

Rancang Bangun *User Management* pada Aplikasi tetanggaBaik Sebagai Jejaring Sosial Untuk Perumahan di Indonesia Berbasis PHP

Ryan Surya, Undergraduate Student Teknik Informatika, Universitas Ciputra Surabaya, UC Town, CitraLand, Surabaya 60219

ABSTRAK

Seiring dengan berkembangnya pembangunan di Indonesia terutama pada bisnis *property*, semakin banyak juga orang yang mulai memilih untuk bertempat tinggal di sebuah kompleks perumahan. Para penghuni perumahan tersebut semakin bertambah akan tetapi banyak diantara penghuni tersebut saling tidak mengenal antara satu dengan yang lain. Selain itu para penghuni ketika mendapatkan masalah di suatu perumahan, sering kali juga para penghuni tersebut tidak tahu harus melapor kemana agar informasi yang mereka sampaikan dapat sampai dengan tepat dan diterima oleh pihak perumahan. Hal inilah yang menjadi tantangan seorang pengusaha *property* jika pengusaha tersebut memiliki keinginan untuk mewujudkan lingkungan perumahan yang harmonis dan aman. Oleh karena itu dibutuhkan suatu media dalam bentuk *Social Network* yang dibuat dengan serangkaian studi mengenai pembuatan perangkat lunak, observasi dan *survey* mengenai pendapat penghuni perumahan, pengujian dan evaluasi agar dapat digunakan dengan mudah dan tentunya juga menjadi berguna bagi pemakai. *Social Network* ini diharapkan menjadi tempat agar para tetangga perumahan tersebut dapat menjalin hubungan antara sesama tetangga maupun dengan perumahan. Metode pengembangan aplikasi diawali dengan mempelajari teknologi yang akan digunakan, kemudian mengumpulkan data dan melakukan analisa mengenai fitur utama, melakukan perancangan sistem dan melakukan pembuatan aplikasi sesuai dengan rancangan dan yang terakhir adalah melakukan pengujian fitur dan melakukan perbaikan jika terjadi error. Hasil dari penerapan ini mampu membantu suatu warga perumahan dalam bertukar informasi dengan pihak perumahan. Hal ini juga dapat menciptakan hubungan baik antar tetangga maupun dengan pihak perumahan. Selain itu, dapat disimpulkan juga bahwa aplikasi tetanggaBaik ini cocok digunakan untuk perumahan yang memiliki pengelola.

Kata Kunci : sistem, jejaring sosial, *web*, pengusaha *property*, penghuni perumahan

1. Pendahuluan

1.1. Latar Belakang

Perkembangan bisnis properti khususnya perumahan berkembang pesat di kota-kota besar di Indonesia. Dengan semakin banyaknya jumlah perumahan, semakin banyak pula orang-orang yang tinggal di perumahan. Arti kata perumahan adalah kumpulan beberapa rumah yang berfungsi sebagai lingkungan tempat tinggal atau lingkungan hunian yang dilengkapi dengan prasarana dan sarana lingkungan (Sugono, 2008). Banyak orang tinggal di perumahan dikarenakan adanya suasana dan lingkungan yang tenang, menyenangkan, dan juga rasa aman atau juga sebagai investasi. Walaupun pada suatu perumahan terdapat fasilitas-fasilitas yang lengkap dari perumahan untuk penghuninya tetapi sering kali terdapat suatu *gap* antara penghuni dan pihak perumahan yang terkadang dapat menjadi sumber masalah misalnya dalam hal penyampaian informasi dari pihak perumahan kepada penghuninya. Selain antara penghuni dan pihak perumahan, terkadang sesama penghuni atau tetangganya bahkan tidak saling mengenal satu dengan yang lain.

Dengan masalah tersebut, maka dibuatlah aplikasi jejaring sosial yang digunakan sebagai jembatan antara pihak perumahan dan penghuninya ataupun antar sesama

penghuninya dalam bertukar informasi. Penggunaan aplikasi ini dari pihak perumahan ditujukan agar pihak perumahan dapat memberikan informasi kepada penghuni tentang hal-hal yang terjadi di lingkungan perumahan, seperti pemadaman listrik dan air secara mendadak, adanya penutupan jalan, pemberian informasi pembukaan sarana dan prasarana baru, pembukaan tipe-tipe rumah baru, dan lain-lainnya. Pihak penghunipun juga dapat menyampaikan informasi kepada pihak perumahan seperti adanya orang mencurigakan berkeliaran di lingkungan perumahan, ataupun terjadi kejadian yang membutuhkan penanganan cepat lainnya seperti kebakaran, banjir, pencurian, adanya sampah yang menumpuk, atau ada pohon tumbang, dan sebagainya. Selain itu sesama penghuni juga dapat saling berinteraksi, saling memberikan informasi misalnya menanyakan apakah ada tetangga yang tahu tentang anjingnya yang hilang, menanyakan informasi tentang tukang yang bagus di daerah tersebut atau saling memberikan informasi klinik terdekat.

Metode yang digunakan untuk melakukan pembuatan perangkat lunak menggunakan metode *waterfall model*. Metode ini digunakan karena melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, perancangan, implementasi, pengujian, dan *maintenance*, sehingga

aplikasi yang dihasilkan akan baik karena tahapan berikutnya dilakukan jika tahapan sebelumnya sudah benar. (Basuki, 2006).

1.2. Rumusan Masalah

Bagaimana merancang dan membangun *user management* pada aplikasi tetanggaBaik sebagai jejaring sosial untuk perumahan di Indonesia berbasis PHP ?

1.3. Ruang Lingkup

- Perangkat lunak yang dibuat dapat saling melihat dan memberikan informasi baik antara pemilik perumahan dan juga pihak perumahan
- Pengguna perangkat lunak ini ditujukan untuk departemen perumahan dan penghuni yang tinggal di perumahan tersebut.
- Informasi yang bersifat darurat akan langsung dapat diterima oleh bagian keamanan setempat *via email* dengan menggunakan mailgun.
- Perangkat lunak ini dibuat dengan menggunakan kombinasi PHP dengan *Framework* Laravel, Twitter Bootstrap, MySQL dan penggunaan *email server* Mailgun.

1.4. Tujuan

Tujuan dari penelitian ini adalah merancang dan membangun *user management* pada aplikasi tetanggaBaik sebagai jejaring sosial untuk perumahan di Indonesia berbasis PHP.

1.5. Manfaat

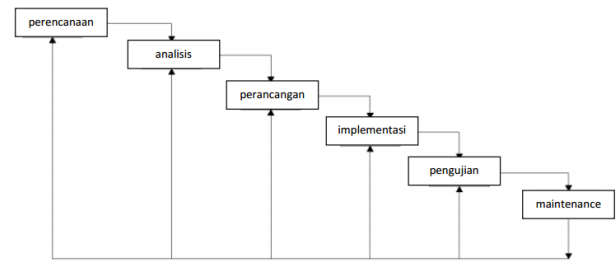
Manfaat yang dapat ditimbulkan adalah adanya *social network* yang dipakai pada suatu perumahan yang dapat membantu proses penyampaian informasi secara cepat antar pihak perumahan dan penghuninya. Dengan adanya aplikasi ini dapat menciptakan hubungan baik dan mempercepat dan mempermudah proses penyampaian informasi antar pihak perumahan dan penghuninya.

2. Landasan Teori

2.1. System Development Life Cycle dengan metode Waterfall.

Dalam pembuatan sebuah sistem perlu dilakukan tahapan-tahapan pengembangan dalam prosesnya. Tahapan itu adalah perencanaan sistem, analisis sistem, desain dan perancangan sistem, implementasi sistem, pengujian sistem dan proses *maintenance*. Tahapan itu saling mempengaruhi dan membentuk sebuah siklus, yaitu jika pada suatu tahapan ternyata ada data-data yang seharusnya dilakukan pada tahapan sebelumnya, maka dapat kembali ke tahapan sebelumnya. (Basuki, 2006)

Untuk melihat bentuk dari metode *waterfall* dapat dilihat pada Gambar 1.



Gambar 1 Tahapan dengan metode Waterfall

2.2. Jejaring Sosial

“Kita mendefinisikan *social network* sebagai *web-base service* yang memberikan kemampuan kepada seseorang untuk (1) membangun profil publik atau *semi-public* dalam sistem yang terikat, (2) artikulasi sekumpulan *user* dimana mereka memiliki suatu koneksi dan (3), menampilkan dan melintasi list koneksi mereka yang terjalin dalam sistem. Hubungan tersebut dapat berubah-ubah dari satu situs dan situs lainnya” (Boyd & Ellison, 2007)

Jejaring sosial secara sederhana dapat dijelaskan sebagai struktur relasi yang menghubungkan antara seseorang dalam suatu lingkungan atau perkumpulan. Jejaring sosial telah memanfaatkan sistem informasi yang telah berkembang sekarang ini agar setiap pengguna dapat menemukan teman baru tanpa dibatasi oleh jarak dan waktu.

2.3. Basis Data

Database adalah kumpulan *field-field* yang mempunyai kaitan antara satu *field* dengan *field* lain sehingga membentuk satu bangunan data untuk menyimpan informasi dalam batasan tertentu. (Pratama, 2013)

2.4. Pengembangan web

2.4.1. Web Server

Web server merupakan sebuah perangkat lunak dalam *server* yang berfungsi menerima permintaan (*request*) berupa halaman web melalui *HTTP* atau *HTTPS* dari klien yang dikenal dengan browser dan mengirimkan kembali (*response*) hasilnya dalam bentuk halaman-halaman *web* (Solichin, 2008).

2.4.2. PHP

“PHP digunakan sebagai *open source*, digunakan dengan tujuan bahasa *scripting*. Pada awal digunakan sebagai pembuatan *web site*. Kenyataannya, awal mula PHP digunakan sebagai alat pembuatan *personal home page* yang di buat oleh Rasmus Lerdorf untuk membantu pengguna dalam pembuatan *web page*-nya. PHP terbukti bermanfaat dan populer menjadikannya berkembang dengan cepat menjadi bahasa yang memiliki fitur yang lengkap, memperoleh nama PHP *Hypertext Preprocessor* untuk menunjukkan kemampuannya yang sudah diperluas dalam mengolah *web page* sebelum ditampilkan.” (Valade, 2004).

2.4.3. Framework Laravel

“Laravel adalah *framework* yang kuat, didesain untuk pembuatan aplikasi *web* dengan menggunakan *php*. Laravel akan memberikan jalan untuk memodulkan *code* kita dan membuatnya terlihat lebih bersih dan mudah

untuk dibaca dan dipahami.” (Jain, 2012)

Laravel merupakan *framework* php dengan konsep MVC sehingga memudahkan untuk pembuatan program berbasis php.

2.4.4. Twitter Bootstrap

Twitter Bootstrap adalah sebuah alat bantu untuk membuat sebuah tampilan halaman *website* yang dapat mempercepat pekerjaan seorang pengembang *website* ataupun pendesain halaman *website*. (Ridha, 2013).

2.5. Mailgun

“Mailgun is an email automation service provided by Rackspace. It offers a complete cloud-based email service for sending, receiving and tracking email sent through your websites and applications. Mailgun features are available through an intuitive RESTful API or using traditional email protocols like SMTP.”

“Mailgun adalah sebuah jasa email otomatisasi yang disediakan oleh Rackspace. Mailgun memberikan jasa dengan basis-*cloud* lengkap untuk mengirimkan, menerima dan menelusuri email yang dikirimkan melalui *website* dan aplikasi anda. Fitur Mailgun dapat digunakan melalui RESTful API intuitif atau menggunakan protokol email tradisional seperti SMTP”. (Nabors, 2015)

Mailgun merupakan jasa email otomatisasi yang membantu pengembang perangkat lunak dalam hal pengiriman email sehingga email yang dikirimkan dari aplikasi tidak masuk pada *spam*.

3. Desain Sistem

Membahas mengenai tahapan Analisis dan Perancangan dari Metode Pembuatan Perangkat Lunak *Waterfall Model*. Melanjutkan dari tahap perencanaan *waterfall model* yang dijelaskan pada Bab II. Pada perancangan perangkat lunak ini menggunakan metode *waterfall model* karena fitur dari perangkat lunak sudah didefinisikan pada awal proses sehingga pengerjaannya akan menjadi lebih mudah, dan juga metode ini pengerjaannya dilakukan secara bertahap oleh karena itu jika perangkat lunak telah selesai dibuat maka dapat dipastikan bahwa perangkat lunak akan berjalan dengan baik sesuai dengan fitur awal aplikasi yang direncanakan.

Pada tahapan analisis, dilakukan survey kepada 45 responden yang berasal dari warga Citraland Surabaya. Dengan pertanyaan yang mencakup : (1) Intensitas masalah yang dapat terjadi pada sebuah perumahan (2) Masalah sosial yang dapat terjadi di dalam sebuah perumahan (3) Masalah yang dapat terjadi dari luar kehendak manusia seperti bencana alam (4) Perilaku pengirim dan penerima laporan (5) Kemampuan seseorang untuk memberikan informasi secara benar dan aktual (6) Kemudahan menggunakan akses internet dan online dengan fitur yang wajib ada dalam sebuah jejaring sosial. Untuk pengolahan datanya menggunakan tabel distribusi frekuensi data dengan menghitung nilai modus dan analisa data berupa kalimat respon dikelompokkan sehingga menghasilkan hasil dalam bentuk kesamaan konteks.

Dari hasil penelitian, masing-masing responden memberikan jawaban dimana sering terjadi masalah dalam suatu perumahan dan dibutuhkan sesuatu sebagai sarana penyampaian informasi dan laporan kepada pihak

perumahan.

Setelah *preliminary research* yang dilakukan maka didapatkan fitur-fitur yang dibutuhkan adalah :

a. Landing Page

Halaman ini akan berisi informasi mengenai keuntungan dan kelebihan menggunakan perangkat lunak, *slider* foto-foto dari perumahan dan menu sebagai berikut:

- i. *About us*: berisi penjelasan tentang *owner* pembuat perangkat lunak ini beserta tujuan dan visi misi mengapa *owner* membuat perangkat lunak ini.
- ii. *Term and condition* : berisi penjelasan tentang peraturan-peraturan dalam menggunakan perangkat lunak yang ditujukan kepada *user*.
- iii. *Help and FAQ* : berisi pertanyaan-pertanyaan dan jawaban dari pertanyaan yang sering ditanyakan oleh *user* perihal tentang perangkat lunak.
- iv. *Login* : terdapat 4 macam *login* yaitu *login* sebagai *user*, *login* sebagai *admin* posting, *login* sebagai *admin* keamanan, *login* sebagai *admin confirm*.
- v. *Forgot Password* : digunakan untuk *user* yang kelupaan akan *password* miliknya dan menginginkan untuk *me-reset passwordnya* dengan yang baru.

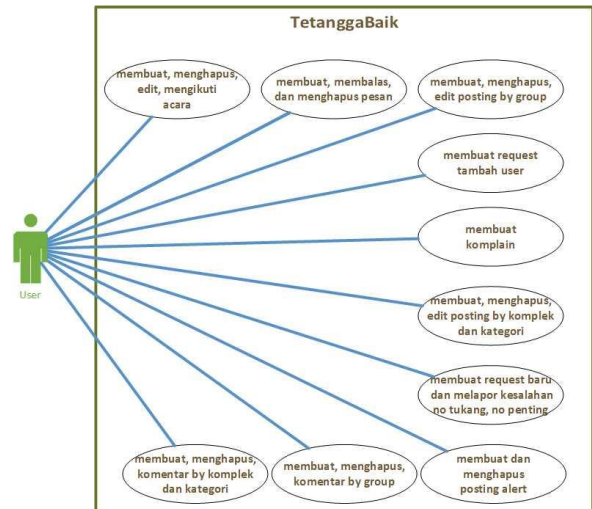
b. Home Page (setelah melakukan login)

i. Main Menu

1. Notifikasi : *user* dapat melihat semua kegiatan yang terjadi pada aplikasi ini baik oleh *user* lain ataupun dirinya sendiri tentang *posting* yang ada ataupun *event* yang dibuat.
2. Posting Saya : *user* dapat melihat posting yang pernah dilakukan oleh dirinya sendiri juga mengedit isi posting yang dilakukan *user*
3. Buat Posting Baru : *user* dapat melakukan *posting* pada halaman *home* atau grup atau *local* atau melakukan *posting* berupa *alert*.
4. Pesan : dapat mengirimkan pesan singkat kepada tetangga dengan meng-*inputkan* nama tetangga dan isi dari pesan singkatnya. Jika ada pesan baru akan ada notifikasi pada halaman *home page user*. Dapat melihat *inbox* dan *outbox* pesan yang dikirimkan kepada kita.
5. Tetangga : dapat melihat tetangga kita dalam 1 *local* area ataupun dalam keseluruhan area perumahan tanpa harus melakukan *add friend* terlebih dahulu.
6. Acara : acara yang berisi informasi tentang acara yang ada pada kompleks pengguna. *User* juga membuat acara yang dapat ditujukan kepada tetangga, atau kompleks area dan juga mendekati hari-h akan ada peringatan berupa notifikasi di halaman notifikasi.
7. Tambah user : digunakan untuk menambahkan keterangan tentang anggota keluarga yang berada dalam 1 rumah.
 - i. Informasi : menampilkan informasi yang di *post* oleh masing-masing departemen. *User* juga dapat *posting* dan memberikan komentar.
 - ii. Kompleks : menampilkan informasi yang di *post* pada 1 *local* area tertentu atau

- keseluruhan dari kompleks area. *User* juga dapat *posting* dan memberikan komentar
- iii. *Groups* : menampilkan informasi yang di *post* pada grup-grup yang diikuti oleh *user*. *User* juga dapat *posting* dan memberikan komentar.
 - iv. *Request Groups* : *user* dapat melakukan *request groups* tertentu kepada *admin* dan *admin* akan membuat grup baru setelah mengetahui tujuan pembuatan grup dengan jelas.
 - v. *Kategori* : menampilkan informasi yang di *post* oleh *user* berdasarkan kategorinya. Secara *default* kategori yang ada pada awalnya adalah peristiwa, hilang-nemu, jual beli. *User* dapat menambahkan kategori sendiri dengan persetujuan dari *admin*. *User* juga dapat *posting* dan memberikan komentar.
 - vi. *Alert* : postingan yang dilakukan dengan menggunakan *alert* akan di emailkan ke pihak departemen yang bersangkutan atau pada 1 kompleks perumahan. *Alert* ini akan langsung terintegrasi dengan pihak departemen yang bersangkutan
 - vii. *Search* : dapat digunakan untuk mencari data yang berhubungan yang terdapat dalam aplikasi dengan *inputan* kata-kata.
 - viii. *Nomor Penting* : akan berisi informasi-informasi yang merupakan kebutuhan *user* yang ada dalam 1 lingkungan perumahan, seperti informasi angkutan umum, informasi klinik, informasi tukang bangunan, dan sebagainya.
 - ix. *Nomor Penting* : berisi informasi dari tukang yang bekerjasama yang dapat dipanggil kerumah. Tukang tersebut dapat dihubungi sendiri oleh *user* melalui telepon atau menggunakan *form* yang ada dan akan di kirimkan ke tukang yang bersangkutan melalui sms dengan menggunakan sms *gateway*.
 - x. *Partner page* : berisi informasi dari *partner* yang bekerjasama, misalnya adalah *partner minimarket* yang dapat mem-*posting* ada promo atau diskon.
 - xi. *Kontak Kami*: berisi *form* yang dapat digunakan untuk memberikan saran, kritik kepada pihak departemen perumahan.
 - xii. *Profile* : berisi informasi tentang *user* baik berupa foto *user*, tanggal lahir *user*, nomor telepon *user* dan sebagainya.
 - xiii. *Riwayat Alert* : berisi semua *alert* yang pernah diposting oleh semua pengguna aplikasi.

3.1. Use Case Diagram



Gambar 3 Use Case User TetanggaBaik

4. Implementasi dan Pengujian

Membahas mengenai tahapan implementasi, pengujian, dan maintenance dari metode pembuatan perangkat lunak *Waterfall Model*.

Pada tahapan implementasi terdiri dari lingkungan pengembangan perangkat lunak yaitu spesifikasi dari laptop yang digunakan untuk membuat, spesifikasi *hosting* yang digunakan untuk meng-*upload* dan sebagai tempat untuk mengakses/menggunakan aplikasi, spesifikasi dari modem yang digunakan pada saat uji coba aplikasi, implementasi penggunaan framework laravel, implementasi penggunaan *email server Mailgun* pada aplikasi.

Pada tahapan pengujian, aplikasi diujikan kepada 10 orang untuk mengetahui dan menguji apakah aplikasi dapat berjalan dengan lancar dan seharusnya tanpa adanya error atau function yang tidak bekerja dengan baik.

4.1. Spesifikasi dalam Pembuatan Aplikasi

Spesifikasi Laptop yang digunakan dalam pembuatan perangkat lunak :

- Toshiba tipe Satellite M-300
- Processor Intel® Core™ Duo CPU T6400 @ 2.00GHz 2.00GHz
- RAM 1 GB
- XAMPP 5.6.3, Apache 2.4.10
- Sistem operasi Microsoft Windows 8.1 pro 32 bit
- RAM 1 GB
- MySQL v 5.6.21
- PHP 5.6.3
- *Framework* Laravel v 4
- *Browser* Chrome v 42.0.2311.135 m

Spesifikasi *Hosting* yang digunakan untuk menaruh dan akses perangkat lunak :

- *Framekwork* Laravel v 4
- *Bandwith* 16 GB
- PHP 5.4.38
- MySQL v 5.5.42-c11
- Apache v 2.2.29
- cPanel v 11.48.3
- Sistem Operasi Linux

4.2. Implementasi Framework Laravel

Dalam pengembangan aplikasi tetanggaBaik digunakan teknologi Laravel sebagai framework untuk berkomunikasi antara server, client dan database. Laravel ini sendiri adalah aplikasi web yang bertujuan untuk membuat proses developing lebih mudah tanpa mengorbankan aplikasi pengembangan

4.3. Implementasi Mailgun

Mailgun merupakan sebuah layanan yang dapat digunakan untuk mengirim email melalui request yang dibuat oleh aplikasi. Menggunakan Mailgun karena penggunaan mailgun telah terintegrasi dengan framework laravel dan pengiriman email melalui email server mailgun guna menghindari email yang dikirim masuk pada spam.



Gambar 4.1 Halaman mailgun

Pada Gambar 4.1 ditunjukkan table pengiriman email yang dikirim dimana warna hijau menunjukkan email yang terkirim dan warna merah menunjukkan email yang di tolak dikarenakan dianggap spam dikarenakan adanya html tag yang terkirim pada email.

Date/Time	Summary
05/03/15 02:41 AM	Delivered: postmaster@mg.ryansurya.com → edwinazmirandan@gmail.com 'Welcomel to tetanggaBaik.com.'
05/03/15 02:41 AM	Accepted: postmaster@mg.ryansurya.com → edwinazmirandan@gmail.com 'Welcomel to tetanggaBaik.com.'
05/03/15 02:36 AM	Delivered: postmaster@mg.ryansurya.com → lamlielong@gmail.com 'Welcomel to tetanggaBaik.com.'
05/03/15 02:36 AM	Accepted: postmaster@mg.ryansurya.com → lamlielong@gmail.com 'Welcomel to tetanggaBaik.com.'
05/03/15 01:54 AM	Delivered: postmaster@mg.ryansurya.com → ricocos.nucifera@yahoo.com 'Add User Notification'
05/03/15 01:54 AM	Accepted: postmaster@mg.ryansurya.com → ricocos.nucifera@yahoo.com 'Add User Notification'

Gambar 4.2 Riwayat pengiriman

Pada Gambar 4.2 ditunjukkan riwayat table pengiriman email yang ditampilkan pada halaman mailgun.

```
Mailgun::send('mails.contact', array('name' => Auth::user()->name,
    'email' => Input::get('email'), 'body' => Input::get('body') ), function($message)
    {
        $admin_posting = user::where('role_id',3)->get();
        foreach ($admin_posting as $admin ) {
            $message->to($admin->email, $admin->name)->subject(Input::get('title'));
            $message->from(Input::get('email'));
        }
    });
```

Gambar 4.3 Code pengiriman email dengan mailgun

pada Gambar 4.3 ditunjukkan contoh code yang dipakai dalam pengiriman email yang ketika melakukan pengiriman email ke admin dengan menggunakan mailgun yang berfungsi agar email yang dikirimkan tidak masuk ke dalam spam.

4.4. Pengujian

Pada tahapan ini merupakan tahapan kelima dari *waterfall model*, pengujian dilakukan kepada 10 orang dengan tujuan dilakukan pengujian aplikasi ini adalah untuk mengetahui apakah aplikasi yang dibuat telah berjalan dengan.

4.4.1. Skenario Pengujian User

• Skenario Login

Tujuan dilakukan pengujian aplikasi ini adalah untuk mengetahui apakah aplikasi yang dibuat telah berjalan dengan baik serta mendapatkan pendapat *user* mengenai struktur *user interface* dan fitur aplikasi.

• Skenario Posting

User membuat *postingan* baru (terserah *user* pada jenis *posting* apapun) , kemudian membuka *post user* kembali pada menu yang telah disediakan dan kemudian *edit post* yang baru *user* buat.

• Skenario Pesan

User diminta membuat pesan baru kepada *admin* "admin posting" kemudian *user* diminta membalas pesan dari "Ryan Surya" dan akhirnya *user* melihat "pesan keluar" apakah pesan telah terkirim atau belum.

• Skenario Cari Tetangga dan Tambah User

User diminta mencari tetangga yang bernama "ryan surya" pada halaman tetangga anda dengan menggunakan fitur pencarian (cari *user*) dan kemudian *user* diminta membuat *request user* baru pada halaman tambah user.

• Skenario Acara

User diminta membuat acara baru dengan syarat minimal mengajak 2 kompleks , kemudian *user* melihat acara "testing aplikasi tetangga baik" yang dibuat oleh "ryan surya", dan memberikan kepastian mengikuti acara atau tidak , setelah itu *user* melihat untuk melihat diminta list peserta pada acara "testing aplikasi tetangga baik".

• Skenario Nomor Penting/Nomor Tukang

User diminta menambah salah satu antara nomor penting / nomor tukang dan kemudian *user* diminta untuk membuat laporan salah nomor, salah satu antara nomor penting / nomor tukang.

• Skenario Kontak Kami

User diminta memberikan komentar pendek tentang aplikasi ini melalui "kontak kami".

• Skenario Komentar

User diminta membuka komentar postingan yang berada pada kategori "hilang nemu" kompleks "puri widya kencana" dan kemudian memberikan komentar pada salah satu *posting*.

• Skenario Profil

User diminta membuka halaman profil *user* dan mengganti *data user* (boleh data akun , ataupun *password*).

4.4.2. Hasil Pengujian

No	Skenario	Expected Result	Actual Result	Pass/Fail
1	Login	User berhasil login dan masuk ke halaman user	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0
2	Posting	User berhasil membuat post baru , kemudian membuka halaman “posting saya” dan mengubah posting dengan link ubah post	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0
3	Pesan	User berhasil membuat pesan kepada admin posting ,berhasil membalas pesan dari ryan surya dan pesan keluar tampil pada halaman pesan keluar	Hasil yang didapatkan sesuai dengang yang di harapkan , terkecuali pada satu user dimana user mendapatkan bug pada saat membalas pesan, dan pada satu user , user bingung cara mencari admin posting	Pass = 8 Fail = 2
4	Cari Tetangga dan Tambah User	User berhasil membuat request user baru dan notifikasi request di email ke admin	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0
5	Acara	User berhasil membuat acara dengan mengajak minimal 2 kompleks , memberikan kepastian mengikuti acara yang telah ada sebelumnya dan dapat melihat list peserta yang mengikuti acara tersebut	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0
6	Nomor Penting/ Nomor Tukang	User berhasil membuat request nomor baru dan berhasil membuat laporan salah nomor yang di email ke admin sebagai	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0

notifikasi				
7	Kontak Kami	User berhasil mengirimkan isi dari form kontak kami kepada email admin	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0
8	Komentar	User berhasil membuka postingan yang telah dibuat sebelumnya dan berhasil membuat komentar untuk posting ini	Hasil yang didapatkan sesuai dengan yang di harapkan	Pass = 10 Fail = 0
9	Profil	User berhasil membuka halaman pofil dan mengganti data diri / password	Hasil yang didapatkan sesuai dengan yang di harapkan , terkecuali pada satu user , terjadi bug dikarenakan faktor extensi gambar	Pass = 9 Fail = 1
Rata-rata tingkat kemudahan akses fitur/user friendly (1-5)				3.7

Kritik dan saran dari pengujian :

- Lokasi masing-masing fitur lebih dipermudah
- Menambah tombol tambah post di halaman “posting saya”
- Notifikasi diperjelas setelah submit form
- Tampilan masih dapat diperbaiki lagi
- Tampilan untuk pertama kali dilihat mungkin agak memusingkan karena banyak text, tetapi kalau sudah terbiasa maka akan menjadi sangat simpel dan tidak akan memakan kuota banyak.
- Fitur fitur bagus. webnya menarik dan berguna juga. tingkatkan lagi!
- Kalau bisa . dibuat aplikasi mobile nya juga
- Ditambahkan fitur dengan menggunakan ajax
- Nomor penting dan nomor tukang dijadikan sub-menu dari Nomor saja
- ada baiknya jika percakapan antar user ditampilkan dalam 1 thread message (seperti gmail)

4.5. Pembahasan

Pada Skenario pesan terjadi gagal dikarenakan faktor UI atau tampilan yang kurang cocok pada aplikasi tahap pertama sehingga mengakibatkan kebingungan user dalam mencari user lain. Dan terjadi bug pada saat pertama kali aplikasi jalan dikarenakan kesalahan pembuat aplikasi sendiri dalam mengirim data aplikasi pada server hosting yang digunakan. Untuk uji coba fitur yang lain berhasil sesuai dengan hasil yang diharapkan.

5. Kesimpulan

Aplikasi yang diuji coba berhasil berjalan dengan baik dan dapat disarankan untuk digunakan pada suatu lingkungan perumahan yang memiliki pengelola (telah memiliki sistem) dalam bertukar informasi antar warganya atau dengan pihak perumahan.

DAFTAR PUSTAKA

- Adi, W., & Biannto, I. (2010). Layanan Informasi Untuk Pasien Rumah Sakit Menggunakan SMS. *Jurnal Penelitian*, 13(2), 199-216. Retrieved May 18, 2014, from
- Boyd, D. M., & Ellison, N. B. (2007). *Social Network Sites: Definition, History, and Scholarship*. *Journal of Computer-Mediated Communication*, 13(1), 210–230. Retrieved May 18, 2014, from <http://onlinelibrary.wiley.com/doi/10.1111/j.1083-6101.2007.00393.x/full>
- Jain, A. (2012). *An Introduction about Laravel Framework*. Retrieved May 18, 2014, from <http://mrbool.com/an-introduction-about-laravel-framework/26410>
- Laravel. (n.d.). *Documentation*. Retrieved from <http://laravel.com/docs/4.0>. Access date 1 May 2015
- Panji (2013). arti kata perumahan. Retrieved May 18, 2014, from <http://glosarium.org/arti/?k=perumahan>
- Pratama, A. (2013). Pengenalan MySQL. Retrieved May 18, 2014, from <http://ilmukomputer.org/2013/02/01/pengenalan-mysql/>
- Ridha, H. (2013). Implementasi Twitter Bootstrap pada https://www.usd.ac.id/lembaga/lppm/jurnal.php?id=abstraksi&model=volume&id_j=14&id_m=58&id_k=271
- Basuki, A. (2006). Pengembangan Perangkat Lunak. Surabaya, Indonesia: PENS-ITS.
- Code Igniter. Retrieved May 18, 2014, from <http://ilmukomputer.org/2013/05/10/implementasi-twitter-bootstrap-pada-codeigniter/>
- Rozidi, R. I. (2004). Membuat Sendiri SMS Gateway (ESME) Berbasis Protokol SMPP. Yogyakarta, Indonesia: Andi Offset.
- SMS Gateway API. (n.d.). *Your Android phone, Your SMS Gateway*. Retrieved from <https://msgateway.me>. Access date 1 May 2015
- Solichin, A. (2008). Pemrograman web dengan PHP dan MySQL. Jakarta, Indonesia.
- Sugono, D. (2008). Perumahan. In Kamus Bahasa Indonesia (4th ed., p. 1228). Jakarta, Indonesia: Pusat Bahasa.
- Utomo, P. A. (2006). Membangun Aplikasi SMS Berbasis Open Source. Yogyakarta, Indonesia: Andi Offset.
- Valade, J. (2004). *PHP 5 for Dummies*. Indianapolis, Indiana: Willey Publising, Inc.