

# Pembuatan Aplikasi Manajemen Kendaraan berbasis Android

Deryan Thimoti Yudianto<sup>1</sup>, Agustinus Noertjahyana<sup>2</sup>, Justinus Andjarwirawan<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra

Jl. Siwalankerto 121-131, Surabaya 60236

Telp. (031)-2983455, Fax. (031)-8417658

E-mail: deryanthimoti@gmail.com<sup>1</sup>, agust@petra.ac.id<sup>2</sup>, justin@petra.ac.id<sup>3</sup>

## ABSTRAK

Banyak pengguna kendaraan yang menggunakan *smartphone* berbasis Android. Dalam menggunakan kendaraan, pengguna kendaraan harus mengetahui bagaimana mengelola kendaraan dengan baik dan benar, agar mereka dapat memastikan kendaraannya dalam keadaan yang baik.

Aplikasi ini dibuat dengan menggunakan bahasa pemrograman Java dan menggunakan aplikasi Android Studio. Tampilan aplikasi ini menggunakan Material Design. Fitur *Backup* dan *Restore* menggunakan Google Drive dan Dropbox sebagai *online storage*. Fitur *Reminder* menggunakan Google Calendar sebagai media pencatatan *event*.

Hasil akhir dari aplikasi ini adalah pengguna aplikasi dapat memantau pengeluarannya atas penggunaan kendaraan. Selain itu pengguna aplikasi juga dapat menyimpan datanya pada *online storage* tanpa harus membayar terlebih dahulu.

**Kata Kunci:** Manajemen kendaraan, Android, Android Studio, Google Drive, Dropbox, Google Calendar

## ABSTRACT

*There are many vehicle users that are using Android based smartphones. When using vehicle, the vehicle users must know how to manage their vehicle so their vehicles is always in a good condition.*

*This application is created using Java programming language and using Android Studio application. The interface of this application uses Material Design. Backup and Restore feature uses Google Drive and Dropbox as online storage. Reminder feature uses Google Calendar as an event recording media.*

*The final result of the application is that users can view their expenses as the result of using their vehicles. Besides, the users can save their data on the online storage without need to pay for anything.*

**Keywords:** Vehicle Management, Android, Android Studio, Google Drive, Dropbox, Google Calendar

## 1. PENDAHULUAN

Kendaraan pribadi seperti mobil dan sepeda motor telah menjadi bagian hidup manusia sehari-harinya. Tanpa adanya kendaraan tersebut, hidup akan menghadapi banyak kesulitan karena tidak dapat melakukan hal-hal yang memerlukan kendaraan, seperti pergi ke luar kota, pergi ke Mall, dan lain-lain. Dalam menggunakan kendaraan, pengguna harus mengetahui bagaimana memmanage kendaraannya, agar mereka dapat memastikan kendaraannya dalam keadaan yang baik. Kegiatan yang termasuk mengelola kendaraan meliputi mengisi bahan bakar, servis berkala, penggantian *spare part*, penggantian oli, dan lain-lain. Untuk memudahkan pengguna kendaraan dalam mengelola kendarannya, maka dibuatlah aplikasi manajemen kendaraan. Berbagai macam aplikasi manajemen kendaraan (khususnya mobil) telah dibuat demi

memudahkan pengguna kendarana dalam mengelola kendaraannya.

Ada beberapa aplikasi manajemen kendaraan yang ada di Google Play Store. Namun, dari aplikasi-aplikasi tersebut, terdapat kelebihan dan kekurangan masing-masing yang saling melengkapi apabila digabungkan. Salah satu kelemahan universal dari aplikasi manajemen kendaraan yang sudah ada adalah tidak adanya sistem *backup* yang baik, sehingga apabila pengguna mengganti *smartphone*, maka data pada aplikasi manajemen kendaraan akan ikut hilang, karena hanya disimpan di *device* saja. Selain itu, kebanyakan aplikasi manajemen kendaraan yang ada di Google Play Store ditargetkan untuk pengguna mobil, sedangkan masyarakat Indonesia banyak yang menggunakan sepeda motor sebagai kendaraan sehari-harinya.

Untuk itulah pada skripsi ini, dibuat sebuah aplikasi manajemen kendaraan berbasis Android yang merupakan perpaduan dari kelebihan-kelebihan aplikasi manajemen kendaraan yang lainnya.

## 2. TINJAUAN PUSTAKA

### 2.1 Java

Java adalah bahasa pemrograman yang dikembangkan oleh Sun Microsystems. Java merupakan bahasa pemrograman berorientasi objek dan telah digunakan komputer dan *smartphone*. Java menggunakan *syntax* yang mirip dengan bahasa pemrograman C/C++. Kemiripan antara Java dan C/C++ adalah adanya beberapa *syntax* dari C/C++ yang sama dengan Java dan juga Java menggunakan operator aritmatika yang sama dengan C/C++, sehingga Java akan lebih mudah dipahami apabila kita telah memahami C/C++. Akan tetapi, Java telah didesain untuk menjadi bahasa pemrograman yang lebih aman daripada C/C++ dengan tidak diijinkannya *overloading operator* dan menghilangkan *pointer*.

Java terdiri dari *virtual machine* dan *execution environment*. Pada *virtual machine* terdapat prosesor berbasis *software* yang menyajikan sekumpulan instruksi. *Execution environment* berisi *library* untuk menjalankan program dan berinteraksi dengan sistem operasi yang mendasarinya. [2].

### 2.2 Dropbox

*Dropbox* adalah layanan *online file hosting* gratis yang memungkinkan penggunanya untuk mengunduh video, foto, dokumen, dan *file* untuk penyimpanan *backup*. Segala sesuatu yang ditambahkan ke *Dropbox* secara otomatis akan disinkronisasi ke suatu akun *Dropbox* yang telah ditentukan yang memungkinkan akses dari komputer atau *smartphone*. *Dropbox* memudahkan *file sharing* dan ketakutan penggunanya akan kehilangan data apabila ada penghapusan data secara tidak sengaja atau kerusakan.

*Dropbox* menyediakan *client-side software* gratis kepada penggunanya, yang membuat *Dropbox folder* setelah instalasi. Pengguna *Dropbox* dapat melakukan *signup* untuk akun gratis ataupun *Pro* dan *business*. Akun gratis memungkinkan

maksimum *upload* sebesar 20 GB per harinya, sedangkan *Pro* dan *business* memiliki maksimum *upload* sebesar 200 GB per harinya. Keuntungan dari menggunakan *Dropbox* adalah akses *file* dari mana saja, *file backup* secara instan, dapat membagikan *file* dengan siapapun, *public file sharing*, dan berbagai macam cara untuk mengakses *file*. [9].

### 2.3 Google Drive

*Google Drive* adalah tempat di mana pengguna dapat menyimpan *file* secara *online* dan mengaksesnya dari manapun. Saat menggunakan *Google Drive*, *file* yang anda miliki disimpan di *web*, tidak di *hard drive*. Ketika *file* yang dimiliki oleh pengguna sudah *online*, pengguna dapat mengakses *file* tidak hanya dari komputer saja, tetapi juga dari *smartphone* ataupun tablet. Pengguna juga bisa membagikan *file* dengan orang-orang lain untuk saling berkolaborasi dan menyelesaikan suatu pekerjaan bersama. Semua tipe *file* dapat disimpan di *Google Drive*, seperti foto, video, PDF, presentasi, dokumen teks, dan lain-lain. *Google Drive* juga menyediakan fitur opsional *sync* yang berjalan di komputer pengguna untuk mensinkronisasi *folder* pada *hard drive* pengguna dan *Google Drive* di *web*. Untuk menggunakan *Google Drive*, pengguna harus memiliki akun *Google* berupa *Gmail*. [8].

*Google Drive* memungkinkan penggunanya untuk melakukan *preview* pada *file* seperti gambar dan *file* dalam bentuk teks seperti pdf maupun *word*, namun tidak untuk melakukan perubahan pada *file-file* tersebut. Pada Android, *Google Drive* mampu melakukan *scan* pada suatu dokumen menggunakan kamera, sehingga suatu gambar dapat otomatis menjadi *file* pdf dan akan tersimpan pada *My Drive*. [5]

### 2.4 Google Drive

*Google Drive* adalah tempat di mana pengguna dapat menyimpan *file* secara *online* dan mengaksesnya dari manapun. Saat menggunakan *Google Drive*, *file* yang anda miliki disimpan di *web*, tidak di *hard drive*. Ketika *file* yang dimiliki oleh pengguna sudah *online*, pengguna dapat mengakses *file* tidak hanya dari komputer saja, tetapi juga dari *smartphone* ataupun tablet. Pengguna juga bisa membagikan *file* dengan orang-orang lain untuk saling berkolaborasi dan menyelesaikan suatu pekerjaan bersama. Semua tipe *file* dapat disimpan di *Google Drive*, seperti foto, video, PDF, presentasi, dokumen teks, dan lain-lain. *Google Drive* juga menyediakan fitur opsional *sync* yang berjalan di komputer pengguna untuk mensinkronisasi *folder* pada *hard drive* pengguna dan *Google Drive* di *web*. Untuk menggunakan *Google Drive*, pengguna harus memiliki akun *Google* berupa *Gmail*. [8].

*Google Drive* memungkinkan penggunanya untuk melakukan *preview* pada *file* seperti gambar dan *file* dalam bentuk teks seperti pdf maupun *word*, namun tidak untuk melakukan perubahan pada *file-file* tersebut. Pada Android, *Google Drive* mampu melakukan *scan* pada suatu dokumen menggunakan kamera, sehingga suatu gambar dapat otomatis menjadi *file* pdf dan akan tersimpan pada *My Drive*. [5]

### 2.5 SQLite

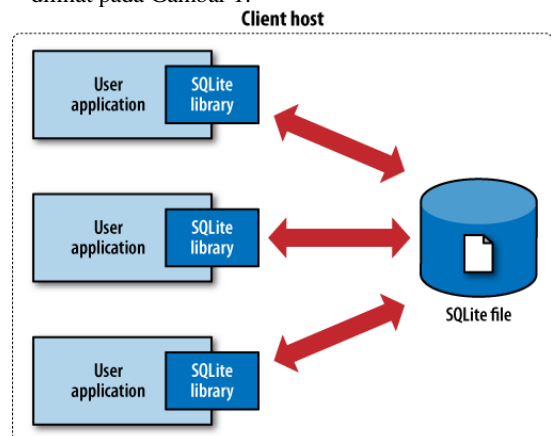
SQLite merupakan sebuah *public-domain software* yang menyediakan RDBMS (*Relational Database Management Systems*). RDBMS sendiri digunakan untuk menyimpan *record* yang ditentukan oleh pengguna ke dalam tabel. "Lite" pada SQLite tidak mengarah pada kemampuan SQLite, melainkan lebih ringan pada tingkat kesulitan pada *setup*, *administrative overhead*, dan penggunaan sumber daya. [4]

SQLite ditulis dalam bahasa C, dan dikembangkan oleh D. Richard Hipp pada tahun 2000 sebagai bagian dari kontrak dengan Angkatan Laut Amerika Serikat yang

diimplementasikan oleh Jendral Dynamics. SQLite adalah domain umum, jadi bisa digunakan oleh semua orang. Salah satu perbedaan SQLite dengan DBMS lainnya adalah SQLite didesain untuk user tunggal. Banyak DBMS lainnya yang dapat mengelola banyak user dan memiliki fitur keamanan yang mengizinkan atau memblokir perintah SQL tertentu. [1]

SQLite memiliki ciri-ciri sebagai berikut :

- *Serverless*  
SQLite tidak memiliki arsitektur *client/server*, sehingga SQLite tidak memerlukan server terpisah untuk beroperasi. *Library* SQLite dapat mengakses file pada penyimpanan secara langsung. Arsitektur SQLite dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur SQLite yang bersifat *serverless*

- *Zero Configuration*  
Pada sudut pandang pengguna, SQLite tidak memerlukan apapun untuk instalasi dan tidak memiliki apapun untuk dikonfigurasi.
- *Cross-Platform*  
Keseluruhan *database instance* berada pada satu file *cross-platform* yang tidak memerlukan administrasi
- *Self-Contained*  
Satu *library* berisi seluruh sistem database, yang terintegrasi secara langsung pada *host application*.
- *Small Runtime Footprint*  
*Default build* dari SQLite mempunyai ukuran yang tidak sampai satu *megabyte* dan hanya memerlukan beberapa *megabyte* memori. Dengan sedikit pengaturan, ukuran *library* dan memori yang digunakan dapat dikurangi secara signifikan.
- *Transactional*  
Transaksi SQLite sepenuhnya bersifat *ACID-compliant*, sehingga memungkinkan akses yang aman dari beberapa proses dan *thread*.
- *Full-Featured*  
SQLite mendukung kebanyakan bahasa *query* yang dapat ditemukan pada standar SQL92.
- *Highly Reliable*  
Tujuan dari database adalah mempertahankan keamanan dan keteraturan data. Untuk mempertahankan tingkat keandalan (*reliability*) yang tinggi, *library* SQLite diuji secara agresif sebelum tiap kali perlisian. Pengujian SQLite terdiri dari lebih dari sepuluh juta pengujian unit dan pengujian *query*. Pengujian ini membuat *bug* pada SQLite terhitung rendah. Pengujian ini juga membuat SQLite dapat mendukung *backward compatibility*. [4].

## 2.6 Google Calendar

Google Calendar adalah aplikasi web dan *mobile* yang bertujuan untuk mengelola waktu. Seperti namanya, Google Calendar dibuat oleh Google. Google Calendar dirilis pada tanggal 13 April 2006, dan berhasil keluar dari tahap beta pada Juli 2009. Untuk menggunakan Google Calendar, pengguna harus memiliki akun Google terlebih dahulu, namun aplikasi *mobile* Google Calendar dapat mendukung sinkronisasi dengan servis kalender pihak ketiga. [3]

Aplikasi yang dibuat pada tugas akhir ini menggunakan Google Calendar sebagai media untuk mencatat pengingat (*reminder*) yang dimasukkan oleh pengguna. Pengguna hanya perlu memasukkan tanggal, waktu, dan kapan *reminder* berbunyi, dan *reminder* juga akan terbuat secara otomatis di Google Calendar.

## 2.7 Manajemen Armada

Manajemen armada (*Fleet Management*) adalah fungsi yang mengawasi, mengkoordinasi, dan memfasilitasi berbagai macam transportasi dan kegiatan yang berhubungan dengan transportasi. Manajemen armada yang efektif bertujuan untuk mengurangi dan meminimalisir biaya atas sumber daya kendaraan seperti bahan bakar, *spare part*, dan lain-lain. Aspek-aspek yang termasuk dalam manajemen armada adalah efisiensi biaya, keselamatan di jalan, keselamatan dalam penggunaan kendaraan, manajemen performa, dan lain-lain. Untuk mencapai aspek-aspek di atas, manajemen armada mengambil kebijakan yang mengharuskan setiap kendaraan mempunyai *log book* dan setiap minggu, *log book* harus diperiksa, mencatat biaya pengeluaran dan pengisian bahan bakar ke dalam *log book* dengan menyertakan pencatatan *odometer* setiap kali melakukan pencatatan ke dalam *log book*. Ketika melakukan servis terhadap kendaraan, ada baiknya servis dijadwalkan secara berkala. Ketika servis, pengguna kendaraan wajib mencatat pekerjaan yang dilakukan oleh mekanik, bagian kendaraan dan cairan yang digunakan, dan biaya pada setiap pengerjaan. [6].

## 2.8 Sistem Manajemen Armada

Sistem Manajemen Kendaraan dibuat dengan terstruktur agar dapat memungkinkan untuk mendapatkan informasi pada berbagai aspek dalam penggunaan, pemeliharaan, dan

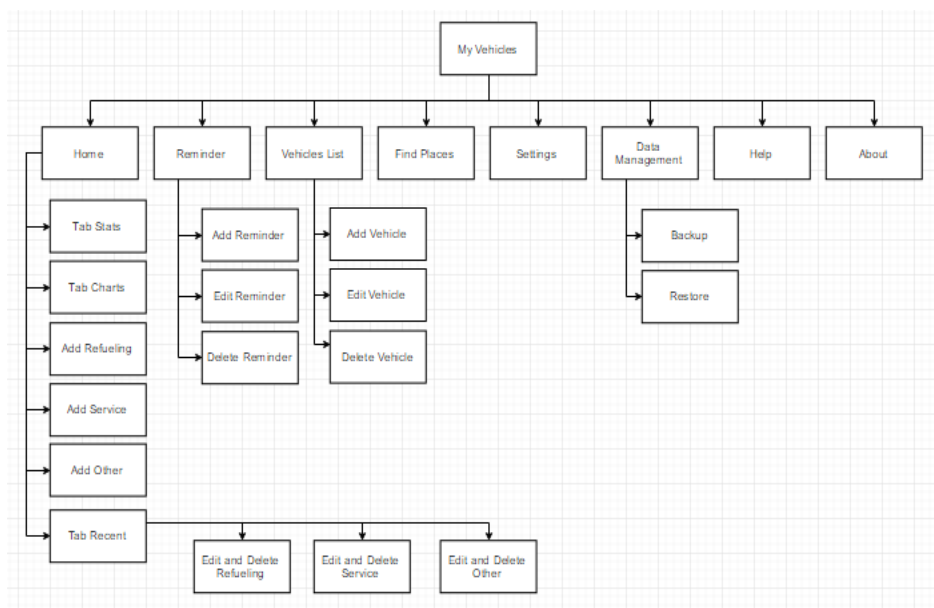
operasional kendaraan. Informasi yang perlu didapatkan antara lain jarak yang ditempuh oleh kendaraan, tujuan yang dicapai oleh kendaraan, konsumsi bahan bakar, perbaikan dan pemeliharaan kendaraan, tingkat konsumsi pada *spare part*, dan perencanaan servis berkala. Laporan dapat dihasilkan setiap seminggu sekali, dua minggu sekali, satu bulan sekali, sesuai dengan kebutuhan. [6]

Berbagai macam tipe servis kendaraan antara lain

- *Basic Service*  
*Basic Service* meliputi penggantian oli dan *filter*, penggantian cairan kendaraan (contoh minyak rem, cairan *wiper*, dan lain-lain), pengecekan rem dan kondisi mesin, dan pengecekan *spare part* kendaraan. *Basic service* dilakukan pada kilometer berapapun.
- *Full Service*  
*Full Service* meliputi *Basic Service*, namun akan dilakukan cek sebanyak 15-30 kali pada komponen kendaraan dibandingkan dengan *Basic Service*. Biasanya pada *Full Service* akan ada pengecekan tambahan seperti pengecekan kampas rem, *shock breaker*, dan lain-lain. *Full Service* biasanya dilakukan pada kilometer 20.000 atau 60.000
- *Major Service*  
*Major Service* meliputi keseluruhan *Full Service*, namun dengan pengecekan tambahan, yaitu keseluruhan bagian dari kendaraan. Pada *Major Service* biasanya akan dilakukan pengecekan evaporator, *bearing roda*, pembersih rem, dan lain-lain. *Major Service* biasanya dilakukan pada kilometer 40.000 dan 80.000 beserta kelipatannya.[7][10]

## 3. ANALISIS DAN DESAIN

Secara garis besar, aplikasi Android memiliki 8 menu pada *Navigation Drawer*, yaitu *Home*, *Reminder*, *Vehicles List*, *Find Places*, *Help*, *Data Management*, *Settings*, dan *About*. Menu lain seperti *Add Vehicle*, *Backup*, *Restore*, dan lain-lain berada pada level sub menu. Desain hirarki sistem pada Android dapat dilihat pada Gambar 2 di bawah ini.

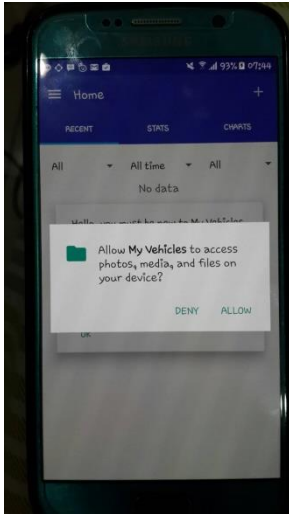


Gambar 2. Hierarki sistem aplikasi My Vehicles

## 4. HASIL DAN PEMBAHASAN

### 4.1 Halaman Awal Aplikasi

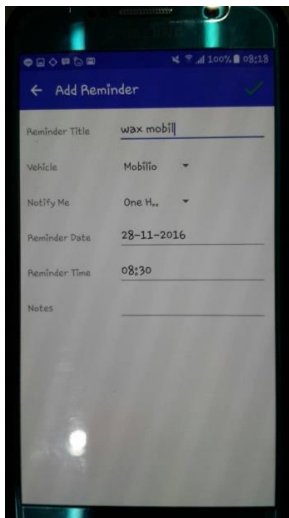
Apabila pengguna pertama kali membuka aplikasi, maka akan muncul *dialog* selamat datang. Khusus untuk pemilik Android dengan versi 6.0 (*Marshmallow*), akan muncul *dialog* untuk menanyakan *permission* untuk melakukan *write* pada *external storage*. Namun untuk pemilik Android dengan versi di bawah 6.0. (sebelum *Marshmallow*), maka *dialog* untuk menanyakan *permission* tidak akan muncul, karena pada Android sebelum versi 6.0 *permission* akan otomatis diberikan. Tampilan halaman awal saat membuka aplikasi dapat dilihat pada Gambar 3.



Gambar 3. Tampilan halaman awal aplikasi

### 4.2 Add Reminder

Untuk melakukan penambahan *reminder*, maka pengguna harus masuk ke halaman *reminder* terlebih dahulu. Apabila pengguna mengisi *reminder*, maka data *reminder* akan masuk ke dalam database dan *reminder* akan otomatis terbuat pada Google Calendar. Tampilan halaman *Add Reminder* dapat dilihat pada Gambar 4, sedangkan tampilan hasil sinkronisasi *reminder* dapat dilihat pada Gambar 5.



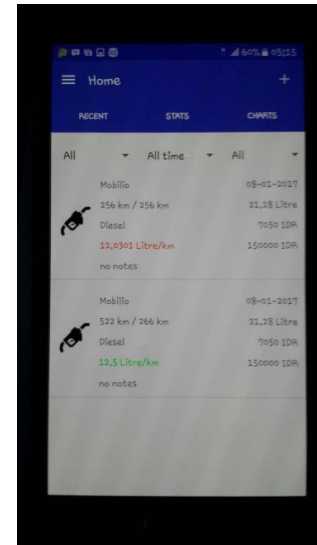
Gambar 4. Tampilan Add Reminder



Gambar 5. Tampilan hasil sinkronisasi dengan Google Calendar

### 4.3 Perhitungan Jarak

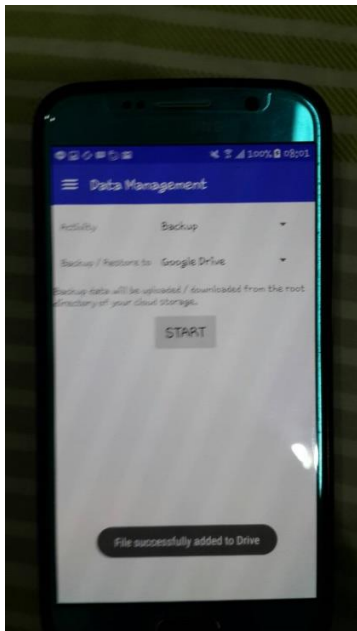
Untuk dapat melakukan perhitungan jarak pada *refueling*, maka pengguna diharuskan untuk mengisi kolom *Initial Mileage* pada kendaraan. Apabila *Initial Mileage* dikosongkan oleh pengguna, maka nilai *default* dari *Initial Mileage* adalah 0. Pengujian dilakukan dengan memasukkan data *refueling* pada hari yang sama, namun dengan odometer yang berbeda. Hasil perhitungan jarak dapat dilihat pada Gambar 6.



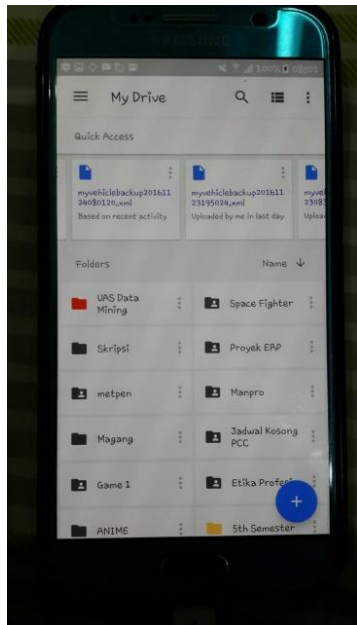
Gambar 6. Tampilan Perhitungan Jarak

### 4.4 Backup Data

Untuk melakukan *backup* data, maka pengguna pertama-tama harus masuk ke halaman *Data Management*, lalu memilih *backup*. Apabila pengguna ingin membackup data, maka *file XML* hasil *backup* akan diupload ke Google Drive. *File XML* berisi data pada database yang telah dikonversi ke dalam bentuk XML. Tampilan hasil *backup* pada aplikasi dapat dilihat pada Gambar 7, sedangkan hasil *backup* pada Google Drive dapat dilihat pada Gambar 8



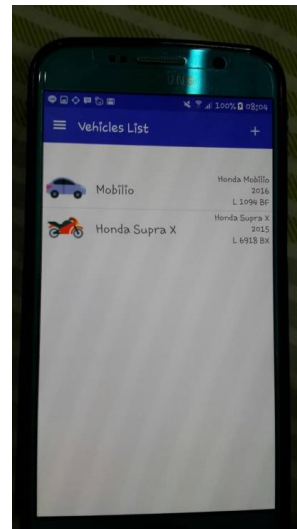
Gambar 7. Tampilan hasil *backup* pada halaman *Data Management*



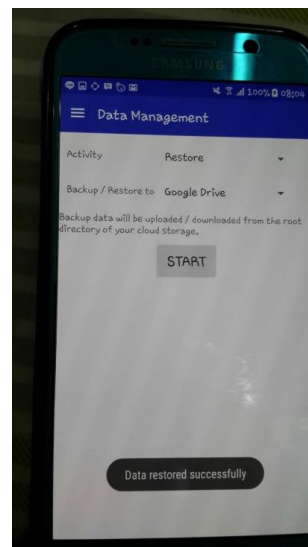
Gambar 8. Tampilan hasil *backup* pada Google Drive

#### 4.5 Restore Data

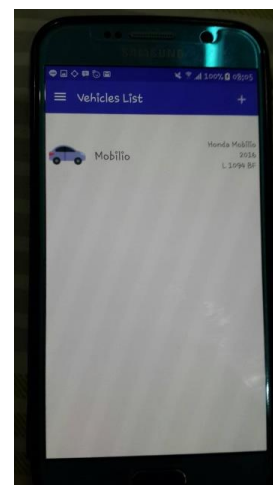
Apabila pengguna ingin merestore data dari Google Drive, maka pertama-tama pengguna harus masuk ke halaman *Data Management* terlebih dahulu, lalu memilih *Download* pada *spinner*. Setelah itu, sistem akan melakukan *download* pada *online storage*. *File* yang dipilih adalah *file* dengan *timestamp* terbaru. Pada pengujian dilakukan dengan menambahkan data *dummy* pada data kendaraan, lalu melakukan *restore*. Tampilan data *Vehicles* pada halaman *Vehicles List* yang diisi dengan data *dummy* dapat dilihat pada Gambar 9. Tampilan hasil *restore* pada halaman *Data Management* dapat dilihat pada Gambar 10, sedangkan hasil *restore* pada halaman *Vehicle List* dapat dilihat pada Gambar 11.



Gambar 9. Tampilan *Vehicle List* dengan tambahan data *dummy*



Gambar 10. Tampilan hasil *restore* pada halaman *Data Management*



Gambar 11. Tampilan hasil *restore* data pada halaman *Vehicles List*

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan hasil perancangan dan pembuatan aplikasi My Vehicles, dapat diambil kesimpulan antara lain :

- Aplikasi dapat melakukan *backup* dan *restore* pada Google Drive.
- Aplikasi dapat mengintegrasikan sistem *reminder* yang dimasukkan oleh pengguna ke dalam Google Calendar.
- *Backup* tidak dapat dilakukan pada *device* yang tidak memiliki *external storage*.
- Sistem *reminder* untuk pengisian bahan bakar berdasarkan pola penggunaan kendaraan dapat mengganggu apabila pengguna kendaraan sering mengisi bahan bakar dalam kondisi tidak penuh. Sebaiknya Aplikasi My Vehicles dibuat agar dapat mengolah data untuk masa depan.

### 5.2 Saran

Saran yang diberikan untuk pengembangan aplikasi di kemudian hari antara lain :

- Aplikasi My Vehicles sebaiknya memberikan pilihan kepada pengguna untuk dapat melakukan *backup* dan *restore* data secara parsial.
- Aplikasi My Vehicles sebaiknya dapat menggunakan Dropbox sebagai media untuk *backup* dan *restore* data.
- Apabila pengguna melakukan *restore* pada *reminder* yang telah dihapus, maka *reminder* pada Google Calendar yang telah dihapus tidak dapat dikembalikan seperti semula, karena apabila *reminder* telah dihapus, maka *reminder* tersebut tidak benar-benar dihapus, melainkan disembunyikan (*hidden*), sehingga tidak dapat membuat *reminder* berdasarkan *event id* yang sudah ada di dalam database. Sebaiknya *reminder* dibuat lebih fleksibel.
- Desain *interface* diperbaiki, sebaiknya jangan menggunakan warna *template* Material Design

## 6. DAFTAR REFERENSI

- [1] Feiler, Jesse. 2015. *Introducing SQLite for Mobile Developers*. New York, United States : Apress Media, LLC.
- [2] Friesen, Jeff. 2013. *Learn Java for Android Development (2nd ed.)*. New York, United States : Apress Media, LLC.
- [3] Google Calendar 2006, 14 April. wikipedia.org : Google Calendar. Diakses 23 November 2016, dari [https://en.wikipedia.org/wiki/Google\\_Calendar](https://en.wikipedia.org/wiki/Google_Calendar)
- [4] Kreibich, Jay A. 2010. *Using SQLite*. Sebastopol, California, United States : O'Reilly Media, Inc.
- [5] Lamont, Ian 2015. *Google Drive and Docs in 30 Minutes (2<sup>nd</sup> Edition)*. Boston, United States : i30 Media Corporation.
- [6] Logistic Capacity Assessment. 2015, 24 Juni. dlca.logcluster.org: Fleet Management. Diakses 18 September 2016, dari [dlca.logcluster.org/display/LOG/Fleet+Management](http://dlca.logcluster.org/display/LOG/Fleet+Management)
- [7] Motangel. 2012. motangel.co.uk : What Is Car Servicing. Diakses 11 Januari 2017, dari [www.motangel.co.uk/article/car-servicing-explained/](http://www.motangel.co.uk/article/car-servicing-explained/)
- [8] Procopio, Mike. 2013. *Instant Google Drive Starter*. Birmingham, United Kingdom : Packt Publishing Ltd.
- [9] Sullivan, Warren R. 2015. *Dropbox For Beginners*. Gamma Mouse. Diakses 12 Mei 2016, dari Google Books
- [10] Wiyono, Eko. 2015. www.mobilku.org: Sparepart yang Diganti Tiap Beberapa Kilometer. Diakses 11 Januari 2017, dari [www.mobilku.org/2015/03/sparepart-yang-diganti-tiap-beberapa.html](http://www.mobilku.org/2015/03/sparepart-yang-diganti-tiap-beberapa.html)