**Pengembangan Modul Pelaporan Aplikasi Aktivitas Kampanye untuk Masyarakat dan Pemerintah Menggunakan REST APIs**

INDRIYANI (G64120028)[[1]](#footnote-1)\*, DEAN APRIANA RAMADHAN

**ABSTRAK**

Pengawasan aktivitas kampanye pada pemilihan umum (Pemilu) dilakukan oleh Badan Pengawas Pemilu (Bawaslu). Bawaslu memiliki kendala dalam pelaksanaan pengawasan yaitu kendala daya dukung pengawasan berupa keterbatasan jumlah Pengawas Pemilu Lapangan (PPL). Melihat kendala tersebut dibutuhkan media untuk pelaporan yang dapat dilakukan oleh masyarakat. Penelitian ini merupakan pengembangan modul pelaporan aplikasi aktivitas kampanye untuk masyarakat dan pemerintah menggunakan *REpresentational State Transfer (REST) Application Programming Interface* (API) dengan struktur basis data NoSQL. Metode yang digunakan dalam penelitian ini adalah *Extreme Programming.* Pemasukkan data pada REST API menggunakan Node.js dan hasil dari data tersebut berupa format teks JSON. Penelitian ini telah berhasil mengembangkan aplikasi *web* yang dapat membantu masyarakat dalam mengetahui informasi seputar calon legislatif serta lokasi-lokasi terjadinya kampanye.

Kata kunci : Bawaslu, JSON, kampanye, REST APIs

***ABSTRACT***

*Supervision of the campaign activity conducted by the Electoral Supervisory Agency (Bawaslu). Bawaslu has disadvantage on the implementing in the supervision that is carrying capacity obstacles which is limiting the number of Election Supervisors (PPL). Considering that obstacles required media for reporting that can be done by the society. This research developed a reporting module on the campaign activity applications for the society and the government using REpresentational State Transfer (REST) Application Programming Interface (API) with NoSQL database structure. The method used in this research is Extreme Programming. REST APIs is an architectural method of communication that was often applied in the development of web-based services. Inputting data at REST APIs using Node.js and the results of the data is the form of JSON text format. This research has successfully developed a web application that can help people to know about informaton from legislative candidates and the occurred locations of a campaign.*

*Keywords: Bawaslu, campaign, JSON, REST APIs, web applications*

**PENDAHULUAN**

**Latar Belakang**

Penerapan *Information* & *Communication Technology* (ICT) di pemerintahan merupakan upaya untuk mengembangkan penyelenggaraan pemerintahan yang berbasis elektronik dalam rangka meningkatkan transparansi dan kualitas pelayanan publik secara efektif dan efisien (Kominfo 2013). Salah satu contohpenerapan ICT dalam pemerintahan adalah *e-campaign*. *E-campaign* merupakan salah satu strategi dalam kampanye karena berkolerasi positif terhadap meningkatnya jumlah suara yang diperoleh (Ramadhan 2015). *E-campaign* membuat proses kampanye menjadi lebih interaktif yang memudahkan terjadinya interaksi antara calon legislatif dengan masyarakat melalui Internet. Menurut hasil riset yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), pertumbuhan penggunaan Internet di Indonesia terus meningkat. Pada tahun 2014, survei menunjukkan pengguna Internet di Indonesia mencapai 82 juta jiwa atau 34.9% dari seluruh penduduk Indonesia. 40 juta jiwa atau 49% pengguna Internet di Indonesia berusia 18–25 tahun, yaitu masyarakat yang telah memiliki hak suara dalam Pemilu.

Kelembagaan Pengawasan Pemilu atau Panitia Pengawas Pelaksanaan Pemilihan Umum (Panwaslak Pemilu) pertama kali muncul pada saat pelaksanaan Pemilu tahun 1982. Pembentukkan Panwaslak dilatari oleh protes atas banyaknya pelanggaran dan manipulasi perhitungan suara yang dilakukan oleh para petugas Pemilu pada tahun 1971. Pada tahun 2007 melalui Undang-Undang Nomor 22 Tahun 2007 tentang Penyelenggaraan Pemilu, kelembagaan pengawas Pemilu dikuatkan dengan dibentuknya sebuah lembaga tetap yang dinamakan Badan Pengawas Pemilu (Bawaslu). Sesuai dengan Undang-Undang RI No. 1 Tahun 2015 tentang Penetapan Peraturan Pemerintah Pengganti Undang-Undang Nomor 1 Tahun 2014 tentang Pemilihan Gubernur, Bupati, dan Walikota Menjadi Undang-Undang, saat ini pengawasan pelaksanaan Pemilu masih dilaksanakan oleh Bawaslu.

Salah satu kendala yang dialami Bawaslu dalam pengawasan Pemilu adalah kendala daya dukung pengawasan berupa keterbatasan jumlah Pengawas Pemilu Lapangan (Bawaslu 2014). Jumlah laporan pelanggaran oleh masyarakat dan peserta pada Pemilu Kepala Daerah dan Wakil Kepala Daerah (Pemilukada) tahun 2013 adalah 1.001 laporan sedangkan pada Pemilu Legislatif tahun 2014 adalah 2.722 laporan. Meningkatnya jumlah laporan dari masyarakat menunjukkan bahwa masyarakat dapat berperan dalam pengawasan Pemilu.

Menurut Peraturan Badan Pengawasan Pemilihan Umum Nomor 14 Tahun 2012 tentang Tata Cara Pelaporan dan Penanganan Pelanggaran Pemilihan Umum Anggota Dewan Perwakilan Rakyat, Dewan Perwakilan Daerah, dan Dewan Perwakilan Rakyat Daerah, tata cara melaporkan pelanggaran dalam pengawasan Pemilu yaitu; (1) Pelapor wajib mengisi dan menandatangani formulir penerimaan Laporan Dugaan Pelanggaran Pemilu Model B.1-DD yang memuat nama dan alamat pelapor, waktu dan tempat peristiwa terjadi, nama dan alamat terlapor, nama dan alamat saksi-saksi, uraian kejadian, dan tanda tangan pelapor; (2) Menyertakan fotokopi Kartu Tanda Penduduk (KTP) atau kartu identitas lain. Bentuk laporan dugaan pelanggaran tersebut dapat dilakukan secara langsung ke kantor bawaslu atau secara tidak langsung dalam bentuk pesan singkat melalui telepon genggam, faksimili, surat elektronik, atau laporan di situs *web*/*website*.

Saat ini sudah banyak aplikasi yang dibuat untuk Pemilu, aplikasi-aplikasi tersebut dapat ditemukan di pemiluapps.org/aplikasi. Salah satu aplikasi yang menjadi perhatian adalah Pantau Pemilu. Dalam aplikasi Pantau Pemilu terdapat beberapa menu yaitu berita, peraturan, data kandidat, *polling*, pantau seputar Pemilu, grafik statistik hasil pantauan dari pengguna, data rekapitulasi KPU, dan hasil pemilihan di TPS.

Penelitian ini mengembangkan aplikasi *web* untuk pelaporan pelanggaran kampanye yang dilengkapi dengan penanda lokasi terjadinya kampanye serta masyarakat dapat meminta lokasi untuk dilakukan kampanye kepada calon legislatif. Perbedaan aplikasi Pantau Pemilu dengan aplikasi pada penelitian ini adalah pada menu kampanye. Dalam penelitian ini terdapat menu kampanye yang berisi informasi lokasi diadakannya kampanye serta masyarakat dapat melakukan *request* lokasi kampanye. Fokus pada penelitian ini adalah membangun REST APIs dari beberapa fungsi sehingga dapat digunakan pada *client* berbasis *website* dan *mobile*. Kelebihan dari penggunaan REST APIs adalah *client* tidak berhubungan langsung dengan basis data. REST APIs tersebut menjembatani komunikasi antara *client* dan *server*.

**Perumusan Masalah**

Rumusan permasalahan pada penelitian ini yaitu bagaimana mengembangkan aplikasi modul pelaporan kampanye yang mampu memudahkan (1) Masyarakat dalam melaporkan pelanggaran dan mengetahui lokasi kampanye serta melalakukan *request* kampanye; (2) Pemerintah dalam memberikan tanggapan terhadap laporan dari masyarakat..

**Tujuan Penelitian**

Tujuan dari penelitian ini adalah mengembangkan sebuah aplikasi *web* aktivitaskampanye berbasis REST APIs yang memfasilitasi:

1. Masyarakat untuk melaporkan pelanggaran Pemilu yang bersifat anonim.
2. Masyarakat dalam mengetahui lokasi kampanye serta melakukan *request* lokasi diadakannya kampanye di daerah tertentu kepada calon legislatif.
3. Pemerintah dalam mendapatkan dan menanggapi laporan pelanggaran kampanye.

**Manfaat Penelitian**

Manfaat yang diperoleh dari hasil penelitian ini adalah (1) Masyarakat dapat melaporkan pelanggaran Pemilu secara cepat (2) Masyarakat dapat memperoleh informasi seputar calon legislatif; (3) Masyarakat dapat memperoleh informasi lokasi kampanye; (4) Pemerintah menjadi lebih mudah dalam menanggapi laporan pelanggaran pemilu; (5) Pemerintah dapat mengarsipkan data secara *online*.

**Ruang Lingkup Penelitian**

Ruang lingkup penelitian ini ialah (1) Pengembangan modul pelaporan hanya dilakukan pada sisi pengguna masyarakat dan pemerintah berbasis *web* *browser*; (2) Tidak melakukan proses validasi terhadap data yang berasal dari masukan masyarakat; (3) REST APIs penelitian ini menggunakan JSON Web Token yang didapatkan pengguna saat melakukan login; (4) Implementasi pada *web* *browser* hanya untuk mengkonsumsi data dari REST APIs tanpa ada fasilitas untuk *sorting* dan *searching*.

**METODE**

**Data Penelitian**

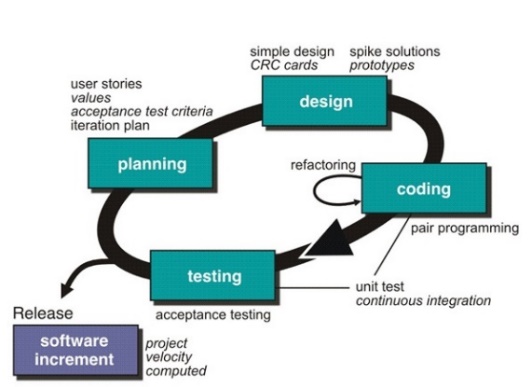
Data yang digunakan dalam penelitian ini terdiri dari data yang berhubungan dengan Pemilu Legislatif seperti data pelanggaran beserta tanggapan pemerintah, data caleg, sebaran lokasi kampanye, dan *request* lokasi penyelenggaraan kampanye. Data pelanggaran dan *request* lokasi penyelenggaraan kampanye diperoleh dari masukan masyarakat. Data caleg diperoleh dari masukan *administrator*.Sebaran lokasi kampanye diperoleh dari masukan calon legislatif daerah. Data yang dimasukkan oleh *user* beracuan pada data Pemilu API *endpoint* Pemilu tahun 2014, sedangkan berita diperoleh dari situs-situs *online* berita seperti Liputan6, Okezone, Tempo, dan Antara News.

**Perancangan Arsitektur Perangkat Lunak**

Perancangan arsitektur perangkat lunak memiliki tiga bagian yaitu data, *client side*, dan *server side*. Pada bagian data terdiri dari basis data dengan struktur MySql atau NoSQL, bagian *Client* terdapat aplikasi berbasis *website* dan *mobile android*, dan pada *server side* terdapat *Google Maps* API dan REST API dengan Node.js, dan pada bagian c*lient* terdapat aplikasi berbasis *website* dengan PHP *Frameworks* Code Igniter 3.0.3 dan *mobile android*.

**Tahapan Penelitian**

Metode yang digunakan dalam penelitian ini adalah metode *Extreme Programming* yang merupakan salah satu model dari *Agile Software Development*. *Extreme Programming* adalah model pengembangan sistem perangkat lunak berbasis *Software Development Life Cycle* (SDLC) yang menyederhanakan berbagai tahapan dalam proses pengembangan sehingga pada Model XP proses menjadi lebih adaptif dan fleksibel. Menurut Mohammadi S *et al*. (2009) *Extreme**Programming* merupakan metode yang paling terkenal karena dalam praktiknya lebih menekankan pada pelanggan sehingga perubahan dapat dengan cepat ditanggapi. Proses pengembangan sistem pada model XP menurut Pressman (2010) terbagi menjadi empat tahapan utama seperti pada Gambar 1.



Gambar 1 Alur pengembangan model Extreme Programming (Pressman, 2010)

Tahapan yang dilakukan dalam penelitian ini adalah:

1. Perencanaan (XP *Planning*)

Tahap ini berfokus untuk mendapatkan gambaran fitur dan fungsi dari perangkat lunak yang akan dibangun. Pada tahap ini akan dibuat *use* case, *user story* dan *fully developed use case description* dari modul masyarakat dan pemerintah.

1. Desain (XP *Design*)

Tahap ini bertujuan untuk mengatur pola logika dari sistem. Desain pada model proses *extreme programming* menjadi panduan dalam membangun perangkat lunak yang didasari pada hasil tahap *planning*. Dalam XP, aktivitas desain terjadi terus-menerus selama proses pengembangan perangkat lunak berlangsung.

1. Pengkodean (XP *Coding*)

Setelah menyelesaikan perencanaan dan desain untuk aplikasi secara keseluruhan, XP lebih memprioritaskan terlebih dahulu membuat modul unit tes yang bertujuan untuk melakukan uji coba setiap modul . Setelah berbagai unit tes selesai dibangun, kemudian dilakukan proses *coding* yang dilakukan permodul. Selanjutnya, modul aplikasi yang sudah selesai dibangun akan digabungkan dengan aplikasi utama.

1. Pengujian (XP *Testing*)

Walaupun pengujian telah dilakukan pada tahap *coding*, namun pada *XP Testing* ini akan mengecek dan memperbaiki semua masalah-masalah yang terjadi.Setelah semua modul telah dikumpulkan dalam sebuah sistem yang sempurna, barulah pengujian penerimaan (*acceptance test*) dilakukan. Pada tahapan pengujian ini aplikasi langsung diuji coba oleh tim secara internal dan mendapat tanggapan langsung mengenai penerapan pada tahap perencanaan yang telah dilakukan sebelumnya.

**HASIL DAN PEMBAHASAN**

**Deskripsi Data**

Data contoh pada penelitian terdiri dari data laporan pelanggaran Pemilu tahun 2014, calon legislatif 2014 dan informasi lokasi kampanye. Data laporan pelanggaran Pemilu tahun 2014 dan calon legislatif 2014 diperoleh dari Pemilu API *endpoint* yang di unduh pada 12 April 2016 pada situs https://github.com/ pemiluAPI /pemilu-data. Data laporan pelanggaran terdapat atribut judul\_ laporan, tanggal\_kejadian, alamat, keterangan, kategori, dan status. Pada penelitian ini atribut yang digunakan hanya judul\_laporan, tanggal\_kejadian, alamat, keterangan, kategori, dan status. Atribut keterangan digunakan untuk mendeskripsikan laporan sedangkan atribut kategori dan status untuk tanggapan pemerintah.

Data calon legislatif terdiri dari 12 atribut yaitu id, nama, alamat, lembaga, nama partai, no urut, jenis kelamin, agama, tanggal lahir, tempat lahir, dapil, dan pendidikan terakhir. Pada penelitian ini hanya diambil 7 calon legislatif dari 29566 calon legislatif yang ada pada Pemilu API. Data informasi lokasi kampanye memiliki 5 atribut yaitu nama\_caleg, lokasi, *latitude*, *longitude*, dan topik\_kampanye.

**Arsitektur Perangkat Lunak**

Penggunaan REST API dilakukan untuk mempermudah aliran data dari *client* dan *server*.REST API juga dapat digunakan oleh bahasa pemograman apapun sehingga dalam pengembangan-nya dapat digunakan oleh aplikasi berbasis *website* dan *mobile*. Penggunaan Google Maps API untuk memberikan penanda lokasi *longitude* dan *latitude* lokasi *request* kampanye, lokasi terjadinya kampanye dan lokasi terjadinya pelanggaran. Dalam implementasi nanti *website browser* sebagai *client* akan menggambil data dari DBMS melalui REST API dengan bahasa pemograman PHP pada *framework* Code Igniter.

**Iterasi Pertama**

1 Tahap Perencanaan (XP *Planning*)

Pada tahap perencanaan ini di buat *user story* dan *fully developed use case description* dari masing-masing *task* utama yang terdapat pada *use case*. *User story* dari *actor* masyarakat dan pemerintah yang diwakili oleh Bawaslu dapat dilihat pada Tabel 1.

Tabel 1 User story melihat data laporan pelanggaran

|  |  |  |  |
| --- | --- | --- | --- |
| *Actor* | *Task* | *User story* | *Priority* |
| Masyarakat | Melaporkan pelanggaran | Masyarakat dapat melaporkan pelangaran-pelanggaran yang dilihat saat kampanye berlangsung secara cepat | 1 |
|  | Melihat data laporan pelanggaran | Masyarakat dapat melihat data laporan pelanggaran yang telah dilaporkan oleh masyarakat lainnya serta ingin mengetahui tindakan apa yang dilakukan oleh Bawaslu terkait dengan laporan tersebut | 2 |
|  | Melakukan *request* lokasi kampanye | Masyarakat dapat Calon Legislatif melakukan kampanye di daerah yang masyarakat inginkan | 3 |
| Bawaslu | Memberikan tindakan terhadap laporan pelanggaran | Bawaslu ingin memberikan tindakan terhadap laporan pelanggaran yang telah dilaporkan oleh masyarakat | 1 |

2 Tahap Desain (XP *Design*)

Pada tahap ini dilakukan pembuatan *mock up* dari *task-task* yang terdapat pada *use case*. *Mock up* dibuat agar pembuatan *website* tidak menyimpang dari tujuan awal. Pada tahap ini juga dilakukan perancangan basis data beracuan pada data Pemilu 2014 yang terdapat pada situs API Pemilu. Rancangan basis data pada iterasi pertama ini menggunakan struktur basis dataMySQL.

3 Tahap Pengkodean (XP *Coding*)

Pada tahap pengkodean telah dilakukan proses pembuatan REST API dari fungsi *input* laporan, *request* kampanye, dan memberi tanggapan terhadap laporan dengan struktur *database* menggunakan MySQL. Fungsi REST API yang dibuat yaitu GET, PUT dan POST. Gambar 2 merupakan kode REST dari fungsi melakukan *input* laporan. Pada kode REST ini *method* yang dilakukan adalah POST yang didefinisikan pada kode router.post. *Method* POST digunakan untuk memasukan data ke dalam tabel yang didefinisikan pada kode var table. *Pair programming* pada iterasi ini dilakukan dengan pengkodean yang dilakukan dengan pertemuan rutin dengan tim.

385 router**.**post**(**"/laporan"**,function(**req**,**res**){**

386 **var** query **=** "INSERT INTO ??(??,??,??,??,??,??,??,??) VALUES (?,?,?,?,?,?,?,?)"**;**

387 **var** table **=** **[**"t\_laporan"**,**"id\_masyarakat"**,**"id\_caleg"**,**"pesan"**,**"bukti"**,**"parent"**,**"latitude"**,**"longitude"**,**"tanggal"**,**req**.**body**.**id\_masyarakat**,**req**.**body**.**id\_caleg**,**req**.**body**.**pesan**,**req**.**body**.**bukti**,**req**.**body**.parent,**req**.**body**.**latitude**,**req**.**body**.**longitude**,**req**.**body**.**tanggal**];**

388 query **=** mysql**.**format**(**query**,**table**);**

389 connection**.**query**(**query**,function(**err**,**rows**){**

390 **if(**err**){**

391 res**.**json**({**"Error" **:** **true,** "Message" **:** "Error executing MySQL query"**});**

392 **}else{**

393 res**.**json**({**"Error" **:** **false,** "Message" **:** "Laporan Added !"**});**

394 **}**

395 **});**

396 **});**

Gambar 2 Kode REST fungsi input laporan dengan struktur database MySQL

4 Tahap Pengujian (XP *Testing*)

Tahap pengujian iterasi pertama dilakukan pada aplikasi Postman. Pengujian ini bertujuan untuk memeriksa apakah REST API berfungsi atau tidak. Pengujian iterasi pertama ini dilakukan oleh penulis dan hasil dari pengujian rest yang diperoleh pada tahap pengkodean berfungsi dengan baik. Pengujian yang dilakukan adalah REST API fungsi GET, POST, DELETE dan PUT.

**Iterasi Kedua**

1 Tahap Perencanaan (XP *Planning*)

Pada tahap perencanaan ini juga di buat *swimlane* diagramyang berfungsi untuk mengetahui aktor yang terlibat dalam suatu *task*. Dari semua *task* utama hanya *task* memberikan tindakan terhadap laporan pelanggaran saja yang melibatkan dua aktor yaitu masyarakat dan Bawaslu yang dapat dilihat pada Gambar 3. Pada iterasi ini terjadi perubahan struktur basis data dari MySQL menjadi NoSQL yaitu MongoDB. Perubahan struktur basis data ini dilakukan karena MongoDB lebih efisien dan performa yang ditawarkan lebih cepat dibandingkan MySQL.



Gambar 3 *Swimlane* diagram memberikan tindakan terhadap laporan pelanggaran

2 Tahap Desain (XP *Design*)

Tahap desain iterasi kedua dilakukan pembaruan *mockup* yang sesuai dengan *fully developed use case description* terbaru. Pada iterasi ini juga dilakukan perancangan tampilan halaman untuk *user* masyarakat dan pemerintah yang dapat dilihat pada Gambar 4. Menu navigasi pada pemodelan ini ditandai dengan persegi panjang berwarna oranye ( ) dan *content* ditandai dengan persegi berwarna biru ( ).



Gambar 4 Tampilan halaman beranda user masyarakat dan pemerintah

3 Tahap Pengkodean (XP *Coding*)

Tahap pengkodean iterasi kedua adalah pembuatan REST API menggunakan struktur basis data NoSQL atau MongoDB. Pengkodean REST API pada struktur *database* MySQL berbeda dengan REST API pada struktur *database* MongoDB. Pengkodean REST API pada struktur basis dataMongoDB dapat dilihat pada Gambar 5. *Method* yang dilakukan pada kode Gambar 5 adalah POST. Proses POST ini dapat dilakukan jika *request body* atau *field* dari *schema* laporantelah diisikan oleh *user*. Jika *request body* telah terpenuhi maka proses selanjutnya adalah menyimpan data dari *request body* ke dalam *schema* atau tabel laporan.

389 apiRoutes**.**post**(**'/laporan'**,** requireAuth**,** **function(**req**,** res**)** **{**

390 console**.**log**(**req**.**body**);**

391 **if(!**req**.**body**.**id\_user **||** **!**req**.**body**.**nama\_caleg **||** **!**req**.**body**.**judul\_laporan **||** **!**req**.**body**.**pesan **||** **!**req**.**body**.**lokasi **||** **!**req**.**body**.**lat **||** **!**req**.**body**.**lng **||** **!**req**.**body**.**tanggal**)** **{**

392 res**.status(**400**).**json**({** success**:** **false,** message**:** 'Please enter data.' **});**

393 **}** **else** **{**

394 **const** newLaporan **=** **new** Laporan**({**

395 id\_user**:** req**.**body**.**id\_user**,**

396 nama\_caleg**:** req**.**body**.**nama\_caleg**,**

397 judul\_laporan**:** req**.**body**.**judul\_laporan**,**

398 pesan**:** req**.**body**.**pesan**,**

399 bukti**:** 'http://localhost:4040/'+req.file.originalname

400 lokasi**:** req**.**body**.**lokasi**,**

401 lat**:** req**.**body**.**lat**,**

402 lng**:** req**.**body**.**lng**,**

403 tanggapan**:**"Belum ada tanggapan"**,**

404 tanggal**:** req**.**body**.**tanggal

405 **});**

406 // Attempt to save the user

407 newLaporan**.**save**(function(**err**)** **{**

408 **if** **(**err**)** **{**

409 **return** res**.status(**400**).**json**({** success**:** **false,** message**:** 'That laporan already exists.'**});**

410 **}**

411 res**.status(**201**).**json**({** success**:** **true,** message**:** 'Successfully created laporan.' **});**

412 **});**

413 **}**

414 **});**

Gambar 5 Kode REST API meng-input laporan dengan struktur database NoSQL

Peng-*input*-an data pada REST API menggunakan Node.js dan hasil dari data tersebut berupa format teks JSON (JavaScript Object Notation) seperti pada Gambar 6. *Array* JSON pada Gambar 6 adalah laporan yang memiliki satu objek atau satu data laporan dengan 11 *field* yaitu \_id, id\_user, nama\_caleg, judul\_laporan, pesan, bukti, lokasi, lat, lng, tanggapan dan tanggal. Pada iterasi ini juga dilakukan implementasi dari fungsi *REST* yang telah dibuat menjadi *web* dengan menggunakan *framework* Code Igniter. Bahasa pemograman yang digunakan *Asynchronous JavaScript* *and XMLHTTP* (AJAX) agar aplikasi web menjadi lebih interaktif dan responsif serta memiliki kemampuan proses *request* ke *server* yang cepat. *Pair programming* pada iterasi ini dilakukan dengan menggunakan Git, Dengan menggunakan Git, setiap orang dalam sebuah tim dapat melakukan perubahan pada *source-code* tanpa harus takut terjadi bentrok ataupun kesulitan dalam menggabungkan hasil perubahan yang mereka lakukan.

4 Tahap Pengujian (XP *Testing*)

Tahap pengujian iterasi kedua dilakukan pengujian REST API yang menggunakan arsitektur *database* NoSQL dengan *software* Postman. Tabel 3 merupakan pengujian dari fungsi REST API *user* masyarakat dan pemerintah dengan menggunakan *software* Postman yang dilakukan oleh internal tim. Setelah pengujian REST API berhasil tahap selanjutnya adalah pengujian terhadap implementasi REST API pada aplikasi web. Tampilan dari implementasi REST lihat laporan dapat dilihat pada Gambar 7.

Gambar 6 Format teks JSON

"laporan": [

{

"\_id": "575588ac923b446822944ff8",

"id\_user": "57551aff5f09aa181c9fee44",

"nama\_caleg": "KH. Tolhah Nokin",

"judul\_laporan": "Alat Peraga di Dalam Kampus UNJ",

"pesan": "Ditemukan dugaan pelanggaran tindak pidana pemilu karena melakukan kampanye di dalam kampus. Lokasi kampanye tepatnya di dalam kawasan kampus Universitas Negeri Jakarta, Universitas Negeri Jakarta, Rawamangun, Pulogadung, East Jakarta City, Jakarta 13220, Indonesia. Kampanye dilakukan dengan memasang alat peraga. Informasi yang ada dalam alat peraga tersebut adalah foto, lambang partai dan juga informasi nomor urut caleg yaitu Tolhah Nokin, Caleg DPRD DKI dari Partai Keadilan Sejahtera nomor urut 2.",

"bukti": "-",

"lokasi": "Jl. Rawamangun Muka, Rawamangun, Rawamangun, Jakarta Timur, Kota Jakarta TimurPulogadung, Daerah Khusus Ibukota Jakarta, Indonesia",

"lat": "-6.195157699999998",

"lng": "106.87831690000007",

"tanggapan": "-",

"tanggal": "2014-03-03",

"\_\_v": 0

},

Tabel 2 Pengujian REST API dengan software Postman

|  |  |  |
| --- | --- | --- |
| *Actor* | Fungsi REST | Status Pengujian\* |
| Masyarakat | *POST* *Register*  *POST Authenticate* (*login*)  *POST Profile* Masyarakat  *GET All* Laporan  *GET* Laporan *by Id User*  *POST* Laporan  *GET All Request* kampanye  *GET Request* Kampanye *by Id*  *POST Request* Kampanye  *GET All* Kampanye  *GET* Kampanye *by User* (caleg)  *GET Profile* Caleg  *GET Logout* | √  √  √  √  √  √  √  √  √  √  √  √  √ |
| Pemerintah | *POST Authenticate* (*login*)  *POST Profile* Pemerintah  *PUT* Laporan *by Id* (memberi tanggapan)  *GET All* Kampanye  *GET All* Laporan  *GET Profile* Caleg  *GET Logout* | √  √  √  √  √  √  √ |

\*Keterangan: Pengujian dilakukan dengan aplikasi Postman

√ : Berhasil

X : Tidak berhasil



Gambar 7 Tampilan halaman lihat laporan

**KESIMPULAN DAN SARAN**

**Simpulan**

Penelitian ini telah berhasil mengembangkan aplikasi *web* berbasis REST APIs yang dikembangkan dengan *framework* Code Igniter versi 3.0.3. Pembuatan REST APIs menggunakan bahasa pemograman Node.js dan struktur basis dataNoSQL yaitu MongoDB. Aplikasi *web* ini dapat membantu proses kampanye menjadi lebih interaktif antara masyarakat dan calon legislatif dengan adanya data lokasi kampanye. Aplikasi *web* ini juga dapat menjadi fasilitas masyarakat dalam melaporkan pelanggaran yang mereka temui dalam pelaksanaan kampanye Pemilu legislatif sehingga membantu Bawaslu dalam pengawasan dan mendapatkan data laporan pelanggaran secara *online*.

**Saran**

Penelitian ini dapat dikembangkan dengan menambahkan fungsi *sharing* kegiatan kampanye kepada masyarakat agar dapat mengoptimalkan pelaksanaan kampanye Pemilu legislatif secara *online*. Aplikasi *web* ini juga dapat dikembangkan dengan menambahkan teknik pengambilan keputusan agar data laporan yang ditampilkan adalah data sebenarnya dan tidak memiliki unsur *black campaign*.

**DAFTAR PUSTAKA**

[APJII] Asosiasi Penyelenggara Jasa Internet Indonesia. 2014. Profil Pengguna Internet Indonesia [Internet]. [diunduh 2015 November 1]. Tersedia pada: http://www.slideshare.net/internetsehat/profil-pengguna-internet-indonesia-2014-riset-oleh-apjii-dan-puskakom-ui.

[Bawaslu] Badan Pengawas Pemilihan Umum. 2015. Laporan Kinerja Badan Pengawasan Pemilihan Umum 2014 [Internet]. [diunduh 2015 Desember 5]. Tersedia pada: http://www.bawaslu.go.id/sites/defult/files/lap%20kinerja%20

bawaslu%202014.pdf

[Kominfo] Kantor Komunikasi dan Infomatika Kota Bogor. 2013. Penerapan Teknologi Informasi dan Komunikasi untuk Mendukung Pengembangan *E-Government* Pemerintah Kota Bogor [Internet]. [diunduh 2015 Desember 12]. Tersedia pada: http://kominfo.kotabogor.go.id/asset/file/sop/penerapan-tik-mendukung-pengembangan-e-gov-2014---2018.pdf

Mohammadi S, Nikkahan B. Sohrabi S. 2009. Challenges of user involvement in extreme programming projects. *International Journal of Software Engineering and Its Applications*. 3(1): 20.

Pressman RS. 2010. *Software Engineering: A Practitioner's Approach*. Ed ke-7. Newyork (US): Mc Graw Hill.

Ramadhan DA, Nurhardyani Y, Hermadi I. 2015. Analisis dan pengembangan *prototype* aplikasi *web* 2.0 dan media sosial pada kampanye pemilu 2014. Di dalam: *Seminar Nasional Teknologi lnformasi dan Multimedia*; 2015 Februari 6-8; Yogyakarta, Indonesia. Yogyakarta (ID): ISSN. hlm 73−78

[RI] Republik Indonesia. 2007. Undang-Undang Republik Indonesia Nomor 22 Tahun 2007 tentang Penyelenggaraan Pemilu. [Internet]. [diunduh 2016 Juli 21]. Tersedia pada: http://bawaslu.go.id/sites/default/files/regulasi/uu\_22\_2007.pdf

[RI] Republik Indonesia. 2012. Peraturan Badan Pengawasan Pemilihan Umum Nomor 14 Tahun 2012 tentang Tata Cara Pelaporan Dan Penanganan Pelanggaran Pemilihan Umum Anggota Dewan Perwakilan Rakyat, Dewan Perwakilan Daerah, dan Dewan Perwakilan Rakyat Daerah [Internet]. [diunduh 2016 Agustus 1]. Tersedia pada: http://www.rumahpemilu.com/public/doc/ 2013\_02\_12\_01\_03\_22\_Perbawaslu%20Nomor%2014%20Tahun%202012%20ttg%20Tata%20Cara%20Pelaporan%20dan%20Penanganan%20Pelanggaran%20Pileg.pdf

[RI] Republik Indonesia. 2015. Undang-Undang Republik Indonesia Nomor 1 Tahun 2015 tentang Penetapan Peraturan Pemerintah Pengganti Undang-Undang Nomor 1 Tahun 2014 tentang Pemilihan Gubernur, Bupati, dan Walikota Menjadi Undang-Undang. [Internet]. [diunduh 2016 Juli 11]. Tersedia pada: http://www.dpr.go.id/dokjdih/ document/uu/1636.pdf

1. Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor, Bogor 16680

   \*Mahasiswa Program Studi Ilmu Komputer, FMIPA-IPB; Surel: indriyani.cs49@gmail.com [↑](#footnote-ref-1)