang cocok untuk dirinya. Gambar 8 dan 9 secara berturut-turut menunjukkan bagaimana mencari rute *ridesharing* dengan dua inputan yang bisa dimasukkan yaitu: berdasarkan peta atau berdasarkan ID (rute perjalanan).

Tabel 5. Lingkungan Pengembangan Perangkat Lunak

Komponen	Spesifikasi
Web Server	Apache 2.4.12
Database	MySQL 5.6.24

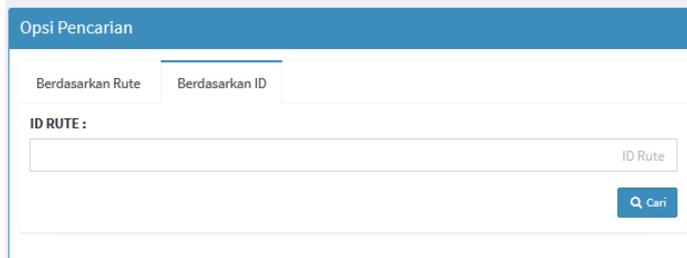
Komponen	Spesifikasi
Bahasa Pemrograman	PHP 5.6.8
Framework Pemrograman	Codeigniter 3.0



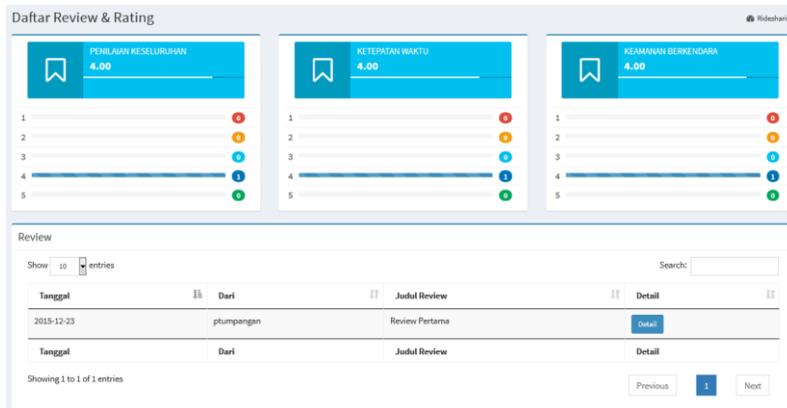
Gambar 7 Halaman Buat Peta Perjalanan



Gambar 8 Halaman Pencarian Ridesharing Berdasarkan Peta



Gambar 9 Halaman Pencarian Ridesharing Berdasarkan ID



Gambar 10 Halaman Melihat Review & Rating

Gambar 10 menunjukkan bagaimana pengguna dapat melihat *review* dan *rating* dari perangkat lunak.

4.4 Pengujian Aplikasi

Pada bagian ini akan dijelaskan mengenai hasil yang didapatkan dari uji coba yang telah dilakukan beserta analisis pembahasan terkait uji coba tersebut.

4.4.1 Uji Coba Fungsional

Teknis uji coba fungsional dilakukan dengan menggunakan *test case*. Hasil uji coba berupa *Requirement Traceability Matrix (RTM)* dapat dilihat pada Tabel 6.

Tabel 6. *Requirement Traceability Matrix*

AC ID	Nama Tes Case	TC ID	Status
AC01-1-01	Membuat perencanaan rute	TC01-1-01	Terpenuhi
AC01-1-02	Mengubah perencanaan rute berstatus draf	TC01-1-02	Terpenuhi
AC01-1-03	Membatalkan perencanaan rute	TC01-1-03	Terpenuhi
AC01-1-04	Membagi Info <i>Ridesharing</i> ke Media Sosial	TC01-1-04	Sebagian
AC01-2-01	Membuat rute perjalanan	TC01-2-01	Terpenuhi
AC01-2-02	Mengubah rute perjalanan berstatus draf	TC01-2-02	Terpenuhi
AC01-2-03	Menghapus rute perjalanan berstatus draf	TC01-2-03	Terpenuhi
AC02-1-01	Merencanakan rute	TC02-1-01	Terpenuhi
AC02-1-02	Mengubah perencanaan rute penumpang	TC02-1-02	Terpenuhi

AC ID	Nama Tes Case	TC ID	Status
AC02-1-03	Menghapus perencanaan rute penumpang	TC02-1-03	Terpenuhi
AC03-1-01	Mencari pasangan <i>ridesharing</i>	TC03-1-01	Terpenuhi
AC04-1-01	Meminta persetujuan untuk menumpang	TC04-1-01	Terpenuhi
AC05-1-01	Mengkonfirmasi permintaan tumpangan	TC05-1-01	Terpenuhi
AC06-1-01	Memperkirakan titik dan waktu penjemputan	TC06-1-01	Terpenuhi
AC07-1-01	Meregistrasi pemberi tumpangan	TC07-1-01	Terpenuhi
AC07-1-02	Meregistrasi penumpang	TC07-1-02	Terpenuhi
AC07-1-03	Meregistrasi pengguna baru	TC07-1-03	Terpenuhi
AC08-1-01	Memvalidasi Pengguna Baru	TC08-1-01	Terpenuhi
AC08-1-02	Membatalkan Validasi Pengguna Lama	TC08-1-02	Terpenuhi
AC09-1-01	Memberi <i>review</i> dan <i>rating</i> ke pemberi tumpangan	TC09-1-01	Terpenuhi
AC10-1-01	Memberi <i>review</i> dan <i>rating</i> ke penumpang	TC10-1-01	Terpenuhi
AC11-1-01	Melakukan login Pengguna	TC11-1-01	Terpenuhi
AC11-1-02	Melakukan Logout Pengguna	TC11-1-02	Terpenuhi
AC12-1-01	Melakukan Login Admin	TC12-1-01	Terpenuhi
AC12-1-02	Melakukan Logout Admin	TC12-1-02	Terpenuhi
AC13-1-01	Melihat detail informasi <i>ridesharing</i>	TC13-1-01	Terpenuhi
AC14-1-01	Membatalkan perencanaan <i>ridesharing</i>	TC14-1-01	Terpenuhi
AC15-1-01	Membatalkan keikutsertaan <i>ridesharing</i>	TC15-1-01	Terpenuhi
AC16-1-01	Melihat rincian informasi penumpang	TC16-1-01	Terpenuhi

AC ID	Nama Tes Case	TC ID	Status
AC17-1-01	Melihat rincian informasi pemberi tumpangan	TC17-1-01	Terpenuhi
AC18-1-01	Mengubah <i>password</i> akun	TC18-1-01	Terpenuhi
AC18-1-02	Me- <i>reset password</i> pengguna	TC18-1-02	Sebagian
AC18-1-03	Mengubah data profil pemberi tumpangan	TC18-1-03	Terpenuhi
AC18-1-04	Mengubah data profil penumpang	TC18-1-04	Terpenuhi
AC19-1-01	Menonaktifkan Pengguna	TC19-1-01	Terpenuhi
AC20-1-01	Melihat <i>Ridesharing</i> Pengguna	TC20-1-01	Terpenuhi

4.4.2 Uji Coba Performa

Uji coba performa dilakukan dengan menggunakan aplikasi Apache JMeter dengan 2 skenario, yaitu 1) melihat *ridesharing* dan 2) mencari *ridesharing*. Uji coba dilakukan empat kali dengan jumlah *request* yang berbeda di setiap uji coba. Jumlah *request* yang digunakan berjumlah 100, 200, 300 dan 400 *request*. Hasil untuk uji coba dengan 100 *request* dapat dilihat pada Tabel 7.

Tabel 7. Hasil Uji Coba dengan 100 request

Test	Samples	Avg	min	max	Error
Melihat Detail Ridesharing	100	3132	85	5413	0,00%
Mencari Ridesharing	100	59662	6917	70992	0,00%

Dari hasil uji coba dengan 100 *request* dapat dilihat *server* masih bisa menangani tanpa adanya error. Hasil uji coba dengan 200 *request* dapat dilihat pada Tabel 8.

Tabel 8. Hasil Uji Coba dengan 200 Concurrent User

Test	Samples	Avg	min	max	Error
Melihat Detail Ridesharing	200	2915	85	5413	0,00%
Mencari Ridesharing	200	90268	3237	117683	0,00%

Dari hasil uji coba dengan 200 *request* dapat dilihat *server* masih bisa menangani tanpa adanya error. Hasil uji coba dengan 300 *request* dapat dilihat pada Tabel 9.

Tabel 9. Hasil Uji Coba dengan 300 *Concurrent User*

Test	Samples	Avg	min	max	Error
Melihat Detail Ridesharing	300	3123	342	3937	0,00%
Mencari Ridesharing	300	100355	4966	131738	0,00%

Dari hasil uji coba dengan 300 *request* dapat dilihat *server* masih bisa menangani tanpa adanya error. Hasil uji coba dengan 400 *request* dapat dilihat pada Tabel 10.

Tabel 10. Hasil Uji Coba dengan 400 *Concurrent User*

Test	Samples	Avg	min	max	Error
Melihat Detail Ridesharing	400	2990	85	5486	7,00%
Mencari Ridesharing	400	83694	996	127982	12,25%

Dari hasil uji coba dengan 400 *request* dapat dilihat server mengalami *error* untuk *request* melihat detail *ridesharing* dan mencari *ridesharing*.

4.4.3 Uji Coba Penerimaan Pengguna

Uji coba penerimaan pengguna dilakukan mengacu format *User Acceptance Testing* (UAT) dari *Swiburne University of Technology* [17]. Pengujian dilakukan pada 10 orang responden. Hasil dari uji coba penerimaan pengguna ditampilkan dalam 2 bentuk, yaitu 1) hasil uji coba per fitur aplikasi yang dapat dilihat pada Tabel 8 dan 2) ringkasan saran / komentar dari pengguna yang dapat dilihat pada Tabel 12. Secara umum, responden merasa puas dengan aktivitas yang ditawarkan oleh perangkat lunak utamanya terkait dengan pembuatan *ridesharing* dan mengkonfirmasi permintaan penumpang. Adapun fitur yang dianggap kurang memuaskan adalah bagaimana membagi info *ridesharing* ke media sosial serta membuat peta perjalanan. Selain itu cara *me-reset password* juga mendapatkan sorotan serius dari para pengguna. Pengguna merasa bahwa cara perangkat lunak untuk *me-reset password* masih jauh dari kemudahan. Detil dari hasil ujia coba penerimaan pengguna per fitur dapat dilihat di Tabel 11 dan sarannya bisa dilihat di Tabel 12.

Tabel 11. Hasil Uji Coba Penerimaan Pengguna Per Fitur

No	Nama Aktivitas	Puas	Tidak
1	Membuat Ridesharing	10	0
2	Membagi Info Ridesharing ke Media Sosial	2	8
3	Membuat Peta perjalanan	7	3
4	Mengkonfirmasi permintaan tumpangan	10	0
5	Meregistrasi pemberi tumpangan	5	5

No	Nama Aktivitas	Puas	Tidak
6	Memberi review dan rating ke penumpang	7	3
7	Membatalkan perencanaan ridesharing	9	1
8	Membuat Peta perjalanan	7	3
9	Mencari pasangan ridesharing	8	2
10	Meminta persetujuan untuk menumpang	10	0
11	Menentukan titik dan waktu penjemputan	6	4
12	Meregistrasi penumpang	5	5
13	Memberi review dan rating ke pemberi tumpangan	7	3
14	Membatalkan keikutsertaan ridesharing	10	0
15	Melihat detail informasi ridesharing	10	0
16	Meregistrasi pengguna baru	8	2
17	Melakukan login Pengguna	10	0
18	Mengubah password akun	10	0
19	Me-reset password pengguna	1	9
20	Mengubah data profil pemberi tumpangan	8	2
21	Melihat rincian informasi penumpang	8	2
22	Melihat rincian informasi pemberi tumpangan	10	0

Tabel 12. Ringkasan Komentar dan Saran dari Pengguna

No	Saran / Komentar
#	<i>Ridesharing</i>
1	Informasi yang di share ke media sosial kurang memadai
2	Penentuan titik penjemputan dan penurunan seharusnya bisa sama dengan yang ditawarkan pemberi tumpangan
3	Penentuan titik koordinat harusnya bisa di jalur rutenya
4	Instruksi penggunaan peta kurang
5	Konfirmasi seharusnya ada pop-up agar penumpang / pemberi tumpangan mengetahui
6	Terlalu rumit dalam memilih rute atau titik jemput/turun

No	Saran / Komentar
8	Waktu Jemput seharusnya di atur sama dengan kriteria pencarian, tidak perlu mengisikan ulang
#	Akun
1	Proses registrasinya rumit
#	Review & Rating
1	Nilai awal rating harusnya kosong tidak bintang 5
2	Pewarnaan pada rating kurang jelas, seharusnya bintangnya warna kuning

5. Kesimpulan

Kesimpulan dan saran diharapkan berguna untuk proses pengembangan sistem selanjutnya. Kesimpulan dan saran dari penelitian ini sebagai berikut :

5.1 Simpulan

Kesimpulan yang dapat ditarik dari pengerjaan penelitian ini adalah sebagai berikut:

- 1) Purwarupa aplikasi berbagi kendaraan (*ridesharing*) untuk membantu mengurangi penggunaan kendaraan pribadi berhasil dibuat. Hal ini dibuktikan dari hasil uji coba fungsional yang telah dilakukan. Berdasarkan hasil uji coba fungsional, hampir semua fungsi telah berjalan dengan baik, hanya terdapat 2 fungsi yang masih belum berjalan dengan semestinya yaitu fungsi membagi *ridesharing* ke media sosial dan fungsi mengirim email. Hal tersebut disebabkan karena lingkungan implementasi yang masih menggunakan *localhost*.
- 2) Dari sisi aksesibilitas, purwarupa ini mampu mengatasi akses sekitar 400 pengguna.
- 3) Purwarupa aplikasi berbagi kendaraan (*ridesharing*) untuk membantu mengurangi penggunaan kendaraan pribadi dapat diterima oleh pengguna berdasarkan hasil uji coba penerimaan pengguna yang telah dilakukan. Berdasarkan hasil uji coba tersebut, 76% responden puas dengan fitur dari aplikasi *ridesharing*.

5.2 Saran

Dalam penelitian ini masih terdapat banyak kekurangan yang sekiranya bisa diperbaiki. Oleh karena itu terdapat beberapa saran yang dapat dijadikan masukan untuk penelitian selanjutnya atau untuk pengembangan dari aplikasi berbagi kendaraan (*ridesharing*) ini, yaitu:

- 1) Penelitian atau pengembangan selanjutnya dapat mengikuti saran atau masukan dari responden yang didapat dari uji coba penerimaan pengguna. Catatan atau *feedback* dari responden dapat digunakan untuk menyempurnakan aplikasi ini pada penelitian atau pengembangan selanjutnya.
- 2) Mengoptimasi skema pencocokan rute dan meningkatkan spesifikasi sistem agar program dapat merespon lebih cepat, karena dari hasil uji coba performa diketahui respon *server* bisa mencapai 100 detik.
- 3) Perlu ada uji coba di lingkungan yang lebih luas. Misalnya daerah yang memiliki akses internet tidak *reliable*. Perlu juga ada uji coba dengan jumlah pengguna yang lebih banyak.

6. Daftar Rujukan

- [1] Dinas Komunikasi dan Informatika Jawa Timur, January 2013. [Online]. Available: <http://kominfo.jatimprov.go.id/read/umum/33955>.
- [2] www.kompas.com, October 2014. [Online]. Available: <http://megapolitan.kompas.com/read/2014/10/22/06543931/Jakarta.Bogor.dan.Bandung.Kota.Paling.Rawan.Macet>.
- [3] A. Tahir, "Angkutan Massal Sebagai Alternatif Mengatasi Persoalan Kemacetan Lalu-lintas Kota Surabaya,," SMARTek, vol. 3, no. 3, pp. 169-183, 2005.
- [4] A. Wokje and M. Keall, "Effectiveness of a web-based intervention to encourage carpooling to work : A case study of Wellington, NewZealand,," Transport Policy, vol. 21, no. 6, pp. 45-51, 2012.
- [5] J. R. Evans, W. K. Jackson and M. P. Thomas, "Planning and analysis of a ridesharing evaluation study," Socio-Economic Planning Sciences, vol. 19, no. 1, pp. 41-49, 1985.
- [6] M. Furuhashi, M. Dessouky, F. Ordóñez, M.-E. Brunet, X. Wang and S. Koenig, "Ridesharing: The state-of-the-art and future directions," Transportation Research Part B, vol. 57, p. 28–46, 2013.
- [7] S. H. Jacobson and D. M. King, "Fuel saving and ridesharing in the US: Motivations, limitations, and opportunities," Transportation Research Part D: Transport and Environment, vol. 14, no. 1, p. 14–21, January 2009.
- [8] S. Seyedabrishami, A. Mamdoohi, A. Barzegar and S. Hasanpour, "Impact of Carpooling on Fuel Saving in Urban Transportation: Case Study of Tehran," in Proceedings of EWGT2012 - 15th Meeting of the EURO Working Group on Transportation, 2012.
- [9] C.-H. Lin, M.-K. Jiau and S.-C. Huang, "A cloud computing framework for real-time carpooling services," Information Science and Service Science and Data Mining (ISSDM), vol. 6, no. 51, pp. 266-271, 2012.
- [10] M. Furuhashi, M. Dessouky, F. Ordóñez, M.-E. Brunet and X. Wang, "Ridesharing: The state-of-the-art and future directions," Transportation Research Part B, vol. 57, pp. 28-46, 2013.
- [11] N. A. Agatz, A. L. Erera, M. W. Savelsbergh and X. Wang, "Dynamic ride-sharing: A simulation study in metro Atlanta," Transportation Research Part B, vol. 45, pp. 1450-1464, 2011.
- [12] J. Friginal, S. Gams, J. Guiochet and M.-O. Killijian, "Towards privacy-driven design of a dynamic carpooling system," Pervasive and Mobile Computing, vol. 14, pp. 71-82, 2014.
- [13] S. Galland, N. Gaud, . A.-U.-H. Yasar, L. Knapen, D. Janssens and O. Lamotte, "Galland, Stephane, et al., Simulation Model of Carpooling with the Janus Multiagent Platform," in The 2nd International Workshop on Agent-based Mobility, Traffic and Transportation, 2013.
- [14] L. Knapen, I. B.-A. Hartman, D. Keren, A.-U.-H. Yasar, S. Cho, T. Bellemans, D. Janssens and G. Wets, "Scalability issues in optimal assignment for carpooling," Journal of Computer and System Sciences, vol. 81, pp. 568-584, 2015.
- [15] A.-J. Fougères, P. Canalda, A. Samaali and T. Ecarot, "A Push Service For Carpooling," Green Computing and Communications (GreenCom), pp. 686-691, 2012.
- [16] S. Cho, A.-U.-H. Yasar, L. Knapen, T. Bellemans, D. Janssens and G. Wets, "A Conceptual Design of an Agent-based Interaction Model for the Carpooling Application," Procedia Computer Science, vol. 10, pp. 801-807, 2010.
- [17] Swinburn University of Technology. [Online]. Available: <https://www.google.co.id/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwil64eh7uzRAhWMq48KHxvHBxIQFgg2MAA&url=http%3A%2F%2Fwww.its.swinburne.edu.au%2Fprojects%2Ftemplates%2FUATChcklistTemplatev1.1-%5BProjectName%5D-%5Bver%5D-%5BYYYYMM>.

