

Rancang Bangun Sistem Informasi Zakat Infaq Shodaqoh Menggunakan Metodologi *Extreme Programming*

Rysa Sahrial

Manajemen Informatika, Fakultas Teknik, Universitas Putra Indonesia
Jl. Dr Muwardi No. 66, Bypass Cianjur 46113
E-mail: rysasahrial@aliendro.id

Masuk: 5 April 2018; Direvisi: 3 Mei 2018; Diterima: 27 Juni 2018;

Abstract. *Poverty is one continuing social issue which is hard to solve. Dealing with this problem, Islam has already had the alternative solution that is tithe (Zakat). Zakat is implemented to decrease economy imbalanced appeared in the society. While in fact, not all the Moslem pay Zakat. There are five factors as the reason why Moslem didn't do that. First, some Muzakki wants to deliver his zakat directly. Second, not all Muzakki know how much Zakat must be paid. The other factors are Limited information about Mustahik home, limited time that Muzakki have to deliver his Zakat directly and the easiness to report Mustahik data. Dealing with those factors, it is required to have an information system which can make Muzakki meets Mustahik. In this research, information system application used Extreme Programming (XP) development method. XP method is required to program a system which will be made by accomodating the users' needs and expectations.*

Keywords: *zakat, information systems, extreme programming.*

Abstrak. Rancang Bangun Sistem Informasi Zakat Infaq Shodaqoh Menggunakan Metodologi *Extreme Programming*. Kemiskinan adalah masalah sosial yang sering ditemui dan masih belum teratasi. Dalam hal ini Islam mengenal alternatif pemecahannya, yakni zakat. Zakat diperuntukkan sebagai pengurang kesenjangan ekonomi yang terjadi di masyarakat. Akan tetapi tidak semua yang wajib mengeluarkan zakat menjalankannya. Penulis menemukan lima faktor yaitu pertama sebagian *muzakki* ingin menyalurkan zakatnya secara langsung. Kedua, tidak semua *muzakki* mengetahui berapa besaran zakat. Faktor lainnya adalah keterbatasan informasi tempat tinggal *mustahik*, keterbatasan waktu untuk meyalurkan zakat secara langsung, dan kemudahan melaporkan data *mustahik*. Untuk itu dibutuhkan sistem informasi yang bisa mempertemukan *muzakki* dan *mustahik*. Pada penelitian ini perancangan sistem informasi menggunakan metode pengembangan *extreme programming (XP)*. Metode XP dibutuhkan untuk merancang sistem yang akan dibuat dengan memperhatikan kebutuhan dan keinginan pengguna.

Kata kunci: *zakat, sistem informasi, extreme programming.*

1. Pendahuluan

Kemiskinan adalah masalah sosial yang sering ditemui baik di negara berkembang maupun negara maju yang mempunyai kemampuan di bidang pembangunan ekonomi. Pembangunan ekonomi diperlukan suatu bangsa untuk menyejahterakan rakyat melalui pemanfaatan sumber daya yang ada. Pembangunan bukanlah sebuah tujuan melainkan hanya alat untuk menurunkan kemiskinan dan mengurangi ketimpangan distribusi pendapatan. Jika pertumbuhan ekonomi tidak diikuti dengan pemerataan maka tidak ada manfaatnya untuk mengurangi kemiskinan.

Selaku warga negara Indonesia, khususnya umat islam berkewajiban mengisi pembangunan. Dalam hal ini islam telah mengenal alternatif pemecahannya, yakni zakat. Zakat merupakan salah satu instrumen islami yang digunakan untuk distribusi pendapatan dan kekayaan. Adanya zakat firaah, zakat maal dan zakat profesi diharapkan dapat menekan tingkat

ketimpangan kekayaan di Indonesia, selain itu juga zakat dapat diandalkan sebagai salah satu mekanisme dalam mengatasi masalah kemiskinan yang terjadi di Indonesia, melalui program zakat produktif (Pratama, 2015).

Pengumpulan zakat, infaq dan shodaqoh di Indonesia diatur oleh UU No 38 tahun 1999, pada tingkat nasional terdapat BAZNAS (Bada Amil Zakat Nasional). Selain membayarkan zakat ke pengelola, zakat juga bisa disalurkan sendiri secara langsung kepada penerima zakat. Akan tetapi kurangnya pemahaman mengenai zakat, seperti berapa ukuran zakat yang harus dikeluarkan, kapan zakat dibayarkan dan siapa saja orang yang berhak menerima zakat. Membuat sebagian pemeluk agama islam yang telah wajib membayar zakat tidak mengeluarkan zakatnya. Selaian itu kendala daftar penerima zakat di lingkungan kadang tidak tersedia. Saat ini, sistem informasi yang ada di BAZNAS belum terintegrasi dengan sistem informasi geografis yang bisa digunakan oleh masyarakat umum. Sehingga masyarakat tidak bisa melihat pemetaan penerima zakat di lingkungan sekitar. Oleh karena itu pemetaan penerima zakat berbasis teknologi sangatlah diperlukan.

Saat ini perkembangan teknologi yang termasuk cepat adalah perangkat *mobile* dengan sistem operasi android. Salah satu keunggulan dari sistem operasi android adalah *open source* dengan fasilitas yang banyak bagi para pengembang aplikasi android. Salah satunya adalah fasilitas Google Maps. Informasi mengenai pemetaan zakat akan lebih mudah diakses oleh masyarakat dengan adanya sistem informasi geografis berbasis android menggunakan fasilitas Goolge Maps.

Pengembangan sistem informasi biasanya diawali dengan proses analisis sehingga dapat ditentukan kebutuhan-kebutuhan dalam pengembangan sistem. Akan tetapi proses pengumpulan kebutuhan sistem tidak selalu ditemukan pada proses analisis. Seringkali pengembangan sistem ditemukan masalah setelah dilakukan proses analisis. Oleh karena itu diperlukan metode yang bersifat terbuka terhadap perubahan kebutuhan yang lebih cepat, dari berbagai metode pengembangan perangkat lunak, metode yang dapat mengakomodasi perubahan kebutuhan yang begitu cepat adalah metode pengembangan *extreme programming*.

Extreme programming merupakan metode pengembangan perangkat lunak yang lebih memperhatikan perubahan cepat dan sederhana dengan melakukan iterasi sesuai dengan keinginan pengguna. Dengan menggunakan metode *extreme programming* penulis melakukan penelitian, perancangan dan evaluasi sistem yang akan mempertemukan *muzakki* dan *mustahik* menggunakan fasilitas Google Maps.

2. Landasan Teori

2.1. Proses Agile

Proses Agile adalah pengembangan perangkat lunak berbasis *iterative* dan *incremental*, dimana persyaratan dapat diubah sesuai dengan kebutuhan pelanggan. Sehingga membantu dalam perencanaan adaptif, pengembangan berulang, dan ketepatan waktu (Sharma, dkk., 2012). Kegiatan pengembangan perangkat lunak seperti *planning*, *analysis*, *design*, *coding*, *testing* dan *maintenance* yang perlu dilakukan sesuai dengan permintaan pelanggan (Ahmed, dkk., 2010).

2.2. Model Extreme Programming

Extreme programming adalah pendekatan iteratif untuk pengembangan perangkat lunak, yang dikenal sebagai metode agile. *Extreme programming* paling banyak digunakan karena sederhana, cepat dan mudah digunakan. Kent Beck awalnya memformulasikan pendekatan *extreme programming* yang terdiri dari nilai, keyakinan utama dan praktek (Krishna, dkk., 2011).

Extreme programming membutuhkan interaksi pelanggan secara maksimal untuk mengembangkan perangkat lunak. Dengan membagi seluruh siklus hidup pengembangan perangkat lunak menjadi beberapa jumlah siklus pengembangan yang singkat. Tujuannya untuk menerima dan menggabungkan perubahan atau persyaratan dari pelanggan pada setiap tahapan dari siklus hidup pengembangan (Sharma, dkk., 2012). Praktek *extreme programming* ditunjukkan pada tabel 1.

Tabel 1. Praktek Extreme Programming (Khrisna, dkk., 2011).

No	Praktek	Keterangan
1	<i>Planning Game</i>	Perancangan untuk iterasi yang akan datang, penggunaan cerita yang diberikan pelanggan. Para teknisi menentukan jadwal, perkiraan biaya, dll. Sebagai kolaborasi antara pelanggan dan pengembang.
2	<i>Small Releases</i>	Memecah rilis aplikasi menjadi beberapa bagian kecil untuk menghasilkan perangkat lunak yang lebih cepat. Rilis lebih kecil memberikan bisa mengurangi risiko lebih awal.
3	<i>Metaphor</i>	Konsep penamaan kelas dan metode untuk memudahkan anggota tim (pelanggan, programmer dan manajer) mengetahui semua fungsi dan cara kerja sistem.
4	<i>Simple Design</i>	Penggunaan <i>refactoring</i> untuk membuat kode kompleks yang lebih sederhana untuk mendapatkan fungsi yang sama.
5	<i>Testing</i>	Pengujian diperlukan untuk mengurangi resiko kegagalan pada perubahan kode sumber.
6	<i>Refactoring</i>	Peningkatan desain sistem diseluruh pengembangan. Pola desain adalah bagian mendasar dari <i>refactoring</i> , selain menjaga perangkat lunak tetap bersih dengan melakukan tanpa duplikasi kode, menjaga komunikasi, sederhana dan lengkap.
7	<i>Pair Programming</i>	Kolaborasi <i>programmer</i> untuk membuat desain, algoritma, kode dan pengujian sistem yang sama.
8	<i>Collective Ownership</i>	Semua kode dimiliki setiap anggota tim, sehingga semua orang dapat melakukan perubahan kode kapan saja, dan mendorong untuk menyumbangkan ide-ide baru.
9	<i>Continuous Integration</i>	Tim pengembangan harus selalu bekerja pada versi terbaru dari perangkat lunak. Karena anggota tim yang berbeda mungkin memiliki versi yang disimpan secara lokal dengan berbagai perubahan dan peningkatan. Integrasi berkelanjutan akan menghindari penundaan di kemudian hari dalam siklus proyek, yang disebabkan oleh masalah integrasi.
10	<i>40-Hour Week</i>	Tim memutuskan bekerja dengan kecepatan, yang nyaman dengan semua anggota tim dan menganjurkan tidak bekerja terlalu keras.
11	<i>On-site Customer</i>	Pelanggan duduk bersama dengan pengembang untuk mengeksplorasi kebutuhan bisnis, membuat keputusan penting, sehingga mengurangi risiko proyek.
12	<i>Coding Standards</i>	Semua <i>programmer</i> harus menulis kode dengan cara yang sama, dengan aturan untuk memastikan bahwa kode tersebut berkomunikasi dengan jelas.

2.3. Sistem Informasi Geografis

Sistem Informasi Geografis (GIS) adalah sistem terkomputerisasi untuk input, penyimpanan, manajemen, tampilan dan analisis data yang dapat secara tepat terkait dengan lokasi geografis. *Dataset* GIS berupa lapisan, seperti lapisan sungai, jalan dan kode pos pada geografis tertentu. Lapisan terdiri dari satu atau banyak fitur, yang meliputi titik, garis, atau batas. Berbagai lapisan ditumpangkan untuk membuat peta berarti yang memiliki tampilan peta dan data. Peta sebagai data secara visual yang dapat menampilkan atribut pada *dataset*. Tampilan data dapat digunakan untuk membuat kumpulan data yang lebih kecil dari kumpulan data besar yang menggunakan *query tool* (Rob, 2003).

2.4. Android

Android adalah sebuah *software stack* bersifat *open source* yang mencakup sistem operasi, *middleware*, dan *key applications* beserta sekumpulan *Application Programming Interface* untuk merancang aplikasi *mobile* (Meier, 2009). Pada awalnya pembuatan aplikasi android menggunakan bahasa pemrograman Java. Tapi dengan perkembangan bahasa pemrograman, menjadikan pembuatan aplikasi android bisa menggunakan berbagai bahasa pemrograman seperti Kotlin, C#, dan HTML5 dengan format aplikasi *native*.

2.5. End User Computing Satisfaction

End User Computing Satisfaction (EUCS) digunakan untuk mengukur pengalaman penggunaan sistem informasi, apakah sistem informasi sudah efektif dan sesuai dengan yang diinginkan. Hal ini berkaitan dengan kualitas informasi yang diterima oleh pengguna atas sistem informasi. EUCS pertama kali diperkenalkan oleh Doll dan Torkzadeh's (1988) dengan skala likert menggunakan 12 indikator (Suzanto & Sidharta, 2015). Dan yang diukur oleh Doll & Torkzadeh untuk dapat menentukan kepuasan pengguna dibagi menjadi lima kategori (Ariaji, dkk., 2014). EUCS ditunjukkan pada tabel 2.

Tabel 2. End User Computing Satisfaction (di adopsi dari Ariaji, dkk., 2014)

No	Kategori	Indikator
1	Content	Aplikasi sistem memberikan informasi yang tepat sesuai dengan yang anda butuhkan?
		Apakah isi informasi yang dihasilkan sistem memenuhi kebutuhan anda?
		Apakah sistem memberikan laporan sesuai dengan yang anda butuhkan?
		Apakah sistem memberikan informasi yang cukup sesuai dengan yang anda butuhkan?
2	Accuracy	Apakah sistem ini akurat?
		Apakah anda puas dengan akurasi sistem?
3	Format	Apakah menurut anda hasil disajikan dalam format yang berguna?
		Apakah informasi yang dihasilkan sistem jelas?
4	Ease of Use	Apakah sistem ini ramah pengguna?
		Apakah sistem mudah digunakan?
5	Timeless	Apakah anda menerima informasi yang anda butuhkan tepat waktu?
		Apakah sistem menyediakan informasi terbaru?

2.6. Penelitian Sebelumnya

Tabel 3 merupakan penelitian sebelumnya yang berkaitan dengan penelitian sistem informasi zakat, infaq, shodaqoh sebagai berikut:

Tabel 3. Penelitian Sebelumnya

No	Nama Penelitian dan Tahun	Judul Penelitian	Metode Penelitian	Hasil Penelitian
1	Amiq Fahmi, Edi Sugiarto Tahun 2015	Aplikasi Sistem Informasi Geografis Manajemen Aset Wakaf	Deskriptif Kualitatif, Waterfall Approach	Hasil penelitian ini diharapkan dapat membantu memperbaiki administrasi dan manajemen aset wakaf dalam jangka panjang.
2	Muhammad Zubair Asghar, Ulfat Batool, Farheen Bibi, Sadia Ismail, Syeda Rabail Zahra	Financial Studio: Android Based Application for Computing Tax, Pension, Zakat and Loan	comprised of four modules, namely: (i) tax calculation, (ii) pension calculation, (iii) zakat calculation, and (iv) tax calculation	Hasil penelitian ini merancang penghitungan zakat, pajak, pension, dan pinjaman berbasis sistem operasi android.

Adapun perbedaan penelitian ini dengan penelitian lain adalah rancangan sistem informasi yang dibuat menggunakan model *extreme programming*, hasil penelitian ini berupa aplikasi sistem informasi zakat, infaq dan shodaqoh berbasis android yang menggunakan sistem informasi geografis dengan Google Maps API serta ditambah penghitungan zakat.

3. Metodologi Penelitian

3.1. Analisis User Story

Dalam penelitian ini penulis melakukan penelitian di Kabupaten Cianjur. Menurut Badan Pusat Statistik (BPS) Pada tahun 2015 Kabupaten Cianjur memiliki jumlah penduduk 2.243.904 orang dengan jumlah penduduk miskin 273.900 orang.

Untuk mendapatkan analisis yang akurat, penulis memilih narasumber dari perwakilan BAZNAS Kabupaten Cianjur, *mustahik*, *muzakki* dan relawan. Hal ini dilakukan untuk memperoleh data dan informasi mengenai zakat, infak dan shodaqoh (ZIS). Dari hasil wawancara pada tabel 4 kemudian dianalisis dan diidentifikasi pokok permasalahan, pengambilan keputusan dan pemecahan masalah. Tabel 5 mengidentifikasi berbagai kebutuhan sistem dan dilanjutkan dengan perancangan, dan implementasi sistem.

Tabel 4. Data Permasalahan Hasil Wawancara

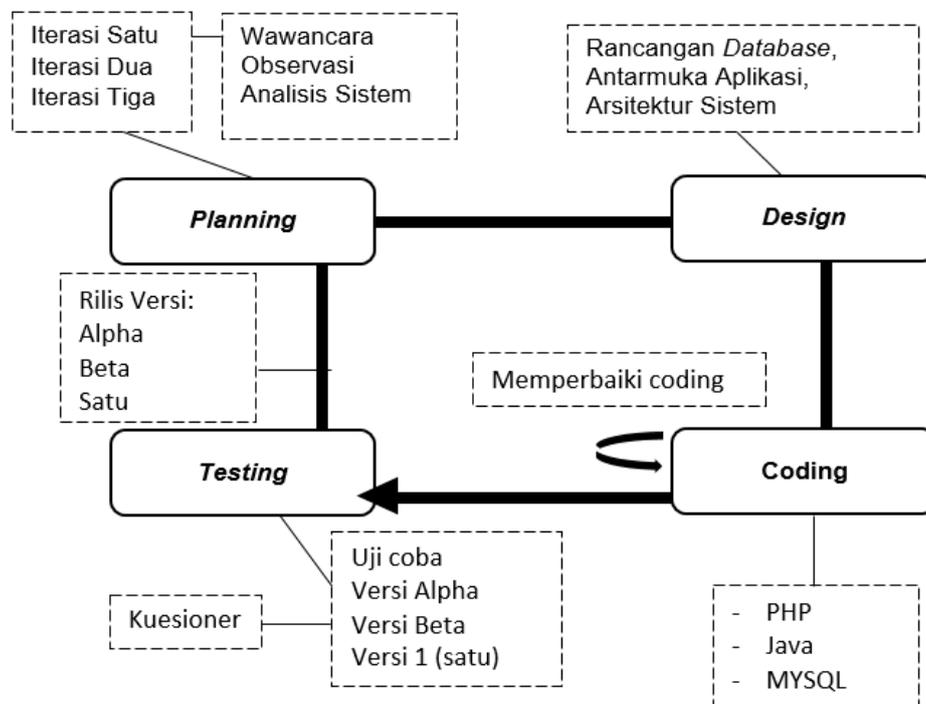
No	Hasil Analisis
1	Sebagian <i>muzakki</i> (orang yang wajib mengeluarkan zakat) ingin menyalurkan zakatnya secara langsung kepada <i>mustahik</i> (orang yang berhak mendapatkan zakat) tanpa melalui Baznas atau yayasan amil zakat.
2	Tidak semua <i>muzakki</i> mengetahui berapa besaran zakat yang harus dikeluarkan dan siapa saja yang berhak menerima zakat.
3	Keterbatasan informasi tempat tinggal <i>mustahik</i> , yatim piatu, anak asuh, yayasan duafa, masjid yang sedang direnovasi, dll.
4	Keterbatasan waktu <i>muzakki</i> yang ingin menyalurkan zakatnya secara langsung.
5	Kemudahan melaporkan data <i>mustahik</i> , yatim piatu, anak asuh, yayasan duafa, masjid yang sedang direnovasi, dll.

Tabel 5. Gambaran Umum Sistem Informasi Zakat, Infaq, Shodaqoh

No	Fitur Sistem Informasi
1	Sistem yang akan dibuat bisa mempertemukan <i>muzakki</i> dengan <i>mustahik</i> , yatim piatu, anak asuh, yayasan duafa, masjid yang sedang direnovasi, dll. Dengan konsep sistem informasi geografis menggunakan Google Maps API.
2	Sistem dilengkapi dengan kalkulator zakat untuk mempermudah <i>muzakki</i> melakukan penghitungan zakat secara benar.
3	Fasilitas rekening bersama untuk <i>muzakki</i> yang tidak mempunyai waktu dan faktor jarak menyalurkan zakat secara langsung.
4	Halaman pendataan untuk para relawan melaporkan data <i>mustahik</i> , yatim piatu, anak asuh, yayasan duafa, masjid yang sedang direnovasi, dll.
5	Fasilitas penilaian dan ulasan <i>mustahik</i> untuk para <i>muzakki</i> , agar data yang di berikan oleh relawan akurat dan tidak fiktif.

3.2. Perancangan Sistem

Dengan menggunakan 12 praktek *extreme programing* untuk melakukan evaluasi terhadap penerapan metode *extreme programming* dalam perancangan sistem informasi ZIS. Pada gambar 1 penulis membagi menjadi tiga iterasi dengan menghasilkan versi rilis yang ditunjukkan pada tabel 6. Setiap iterasi akan melalui tiga tahapan yaitu analisis sistem, desain sistem, *coding* dan pengujian versi.



Gambar 1. Alur Proses Extreme Programming Perancangan Sistem Informasi ZIS

Tabel 6. Planning Game Perancangan Sistem Informasi ZIS Dengan Tiga Iterasi

No	Versi	Keterangan
1	Iterasi Satu	Prose perancangan sistem diawali dengan menganalisa kebutuhan sistem berdasarkan <i>user story</i> , merancang desain sistem (<i>database</i> , arsitektur dan antarmuka). Selanjutnya proses <i>coding</i> sistem website untuk admin serta aplikasi android untuk <i>muzakki</i> dan relawan. Hasil iterasi satu berupa rilis versi <i>alpha</i> yang diujicoba oleh penulis. Tujuan dari pengujian versi <i>alpha</i> untuk melihat rancangan awal sistem sesuai dengan <i>user story</i> .
2	Iterasi Dua	Pengembangan sistem dilanjutkan pada tahap dua berdasarkan hasil ujicoba oleh penulis. Hasil sistem iterasi dua ini penulis akan meminta pengguna (<i>muzaki</i> dan relawan) untuk menguji sistem apakah sudah berjalan dengan benar dan sesuai dengan kebutuhan pengguna. Selain menguji sistem, pengguna juga memberikan umpan balik dengan mengisi kuesioner. Dengan metode EUCS yang mengukur tingkat kepuasan pengguna akhir terhadap suatu sistem informasi berdasarkan beberapa variabel yang sudah ditentukan. Hasilnya digunakan untuk mengukur tingkat keberhasilan sistem informasi.

3	Iterasi Tiga	Tahap ini merupakan proses pengembangan sistem yang sudah mencapai akhir. Dengan dua kali pengujian dan hasil dari kuesioner menghasilkan versi rilis satu. Dan perancangan aplikasi sudah sesuai dengan hasil <i>user story</i> dan umpan balik pengguna.
---	--------------	--

3.3. Spesifikasi Kebutuhan Sistem

Tabel 7 merupakan perancangan antarmuka pengguna yang dibuat pada sistem informasi ZIS ini menggunakan dua antarmuka yang berbeda. *Website* untuk Admin dan aplikasi Android untuk *muzakki* dan relawan.

Tabel 7. Spesifikasi Kebutuhan Sistem

No	Antar Muka	Keterangan
1	<i>Website</i> (Admin)	Perancangan <i>website</i> dibuat menggunakan bahasa pemrograman PHP dan komponen <i>website</i> dengan CSS, prosesnya sebagai berikut: a. Codeigniter digunakan untuk mempermudah perancangan dan pembuatan <i>website</i> secara dinamis. b. Tampilan <i>website</i> menggunakan Bootstrap supaya tampilan <i>website</i> lebih menarik dan responsif sehingga bisa digunakan diberbagai ukuran layar. c. PHP juga digunakan sebagai koneksi data dengan database MYSQL. d. JSON digunakan untuk melakukan pertukaran data meski bahasa pemrograman yang digunakan berbeda yaitu PHP dan Java. e. Notepad++ digunakan untuk mengedit bahasa pemrograman PHP, JSON dan mengatur beberapa komponen CSS. f. Data peta yang ditampilkan menggunakan fasilitas Google Maps API. g. Pesan yang dikirimkan oleh admin menggunakan fasilitas Google <i>Cloud Messaging</i> . h. Pengujian pada lokal menggunakan aplikasi XAMPP sebagai <i>server</i> dan Mozilla sebagai penelusur <i>web (browser)</i> .
2	Android Muzakki dan Relawan	Perancangan aplikasi android dibuat menggunakan bahasa pemrograman Java, prosesnya sebagai berikut: a. Android Studio (AS) digunakan sebagai perancangan aplikasi android <i>muzakki</i> dan <i>mustahik</i> dengan bahasa pemrograman Java. Versi AS yang digunakan adalah 2.3.3, Gradle 3.5 dan JDK 1.8.0. b. Agar aplikasi bisa digunakan diberbagai <i>smartphone</i> dan bisa diunggah atau dipasarkan di <i>market</i> Google Playstore, maka aplikasi harus dirancang menggunakan Android SDK 9. Sesuai dengan minimal SDK yang diperbolehkan oleh Google. c. Untuk menggunakan fasilitas Google Maps API maka aplikasi harus menggunakan API <i>play-services-maps:9.8.0</i> , dan API <i>play-services-gcm:9.8.0</i> untuk menerima notifikasi pesan yang dikirimkan oleh admin. Pada penulisan penelitian ini, API yang telah disediakan Google sudah mencapai versi <i>play-services:11.0.4</i> . Tetapi versi API terbaru hanya bisa digunakan minimal Android SDK 14. Yang artinya akan mengurangi jumlah <i>smartphone</i> yang bisa dipasang aplikasi sistem informasi ZIS. d. Pengujian lokal aplikasi android <i>muzakki</i> dan relawan menggunakan <i>emulator</i> Android dan <i>smartphone</i> dengan sistem operasi Android v5.1 Lollipop. Pengujian akan berhasil jika <i>smartphone</i> sebagai <i>client</i> dan komputer sebagai <i>server</i> saling terhubung menggunakan Wi-Fi.

4. Hasil Dan Pembahasan

Tabel 8 merupakan rancangan sistem pada iterasi satu yang menghasilkan sistem versi *alpha* sesuai dengan kebutuhan pengguna. Rancangan yang dibuat dalam penulisan ini hanya pada rancangan iterasi satu.

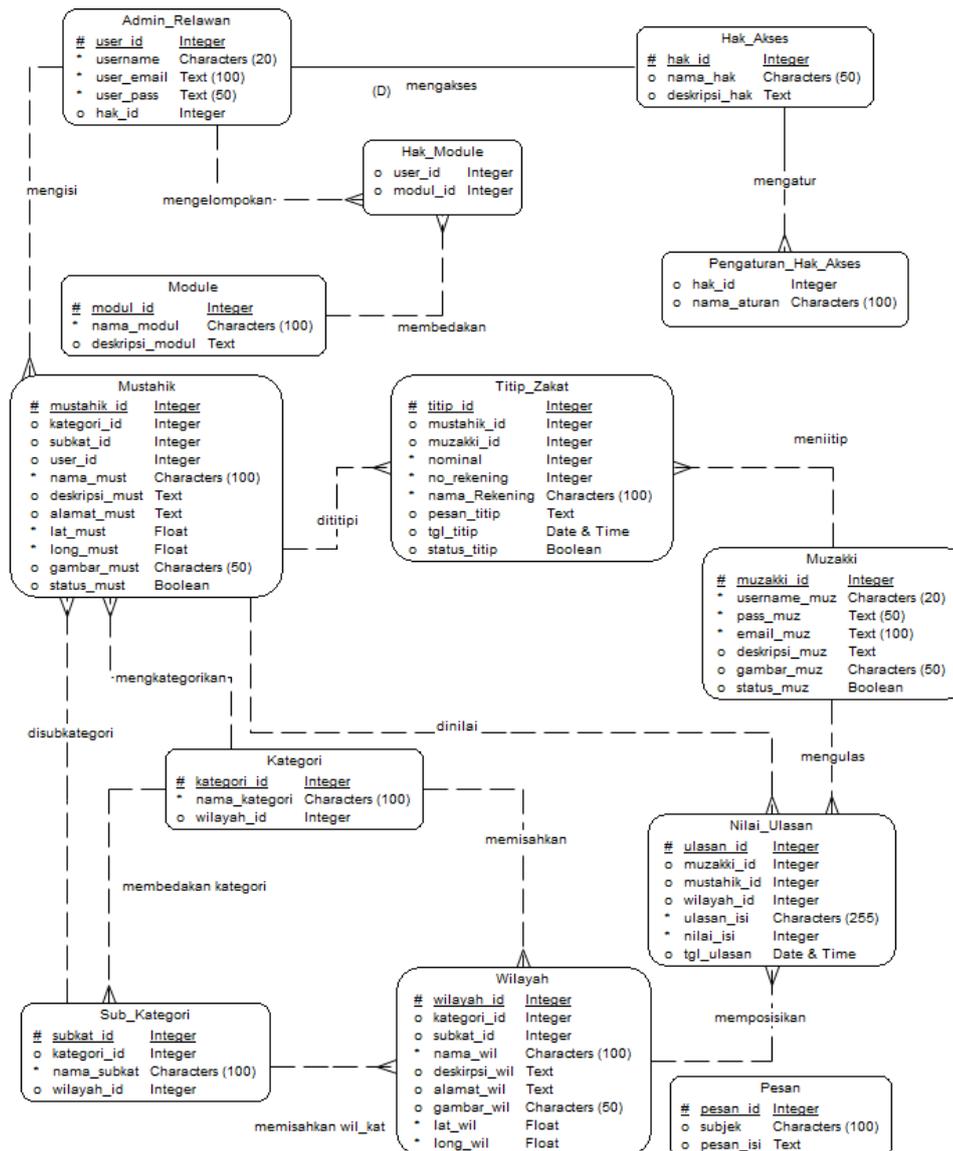
Tabel 8. Planning Iterasi Satu.

No	Use Case	Keterangan
1	Seleksi Relawan	Admin dapat menyeleksi dan mendaftarkan para relawan, karena relawan tidak bisa mendaftar menggunakan aplikasi. Tetapi harus mengirimkan berkas tambahan (KTP, foto, surat pernyataan kesediaan menjadi relawan dan mengikuti semua aturan) selain biodata. Setelah admin menyetujui pendaftaran relawan, admin mengirim <i>username</i> , <i>password</i> dan aplikasi android khusus relawan ke <i>email</i> yang didaftarkan. Tujuan dari penyeleksian ini untuk meminimalisir penggunaan sistem zakat oleh pihak yang tidak bertanggung jawab. Seperti mengirim data palsu/fiktif <i>mustahik</i> .
2	Kelola Mustahik	a. Admin dapat mengelola (menambah, mengubah, menghapus) data <i>mustahik</i> yang didaftarkan baik oleh admin maupun relawan. b. Relawan hanya dapat mengelola (menambah, mengubah, menghapus) data <i>mustahik</i> yang didaftarkan relawan. c. <i>Muzakki</i> bisa memilih <i>mustahik</i> berdasarkan kategori. Setelah memilih <i>mustahik</i> , <i>muzakki</i> bisa membaca biodata, memberi ulasan dan penilaian, mengirim zakat menggunakan

No	Use Case	Keterangan
		rekening bersama, dan melacak posisi keberadaan <i>mustahik</i> pada peta untuk melakukan pembayaran zakat secara langsung.
3	Kelola Ulasan dan Penilaian Mustahik	Admin dapat mengelola semua ulasan dan penilaian <i>mustahik</i> yang diberikan oleh <i>muzakki</i> .
4	Kelola Kategori dan Sub Kategori Mustahik	a. Admin hanya bisa mengubah status <i>muzakki</i> (<i>banned/ publish</i>). b. <i>Muzakki</i> bisa mengubah photo profil, <i>password</i> dan deskripsi.
5	Kelola Zakat	Admin dapat mengelola semua zakat yang dikirim oleh <i>muzakki</i> .
6	Kelola Pesan	a. <i>Muzakki</i> bisa melihat pesan yang dikirimkan oleh admin. b. Admin dapat mengirim pesan ke semua <i>muzakki</i> . Pesan berupa notifikasi yang berisi berita, info dan laporan seputar zakat.
7	Kelola Wilayah	Admin dapat menambah, mengubah dan menghapus wilayah penerima zakat.
8	Pendaftaran Muzakki	Tamu bisa melakukan pendaftaran sebagai <i>muzakki</i> . Untuk mendaftar tamu diwajibkan mengisi <i>username</i> , <i>password</i> dan <i>email</i> . Setelah berhasil mendaftar tamu bisa langsung <i>login</i> ke dalam sistem karena menggunakan <i>approval</i> secara otomatis.
9	Kelola Zakat	Admin dapat mengelola semua zakat yang dikirim oleh <i>muzakki</i> .
10	Hitung Zakat	<i>Muzakki</i> dan tamu bisa menggunakan menu kalkulator zakat tanpa harus <i>login</i> . Pada menu kalkulator zakat tersedia penghitungan zakat profesi, perdagangan, peternakan, pertanian, harta simpanan, harta temuan dan pengetahuan zakat.

4.1. Relasi Antar Tabel

Relasi Antar Tabel dibuat berdasarkan kebutuhan sistem. Bentuk relasi antar tabel yang digunakan ditunjukkan pada gambar 2.

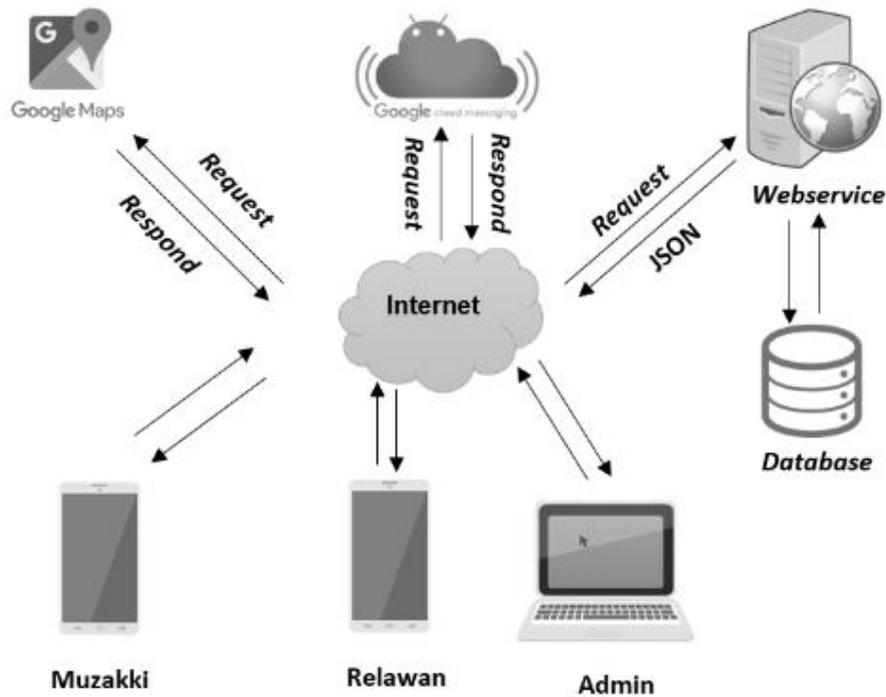


Gambar 2. Relasi Antar Tabel Sistem Informasi ZIS

4.2. Arsitektur Sistem Informasi ZIS

Berikut merupakan arsitektur sistem dari sistem informasi ZIS yang akan dirancang. Arsitektur ini menggambarkan tentang perangkat apa saja yang berhubungan dengan sistem.

Database yang ada di sistem informasi dapat diakses melalui beberapa *device* seperti *smartphone*, *laptop*, dan *personal computer*. Terlihat pada gambar 3 sistem informasi ZIS sangat bergantung pada koneksi *internet* untuk menghubungkan dan menerima data, baik itu mengambil dan mengirim data ke *database*, maupun dibutuhkan untuk menggunakan fasilitas *Google Maps API* dan *Google Cloud Messaging*. Sedangkan *JSON* digunakan untuk melakukan pertukaran data dengan bahasa pemrograman berbeda.



Gambar 3. Arsitektur Sistem Informasi ZIS

4.3. Pengujian Sistem

4.3.1 Pengujian Versi Alpha

Pengujian *alpha* adalah pengujian sistem yang dilakukan secara langsung oleh penulis dengan beberapa pengujian yang ditunjukkan pada tabel 9.

Tabel. 9 Proses Pengujian Versi Alpha

No	Pengujian
1	Cek koneksi antara <i>database</i> dengan <i>website</i> , aplikasi <i>muzakki</i> dan <i>relawan</i> .
2	Melihat tampilan <i>website</i> pada berbagai ukuran layar, apakah sudah <i>responsive</i> mengikuti ukuran layar.
3	Uji coba data, yaitu dengan memasukkan data yang benar dan data yang salah.

4.3.2 Pengujian Versi Beta

Pengujian *beta* adalah pengujian sistem yang dilakukan secara objektif yaitu pengujian langsung oleh *muzakki* dan *relawan* dengan bentuk pertanyaan EUCS. Untuk mengetahui pendapat responden terhadap sistem yang dibangun. Pengujian sistem dilakukan oleh 25 responden dari *muzakki* dan lima responden *relawan*, kemudian pengujian tersebut akan dituliskan pada kuesioner yang telah diberikan. Hasil pengujian ditunjukkan pada tabel 10.

Tabel 10. Pengujian Versi Beta Relawan dengan Bentuk Pertanyaan *Definisi End User Computing Satisfaction (EUCS)*

No	Variabel	Pertanyaan	STP	TP	N	P	SP
1	Content	Aplikasi sistem memberikan informasi zakat (pengetahuan zakat, kalkulator zakat, daftar dan peta <i>mustahik</i>) yang tepat sesuai dengan yang anda butuhkan oleh para <i>muzakki/relawan</i> ?			3	21	6
2		Apakah isi informasi yang dihasilkan sistem memenuhi kebutuhan anda sebagai <i>muzakki/relawan</i> ?			3	21	6
3		Apakah sistem memberikan laporan zakat sesuai dengan yang anda butuhkan?			6	22	2
4		Apakah sistem memberikan informasi <i>mustahik</i> yang cukup sesuai dengan yang anda butuhkan sebagai <i>muzakki</i> atau <i>relawan</i> ?			4	24	2

No	Variabel	Pertanyaan	STP	TP	N	P	SP
5		Apakah sistem ini memberikan lokasi <i>mustahik</i> dengan akurat pada halaman Google maps?		2	19	9	
6	Accuracy	Apakah anda puas dengan akurasi sistem menghitung zakat?		1	11	18	
7		Apakah menurut anda hasil data <i>mustahik</i> disajikan dalam format yang berguna pada peta Google maps?				12	18
8	Format	Apakah informasi yang dihasilkan sistem dengan format data spasial (data pada Google maps) jelas untuk anda?				13	17
9		Apakah menu dan fasilitas sistem ini sangat mudah digunakan?			1	14	15
10	Ease of Use	Apakah sistem bisa mudah dipasang di <i>smartphone</i> anda?				14	16
11		Apakah anda menerima informasi zakat dan <i>mustahik</i> yang anda butuhkan tepat waktu?		1	5	24	
12	Timeless	Apakah sistem menyediakan informasi zakat terbaru		1	14	15	
Total			0	5	65	207	82
			360				

Keterangan: SP: Sangat Puas (nilai 5), P: Puas (nilai 4), N: Netral (nilai 3), TP: Tidak Puas (nilai 2), STP: Sangat Tidak Puas (nilai 1).

Tabel 11 merupakan hasil dari kuesioner diukur dengan penilaian skala likert. Skala likert digunakan untuk mengukur sikap, pendapat dan persepsi seseorang atau sekelompok orang tentang fenomena sosial (Sugiyono, 2010). Kecenderungan jawaban responden ini dapat dilihat dari bentuk statistik deskriptif dari masing-masing variabel. Untuk mendapatkan total skor tiap pertanyaan menggunakan rumus: T (Jumlah responden yang memilih) \times P_n (Pilihan skor likert). Maka didapatkan total skor kepuasan tiap variabel. Dari hasil skor kepuasan akan didapatkan rerata kepuasan dengan menggunakan rumus $\text{index \%} = \frac{\text{total skor}}{Y} \times 100$, dimana Y adalah nilai tertinggi likert \times jumlah responden. $Y = 5 \times 30 = 150$.

Tabel 11. Nilai Skor Dan Rerata Kepuasan EUCS

No	Pertanyaan	Total Skor	Rerata
		$T \times P_n$	$\frac{\text{Total Skor}}{Y} \times 100$
1	Aplikasi sistem memberikan informasi zakat (pengetahuan zakat, kalkulator zakat, daftar dan peta <i>mustahik</i>) yang tepat sesuai dengan yang anda butuhkan oleh para <i>muzakki</i> /relawan?	N: $3 \times 3 = 9$ S: $21 \times 4 = 84$ SP: $6 \times 5 = 30$ Total skor 123	$123/150 \times 100 = 82\%$
2	Apakah isi informasi yang dihasilkan sistem memenuhi kebutuhan anda sebagai <i>muzakki</i> /relawan	N: $3 \times 3 = 9$ S: $21 \times 4 = 84$ SP: $6 \times 5 = 30$ Total skor 123	$123/150 \times 100 = 82\%$
3	Apakah sistem memberikan laporan zakat sesuai dengan yang anda butuhkan?	N: $6 \times 3 = 18$ S: $22 \times 4 = 88$ SP: $2 \times 5 = 10$ Total skor 116	$116/150 \times 100 = 77.3\%$
4	Apakah sistem memberikan informasi <i>mustahik</i> yang cukup sesuai dengan yang anda butuhkan sebagai <i>muzakki</i> atau relawan?	N: $4 \times 3 = 12$ S: $24 \times 4 = 96$ SP: $2 \times 5 = 10$ Total skor 118	$118/150 \times 100 = 77.7\%$
5	Apakah sistem ini memberikan lokasi <i>mustahik</i> dengan akurat pada halaman Google maps?	TP: $2 \times 2 = 4$ N: $19 \times 3 = 57$ S: $9 \times 4 = 36$ Total skor 97	$97/150 \times 100 = 64.6\%$
6	Apakah anda puas dengan akurasi sistem menghitung zakat?	TP: $1 \times 2 = 2$ N: $11 \times 3 = 33$ S: $12 \times 4 = 48$ Total skor 83	$83/150 \times 100 = 55.3\%$
7	Apakah menurut anda hasil data <i>mustahik</i> disajikan dalam format yang berguna pada peta Google maps?	S: $12 \times 4 = 48$ SP: $18 \times 5 = 90$ Total skor 138	$138/150 \times 100 = 92\%$
8	Apakah informasi yang dihasilkan sistem dengan format data spasial (data pada Google maps) jelas untuk anda?	S: $13 \times 4 = 52$ SP: $17 \times 5 = 85$ Total skor 137	$137/150 \times 100 = 91.3\%$
9	Apakah menu dan fasilitas sistem ini sangat mudah digunakan?	N: $1 \times 3 = 3$ S: $14 \times 4 = 56$ SP: $15 \times 5 = 75$ Total skor 134	$134/150 \times 100 = 89.3\%$
10	Apakah sistem bisa mudah dipasang di <i>smartphone</i> anda?	S: $14 \times 4 = 56$ SP: $16 \times 5 = 80$ Total skor 137	$137/150 \times 100 = 90.6\%$

No	Pertanyaan	Total Skor	Rerata
		TxPn	Total Skor / Y x 100
11	Apakah anda menerima informasi zakat dan <i>mustahik</i> yang anda butuhkan tepat waktu?	TP: 1 x 2 = 2 N: 4 x 3 = 12 S: 24 x 4 = 96 Total skor 110	110/150*100 = 73,3%
12	Apakah sistem menyediakan informasi zakat terbaru	TP: 1 x 2 = 2 N: 14 x 3 = 36 S: 15 x 4 = 60 Total skor 98	98/150*100 = 65,3%
Rerata Keseluruhan Pertanyaan			78.4%

Hasil rerata keseluruhan pertanyaan kuesioner didapatkan nilai 78.4%. Berdasarkan skala ukuran, sistem informasi ZIS mendapatkan hasil “Puas” dengan rentang nilai 60-79.99%.

4.3.3 Pengujian Versi Rilis Satu

Pengujian versi rilis satu dilakukan sepenuhnya oleh penulis, versi rilis merupakan perancangan sistem iterasi ketiga hasil dari pengujian versi *alpha* dan kuesioner pada pengujian versi *beta*. Pengujian versi rilis dilakukan dengan cara turun langsung ke lapangan menentukan posisi dan pelacakan keakuratan posisi *mustahik*, serta melakukan pengecekan pada kalkulator zakat. Berikut hasil iterasi ke tiga dibuat pada versi rilis yang ditunjukkan pada tabel 12.

Tabel 12. Penambahan Dan Perubahan Sistem Informasi ZIS Berdasarkan Tiga Kali Iterasi

No	Aplikasi Android Muzakki		
	Penambahan	Perubahan	Keterangan
1	Halaman profil <i>muzakki</i>		Setelah <i>muzakki login</i> akan diarahkan langsung ke halaman profil.
2	Halaman penilaian		Penilaian dan ulasan dijadikan 2 halaman dengan 2 tabel. Halaman ulasan Halaman nilai Tabel Nilai_Ulasan menjadi Tabel_Nilai dan tabel_Ulasan
3	Halaman Peta <i>Mustahik</i>		Menampilkan seluruh <i>mustahik</i> pada peta Google Maps berdasarkan wilayah dan kategori.
4	Penggunaan Firebase dan file JSON <i>google-services.json</i>	API <i>play-services-maps:11+</i> API <i>play-services-gcm:11+</i>	Memperbaiki masalah akurasi posisi <i>mustahik</i> .
5		Kalkulator zakat	Perbaikan pada rumus dan hasil zakat.
6	<i>News Feed</i>	.	Selain pesan, <i>muzakki</i> bisa mendapat informasi berdasarkan wilayah. <i>News feed</i> dibutuhkan karena pada beberapa <i>smartphone</i> notifikasi pesan tidak tampil.
Website Admin			
	Penambahan	Perubahan	Keterangan
7	Halaman <i>news feed</i> , tabel_Pesan_Feed		Untuk mengirim <i>news feed</i> kepada para <i>muzakki</i>
8	Halaman profil admin		
9	Status <i>publish</i> kategori	Tabel Kategori	
10	Status <i>publish</i> sub kategori	Tabel sub_kategori	

5. Kesimpulan dan Saran

5.1. Kesimpulan

Dari masalah yang telah diteliti tentang penyaluran zakat secara langsung, maka diperlukan sistem informasi yang bisa mempertemukan *muzakki* dan *mustahik*. Untuk mencapai tujuan tersebut diperlukan perancangan sistem informasi zakat yang terintegrasi dengan Google Maps. Beberapa hal dapat disimpulkan sebagai hasil dari penelitian yang dilakukan berdasarkan tahapan perancangan sistem pada tabel 13.

Tabel 13. Kesimpulan Rancangan Sistem Informasi ZIS

No	Keterangan
----	------------

1	Penelitian ini memfokuskan pada perancangan dengan metode <i>extreme programming</i> yang mengutamakan kebutuhan pengguna. Setiap permasalahan dan kebutuhan perancangan diubah dengan cepat dan berulang.
2	Tahap perancangan melalui tiga kali iterasi, dan setiap iterasi dilakukan pengujian (versi <i>alpha</i> , <i>beta</i> dan rilis). Dengan umpan balik dari pengguna akhir menjadi bahan iterasi dan perbaikan untuk menyempurnakan perancangan sistem.
3	Pada hasil kuesioner keakuratan posisi <i>mustahik</i> pada peta masih menjadi kendala, ditunjukkan dengan hasil 2 orang memilih tidak puas dan 19 orang memilih netral. Tetapi hasil rerata keseluruhan pertanyaan kuesioner didapatkan nilai 78.4% dari pengguna akhir. Berdasarkan skala ukuran, sistem informasi ZIS mendapatkan hasil "Puas" dengan rentang nilai 60-79.99%.
4	Hasil akhir perancangan yaitu aplikasi sistem informasi ZIS versi rilis satu.

5.2. Saran

Sistem informasi ZIS ini tentunya tidak terlepas dari kekurangan dan kelemahan. Oleh karena itu, untuk kebaikan pengembangan sistem selanjutnya, maka penulis menyarankan beberapa hal, diantaranya: Perlu adanya pengecekan dan perbaikan sistem dari sisi keamanan (*security*), perbaikan pada akurasi *latitude* dan *longitude*, perbaikan pada kalkulator zakat yang belum terkoneksi secara otomatis dengan harga emas ANTAM dan sistem yang dibuat sebatas aplikasi android dan *website*, belum pada tahap perancangan *multi-platform* (sistem operasi IOS, Windows *Mobile*, dan Symbian).

Referensi

- Ahmed, A., Ahmad, S., Ehsan, N., Mirza, E., & Sarwar, S. Z. (2010). *Agile software development: Impact on productivity and quality*. In Management of innovation and technology (ICMIT), 2010 IEEE international conference on(pp. 287-291).
- Ariaji, T., Utami, E., & Sunyoto, A. (2014). Evaluasi Sistem Informasi Yang Dikembangkan Dengan Metodologi *Extreme Programming*. *Jurnal Ilmiah DASI*, 15(4) : 53-62.
- Asghar, M.Z., Batool, U., Bibi, F., Ismail, S., & Zahra, S.R. (2016). Financial Studio: Android Based Application for Computing Tax, Pension. *International Journal of Academic*, 4 (2):96-117.
- Badan Pusat Statistik (BPS), diakses dari <http://www.bps.go.id/>, diakses pada tanggal 5 Juni 2018 pada jam 08.20 WIB.
- Doll, W. J. & Torkzadeh, G. (1988). The Measurement of End-User Computing Satisfaction. *MIS Quarterly*, 12(2) : 259-274.
- Fahmi, A., & Sugiarto, E. (2015). Aplikasi Sistem Informasi Geografis Manajemen Aset Wakaf. *Prosiding SNATIF ke-2*, pp.357-364.
- Krishna, T.S.R., Kanth, CH.P., Krishna, CH.V., & Krishna. T.T.V. (2011). Survey on Extreme Programming in Software Engineering. *International Journal of Computer Trends and Technology*, 2(2).
- Meier R. (2009). *Professional Android Application Development*. Indianapolis. Indiana, United States of America: Wiley Publishing, Inc.
- Pratama, Y.C. (2015). Peranan Zakat Dalam Penanggulangan Kemiskinan (Studi Kasus: Program Zakat Produktif Pada Badan Amil Zakat Nasional). *The Journal of Tauhidinomics*, 9(1) : 93-104.
- Republik Indonesia. (1992). *UU Nomor 38 Tahun 1999 Tentang Pengelolaan Zakat*. Lembaran Negara Republik Indonesia Tahun 1999 Nomor 164. Jakarta: Sekretariat Negara.
- Rob, M. A. (2003). Applications of geographical information systems in understanding spatial distribution of asthma. *INFORMING SCIENCE*, 6 : 89-99.
- Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile processes and methodologies: A conceptual study. *International journal on computer science and Engineering*, 4(5) : 892.
- Sugiyono. (2010). *Metode Penelitian Kuantitatif kualitatif & RND*. Bandung: Alfabeta.
- Suzanto, B., & Sidharta, I. (2015). Pengukuran *End-User Computing Satisfaction* Atas Penggunaan Sistem Informasi Akademik. *Jurnal Ekonomi, Bisnis & Entrepreneurship*, 9(1) : 16-28.