APLIKASI KEAMANAN SEPEDA MOTOR BERBASIS *GLOBAL POSITIONING SYSTEM* MENGGUNAKAN *ANDROID*

John Adler[1], Aprianti Putri Sujana[2], Tengku Charlie Hardiansyah[3]

[123]Jurusan Sistem Komputer Universitas Komputer Indonesia, Bandung

john.adler@email.unikom.ac.id[2], aprianti.putri.sujana@email.unikom.ac.id[3], crlie@hotmail.com [1]

ABSTRAK

Kriminalitas, khususnya pencurian sepeda motor selalu menjadi momok menakutkan di manapun termasuk di Provinsi Jawa Barat. Berbagai cara dan modus operandi yang digunakan oleh para pelaku dalam melakukan aksinya cukup meresahkan para pemilik sepeda motor. Perancangan aplikasi pengamanan sepeda motor dengan smartphone android dimaksudkan sebagai pengamanan tambahan bagi pemilik sepeda motor. Smartphone android yang saat ini dominan dipakai menjadi alat untuk memantau sepeda motor. Di mana nantinya smartphone pada sepeda motor menjadi transmitter dan smartphone pada pemilik sepeda motor menjadi receiver. Ketika melewati radius ± 10 meter dari posisi awal, transmitter akan mengirimkan update posisi setiap 30 detik dan memunculkan notifikasi ke receiver bahwa sepeda motor berpotensi dicuri. Tidak hanya kepada receiver, transmitter juga akan mengirimkan update koordinat dan data pemilik sepeda motor ke database sehingga dapat diakses dengan personal komputer contohnya. Hasil pengujian terhadap aplikasi secara umum mampu berjalan dengan baik.

Kata kunci : Sepeda motor, Smartphone, Android, Transmitter, Receiver

1. PENDAHULUAN

Dewasa ini tindak kriminalitas semakin marak terjadi terutama terkait masalah pencurian kendaraan bermotor atau biasa disebut curanmor. Menurut data Polrestabes Bandung sepanjang 2016, telah terjadi 460 kasus. Namun ironisnya, pengungkapan kasus tersebut hanya 89 kasus atau 19 persen dari total kasus yang terjadi. Hal ini, dikatakan oleh Kapolrestabes Bandung, Kombes Pol. Hendro Pandowo, di Mapolrestabes Bandung, pada Sabtu, 31 Desember 2016.[4]

Curanmor terjadi karena pengamanan kendaraan seperti, penguncian pada leher motor, kunci ganda dengan menggunakan gembok ataupun pemilik sepeda motor yang lupa dalam menerapkannya. Sehingga perlu dibuatnya sebuah aplikasi keamanan sepeda motor yang terintegrasi dengan smartphone android yang cukup banyak digunakan oleh masyarakat saat ini. Aplikasi ini mampu mengetahui keberadaan dari sepeda motor yang dicuri berdasarkan koordinat dan lokasi yang dikirim oleh smartphone yang diletakkan pada sepeda motor (*transmitter*) ke aplikasi pada smartphone pemilik (*receiver*). Selain koordinat dan lokasi, data pemilik sepeda motor juga terkirim dan tersimpan ke database yang dapat diakses dari *web*.

2. PERANCANGAN

Dasar Teori

Berikut adalah tinjauan pustaka yang terdapat pada penelitian ini.

Android Operating System (Android OS)

Awalnya *Android* dikembangkan oleh Android Inc. Google pun membantu finansialnya dan membelinya pada tahun 2005. Kemudian pada November 2007, dibentuk sebuah konsorsium yang dinamakan Open Handset Alliance untuk membuat dan mengembangkan standar terbuka (*open source*) bagi perangkat *mobile.*

Produk pertama dari konsorsium ini adalah *Android Platform. Platform* ini menjadi *open source* yang dipimpin oleh Google dan di bawah naungan Open Handset Alliance. Tujuan dari *Android Open Source Project* ini adalah menyediakan *platform* terbuka sehingga siapa saja dapat mengembangkan pengalaman pengguna (*user experience*) khususnya perangkat *mobile.* [5]

Android Studio

Android Studio adalah sebuah peralatan (*tool*) atau yang biasa disebut dengan Integrated Development Environment (IDE) untuk pengembangan aplikasi yang berbasis *Android* dan dikembangkan oleh Google secara mandiri yang diperkenalkan pada 16 Mei 2013 dalam acara Google I/O. Hingga saat ini Android Studio telah mencapai versi 2.2.[7]

Google Map

Google Map adalah sebuah layanan pemetaan yang disediakan oleh Google secara gratis. Menawarkan peta dan gambar satelit untuk seluruh dunia. Google juga telah menyediakan *Application Programming Interface (API) Google Map*, layanan untuk para pengembang aplikasi *Android* yang ingin menggunakan layanan perpetaan ini. Terdapat dua API *Google Map* yang disediakan, yaitu standar dan untuk bisnis.[2]

Global Positioning System (GPS)* dan *Assisted-GPS

Global Positioning System atau *GPS* adalah sistem yang dibuat untuk menentukan letak di permukaan bumi dengan bantuan penyelarasan (*synchronization*) sinyal satelit. [3]

Cara kerja dari GPS meliputi 5 langkah, yaitu :

1. Memakai perhitungan triangulasi dari satelit.
2. Untuk perhitungan triangulasi, GPS mengukur jarak menggunakan *travel time* sinyal radio.
3. Untuk mengukur *travel time*, GPS memerlukan akurasi waktu yang tinggi.
4. Untuk perhitungan jarak, kita harus tahu dengan pasti posisi satelit dan ketinggian pada orbitnya.
5. Harus mengoreksi *delay* sinyal waktu perjalanan di atmosfer sampai diterima *receiver*.

Sedangkan *Assisted GPS* adalah jenis *GPS* yang tertanam pada perangkat *mobile* dan mengandalkan sebuah server bantuan, selain dari satelit. *A-GPS* dikembangkan untuk meningkatkan kinerja *GPS*. Server bantuan tersebut akan memberikan informasi tambahan ke perangkat yang dapat membantu dalam pencarian lokasi. Hal ini berguna apabila *chip GPS* mungkin mengalami kesulitan dalam mendapatkan sinyal satelit. [6]

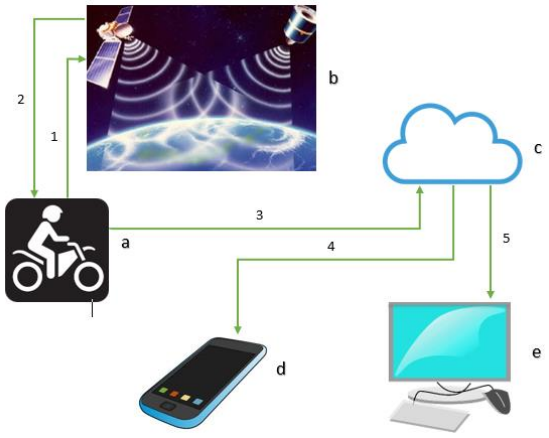
PHP dan *MySql*

PHP atau *Hypertext Prepocessor* merupakan salah satu bahasa pemrograman dari sisi *server (web server-side)*, terintegrasi dengan HTML dan mampu membuat halaman *website* menjadi dinamis. Arti dari dinamis sendiri adalah halaman yang dibuat dan ditampilkan dari sisi *client* pada suatu browser atau pengguna akan mendapat pembaruan sesuai dengan apa yang terdapat di sisi *server* tersebut.

MySql merupakan salah satu *Database Management System* (DBMS) *SQL* yang bersifat gratis di bawah lisensi GNU General Public License (GPL). Selain *MySQL*, banyak juga jenis DBMS seperti *Oracle*, *MS SQL*, *Postagre SQL* dan lainnya. [1]

Perancangan *Software*

Perancangan *software* ditujukan untuk menggambarkan bagaimana aplikasi nantinya akan berjalan dan memenuhi fungsi fungsi : adanya *push notification* notifikasi pada *receiver*, tampilnya lokasi awal dan lokasi saat ini dalam *google map* pada perangkat serta keterangan lain dari *transmitter* ke *receiver*, dan terkirimnya data-data dari *transmitter* ke *database.*



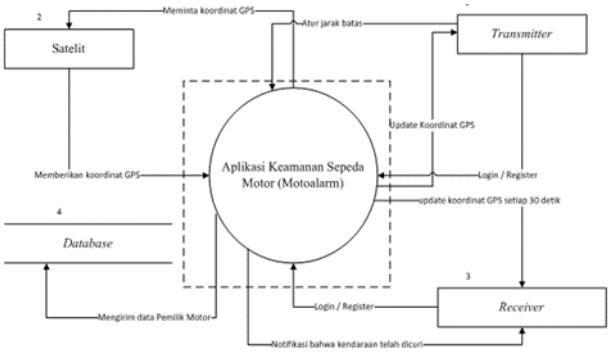
Gambar 1 Diagram blok Aplikasi *Motoalarm*

Pada blok sistem di atas digambarkan *transmitter* akan meminta kepada satelit koordinat dan lokasi di mana sepeda motor berada. Setelah bagian pertama disetujui, maka akan dikirimkan data lokasi ke *receiver* untuk menandai jarak awal sepeda motor berada.

Ketika radius ± 10 meter terlewati, *transmitter* akan mengirimkan notifikasi dan lokasi sepeda motor setiap 30 detik ke *receiver*. Tidak hanya ke *receiver*, aplikasi juga mengirimkan lokasi beserta identitas pemilik sepeda motor ke database.

Diagram Context Diagram (DCD)

Diagram Context Diagram ditunjukkan pada Gambar 2.

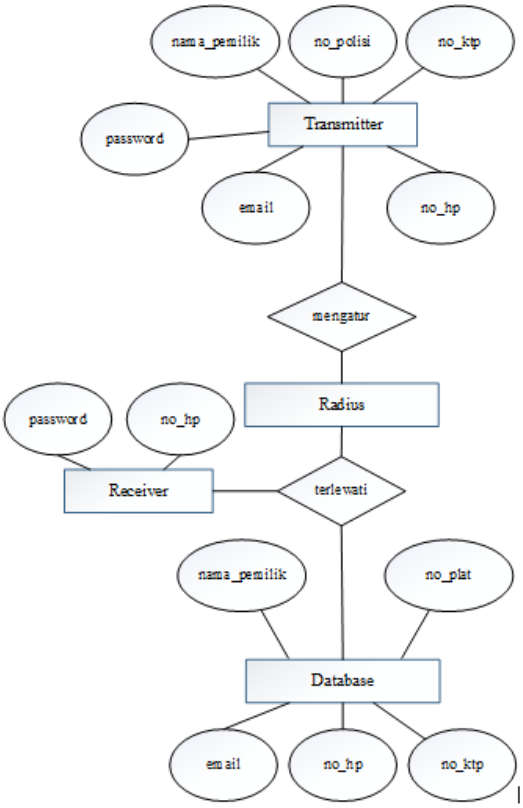


Gambar 2 Diagram Context Diagram aplikasi *Motoalarm*

Dijelaskan pada gambar 2 bahwa *transmitter* harus registrasi ke aplikasi sehingga dapat login dan menggunakan aplikasi. Ketika aplikasi mendeteksi bahwa radius terlewati, maka *transmitter* meminta kepada aplikasi koordinat dan lokasi *transmitter* berada. Setelah adanya permintaan dari *transmitter*, maka aplikasi meminta satelit untuk mengirimkan lokasi dari *transmitter* setiap 30 detik. *Receiver* sebelumnya juga harus register ke aplikasi dan *login* untuk menggunakan aplikasi ini. *Receiver* akan menerima pemberitahuan setiap 30 detik dari lokasi *transmitter* ketika dicuri. Pada *database* akan tercatat koordinat, lokasi, dan data pemilik sepeda motor.

Entity Relationship Diagram

Gambar 3 menerangkan tentang hubungan antar *entity.*



Gambar 3 Entity Relationship Diagram Aplikasi *Motoalarm*

Berdasarkan ERD pada gambar 3, dapat dijelaskan bahwa *transmitter* memiliki data : nomor ID (KTP/SIM), nomor plat sepeda motor, nama pemilik sepeda motor, nomor telepon, email, dan password. Kemudian *transmitter* mengatur jarak batas. Jika terlewati, akan mengirimkan notifikasi dan mengirim lokasi sepeda motor setiap 30 detik ke *receiver* dan *database*. *Database* akan mendapatkan data dari *transmitter* yaitu nomor ID (KTP/SIM), nomor plat sepeda motor, nama pemilik sepeda motor, nomor telepon, *email*, dan lokasi.

Skenario Pengujian

Dalam pengujian aplikasi, ada beberapa skenario yang akan dilakukan, yaitu :

1. Sepeda motor yang dijadikan obyek penelitian diasumsikan dipakai oleh pencuri ataupun akan dibawa di perjalanan (outdoor) sehingga tidak memperdulikan dimana posisi parkir (baik di *basement* atau lapangan terbuka/*outdoor*).

2. Seperti dalam batasan masalah pada Bab I, akses yang digunakan adalah *multiuser*, artinya aplikasi dapat dipakai oleh setiap orang jika memiliki aplikasi dan tahu *username* dan *password* yang digunakan.
3. Informasi yang dipakai hanya menggunakan akses paket data seluler.

3. HASIL PENGUJIAN DAN ANALISA

Ada beberapa pengujian yang dilakukan agar aplikasi dapat berjalan optimal, yaitu pengujian pada perangkat yang digunakan dan pengujian *software* itu sendiri.

Pengujian *Device* (Perangkat)

Dalam menjalankan aplikasi *Motoalarm* ini, ada beberapa hal yang perlu diperhatikan salah satunya adalah dari segi device atau smartphone yang digunakan.

Berikut beberapa smartphone dan hasil pengujiannya :

Tabel 1 Hasil Pengujian dengan Berbagai Perangkat

Smartphone	Versi *Android*	Ukuran Layar (inci)	Hasil Pengujian
Xiaomi Redmi 3s Pro	6.0.1 (*Marshmallow*)	5	• Peta tidak menunjukkan lokasi awal ketika *login* • Tidak bisa mengirimkan *update* lokasi transmitter • Keterangan dalam aplikasi tertampil
			• *Push notification*, nada dering, dan getaran tidak ada ketika melewati radiusnya pada mode *receiver*
Andromax C3 dan C3s	4.4.2 (*Kitkat*)	4.5	• Menunjukkan lokasi awal ketika *login* • Keterangan yang seharusnya tertampil tidak ada • *Push notification* dan nada dering tertampil, namun getaran tidak ketika pada mode *receiver*
Xiaomi Redmi 3 Pro	5.1 (*Lollipop*)	5	• Tampilan peta langsung tertuju ke tempat awal *login* • Keterangan tertampil sempurna dan *update* secara *real-time* • *Push notification*, nada dering, dan getaran berjalan ketika pada mode *receiver*
LG Optimus LII	4.1.2 (*Jelly Bean*)	3,5	• Peta tidak menunjukkan langsung lokasi awal ketika *login* • Keterangan tidak tertampil • *Push notification* tertampil, namun getaran dan nada dering tidak ketika pada mode *receiver*
Lenovo P7	5.1 (*Lollipop*)	5	• Tampilan peta langsung tertuju ke tempat awal *login* • Keterangan tertampil sempurna dan *update* secara *real-time* • *Push notification*, nada dering, dan getaran berjalan ketika pada mode *receiver*

Berdasarkan Tabel 1, dapat diketahui bahwa :

1. Pemakaian aplikasi pada smartphone dengan layar 5 inci dan sistem operasi Android 5.1 (Lollipop) bekerja maksimal. Dalam hal ini, yang dikatakan maksimal adalah tampilnya antarmuka peta dan menunjukkan posisi *transmitter* , keterangan dari posisi dan pemilik sepeda motor, dan adanya *push notification* dan getaran ketika *transmitter* melewati radius ± 10 meter (mode *receiver*).
2. Smartphone dengan layar di bawah 5 inci hanya dapat menampilkan antarmuka peta, *push notification* beserta getaran. Namun keterangan tidak tertampil.
3. Sehingga pada pengujian aplikasi, smartphone yang digunakan adalah Xiaomi Redmi 3 dan Andromax C3

Pengujian *Software*

Ada tiga pengujian yang dilakukan untuk menentukan apakah aplikasi *Motoalarm* ini berjalan dengan baik. Berikut pengujian-pengujiannya :

1. Pengujian Aplikasi Ketika melewati Radius

Ketika *transmitter* telah masuk (*login*), maka otomatis titik awal telah ditentukan. Sehingga jika bergerak lebih dari 5 meter, maka akan mengirimkan data ke receiver. Pengujian ini sebagai acuan yang digunakan bahwa sepeda motor telah berpindah lebih dari 10 meter apakah mengirim notifikasi atau tidak.

Akan dilakukan 10 pengujian untuk membuktikan keadaan tersebut. Berikut hasil pengujian yang didapat pada tabel 3.

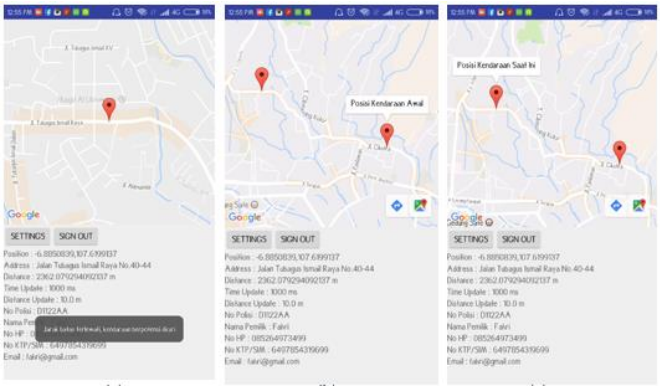
Tabel 2 Notifikasi jika Radius terlewati

No	Radius ± 10 meter (dalam meter)	Keterangan
1	9,1 – 10	Tidak ada *push notification*, nada dering, dan getar
2	10,1 – 11	ada *push notification*, nada dering, dan getar
3	11,1 – 12	ada *push notification*, nada dering, dan getar
4	12,1 – 13	ada *push notification*, nada dering, dan getar
5	13,1 – 14	ada *push notification*, nada dering, dan getar
6	14,1 – 15	ada *push notification*, nada dering, dan getar
7	15,1 – 16	ada *push notification*, nada dering, dan getar
8	16,1 – 17	ada *push notification*, nada dering, dan getar
9	17,1 – 18	ada *push notification*, nada dering dan getar
10	18,1 – 19	ada *push notification*, *nada dering*, dan getar

Dari Tabel 2 dapat disimpulkan bahwa radius kurang atau sama dengan 10 meter, maka tidak ada notifikasi, baik *push notification*, nada dering, maupun getaran. Sedangkan jika sepeda motor melewati radius lebih dari 10 meter, maka ada *push notification*, nada dering, dan getaran pada *receiver*.

2. Pengiriman Data dari *Transmitter* ke *Receiver*

Selanjutnya pada bagian ini, akan diuji bagaimana smartphone pada sepeda motor (*transmitter*) akan mengirimkan data lokasi ke *smartphone* pada pemilik (*receiver*) yang dapat dilihat pada gambar berikut.



Gambar 4 Tampilan pada *receiver*
(a) awal *login* (b) pin posisi awal (c) pin posisi akhir

Pada Gambar 4 dapat dilihat ketika melewati radius yang ditentukan, pada saat awal *login*, *receiver* akan memunculkan *push notification* serta langsung merujuk ke posisi sepeda motor terakhir. Sedangkan pada gambar 4 (b) dan (c), akan ditampilkan posisi awal dan akhir dari sepeda motor

3. Pengiriman Data dari *Transmitter* Ke *Database*

Selanjutnya pengujian akan dilakukan untuk pengiriman dari *transmitter* ke *database*. Di dalam *database* pada gambar 5, akan terisi data-data terkait koordinat dan lokasi. Seperti nomor polisi, koordinat awal dan akhir kendaraan, beserta jarak antara kedua koordinat.

Tabel 3 Data dari *transmitter* ke *database* komputer

no_polisi	lat_awal	lon_awal	lat_akhir	lon_akhir	jarak	lokasi
D1122AA [->]	-6.893733944811 06	107.639456576160 17	-6.8847829	107.6044697	3978.294823632689	Jalan Setiabudhi
D2233BB [->]	-6.8850839	107.6199137	-6.8843914	107.6041923	1737.242011759116	Jalan Setiabudhi No.8
D3344CC [->]	-6.8879188	107.6023428	-6.8843914	107.6041923	442.1872614860535	Jalan Setiabudhi No.8
D4455DD [->]	-6.8847829	107.6044697	-6.9299432	107.6162144	5186.257731167755	Jalan Karaputan No.121
D5566EE [->]	-6.8919604	107.6028513	-6.8626373	107.5873159	3684.115073414285 2	Jalan Geger Kalong Girang No.78C

Tabel 4 *Update* koordinat dan lokasi setiap 30 detik

no_polisi	lat_akhir	lon_akhir	lokasi_akhir	jarak	waktu
D1122AA	-6.8938598	107.639694	Jalan Cikutra No.205	0	2017-02-17 20:01:56
D1122AA	-6.8933259	107.6400754	Jalan Cikutra No.278	72.7812679577681	2017-02-17 20:02:18
D1122AA	-6.8934203	107.6397054	Jalan Cikutra No.278	48.88637088267668	2017-02-17 20:02:33
D1122AA	-6.8931721	107.6398904	Jalan Cikutra No.278	79.48287501480512	2017-02-17 20:02:53
D1122AA	-6.893242	107.6401678	Jalan Cikutra No.278	86.34121895347005	2017-02-17 20:03:13
D1122AA	-6.8938631	107.6397597	Jalan Cikutra No.278	7.261966030342112	2017-02-17 20:04:16
D1122AA	-6.8931616	107.6401678	Jalan Cikutra No.278	93.61105631886957	2017-02-17 20:04:53
D1122AA	-6.8938636	107.6397479	Jalan Cikutra No.278	5.965059815398571	2017-02-17 20:05:34
D1122AA	-6.893242	107.6401678	Jalan Cikutra No.278	86.34121895347005	2017-02-17 20:06:56
D1122AA	-6.8931686	107.6399829	Jalan Cikutra No.278	83.21202256243605	2017-02-17 20:07:14
D1122AA	-6.8932385	107.6402603	Jalan Cikutra No.278	93.17109312942085	2017-02-17 20:07:35

Pada tabel 4 dapat dilihat bagaimana data yang diperbarui koordinat, lokasi, dan jarak setiap 30 detik atau bahkan lebih cepat. Sedangkan pada tabel 5 yang telah diisikan identitas yang telah didaftarkan pada *form* register. Data yang terkirim pada database meliputi, *username* (nomor plat sepeda motor), *password* (yang sudah terenkripsi), nama (nama pemilik), hp (nomor telepon), ktp (nomor KTP), dan *email*.

Tabel 5 Data dari pemilik dalam database komputer

no_polisi	password	nama_pemilik	no_hp	no_ktp	email
D-4011-VAB	e10adc3949ba59abbe56e057f20f883e	Charlie	081537577720	9645820231092000	cha373@gmail.com
F 1234 SS	25f9e794323b453885f5181f1b624d0b	Uwi	08412365789	12345678	asal@inget.com
T-1234-AD	5f4dcc3b5aa765d61d8327deb882cf99	Abah	0854649127	951243784578	abah@gmail.com
T-1992-DA	5f4dcc3b5aa765d61d8327deb882cf99	Ahmad Kosasih	085697119571	321527061290	ahmad.kosasih@outlook.com
T-9999-AB	6bec9c852847242e384a4d5ac0962ba0	Udin	0845421818	12345678954	udin@gmail.com

Analisa Sistem

Berdasarkan pada pengujian yang dilakukan terdapat beberapa catatan yaitu :

1. Pengujian *form* registrasi berhasil menyimpan data-data yang diinputkan. Namun ketika menggunakan LG Optimus LII, penyimpanan data selalu tidak berhasil.
2. Pada *form login*, *smartphone* berhasil masuk ke aplikasi setelah mengisi nomor telepon dan *password* sesuai yang telah diregistrasikan sebelumnya. Selain itu, keduanya juga bisa login sebagai *transmitter* maupun *receiver*. Namun ketika menggunakan LG Optimus LII sebagai *receiver* dan *transmitter*, tidak berjalan dan hanya gambar peta saja yang muncul.
3. Pengujian dengan menggunakan *smartphone* yang dicoba bisa mengirimkan dan menerima notifikasi ketika melewati radius yang ditentukan.
4. Ketika pengiriman data ke *receiver*, bisa memunculkan *push notification* dan getaran dan berhasil mengirimkan keterangan yang terdapat pada *transmitter*.
5. Setelah beberapa kali pengujian, *update* lokasi sepeda motor ke *receiver* ketika

berpindah bisa lebih cepat dari 30 detik (seperti yang ditentukan di awal.

7. Ketika pengiriman data ke database, bisa memunculkan tempat dan lokasi, serta data dari pemilik sepeda motor.

4. SIMPULAN DAN SARAN

Hasil perancangan dan pengujian pada penelitian ini menunjukkan bahwa aplikasi dapat berjalan dengan baik. Aplikasi pada *transmitter* dapat mengirimkan *push notification*, getaran maupun data koordinat dan lokasi pada *receiver* ketika melewati radius ± 10 meter. Selain itu dapat mengirimkan koordinat dan data pemilik ke *database.*

Perangkat yang digunakan masih berupa *smartphone*, sehingga permasalahan ketahanan yaitu baterai serta akurasi yang masih menjadi kendala. Perlu dibuat dengan *hardware* yang mumpuni dan sedemikian rupa sehingga aplikasi dapat berjalan dengan lebih baik. Pengujian juga seharusnya dilakukan di berbagai tempat dan cuaca sehingga aplikasi dapat berjalan dengan maksimal atau tidak. Selain itu juga *database* juga dapat dibuat seperti halaman *web* sehingga lebih *user-friendly.*

5. DAFTAR PUSTAKA

[1] Anhar, 2010, *Panduan Menguasai PHP dan MySQL Secara Otodidak*. Penerbit : Mediakita. Jakarta.

[2] Mufti, Y., 2015, *Panduan Mudah Pengembangan Google Map Android*. Penerbit : C.V ANDI. Yogyakarta.

[3] Arya, Aristya, 2012, *Cara Kerja GPS*, diakses pada 22 Februari 2016, pada *World Wide Web*: http://radenseta.blogspot.com/2012/10/prinsip-kerja-gps-global-positioning.html

[4] Gandapurnama, Baban, 2016, *Ada 460 Kasus Pencurian Motor Selama Tahun 2016 di Bandung*, diakses Pada 18 Januari 2017, pada *World Wide Web*: https://news.detik.com/berita/d-3385135/ada-460-kasus-pencurian-motor-selama-tahun-2016-di-bandung

[5] Noname, 2014, Android (Sistem Operasi), diakses 26 Januari 2016, pada World Wide Web: http://id.wikipedia.org/wiki/Android_%28sistem_operasi%29

[6] Soegie, 2013, GPS vs A-GPS, apa sih bedanya?, diakses pada 24 Februari 2016, pada World Wide Web: http://www.kangsoegie.com

[7] Noname, 2014, Android Studio, diakses pada 20 Februari 2016, pada World Wide Web: http://wikipedia.org/wiki//Android_Studio