

RANCANG BANGUN APLIKASI PELAPORAN PENYALURAN GAS LPG 3KG PADA PANGKALAN GAS LPG.

PROPOSAL

Diajukan untuk melengkapi tugas-tugas dan
memenuhi syarat-syarat guna pelaksanaan penelitian Tugas Akhir

Oleh:

ANDRI DARNIUS
1608107010057



**JURUSAN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SYIAH KUALA
DARUSSALAM, BANDA ACEH
AGUSTUS, 2018**

PENGESAHAN PROPOSAL

RANCANG BANGUN APLIKASI PELAPORAN PENYALURAN GAS LPG 3KG PADA PANGKALAN GAS LPG.

Oleh:

Nama : Andri Darnius
NPM : 1608107010057
Jurusan : Informatika

Menyetujui:

Pembimbing I

Pembimbing II

Rahmad Dawood, S.Kom, M.Sc.
NIP. 197203181995121001

Kurnia Saputra, S.T., M.Sc.
NIP. 198003262014041001

Mengetahui:

Ketua Jurusan Informatika FMIPA
Universitas Syiah Kuala,

Dr. Muhammad Subianto, S.Si., M.Si
NIP. 196812111994031005

KATA PENGANTAR

Syukur Alhamdulillah penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan rahmat-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul **“Rancang Bangun Aplikasi Pelaporan Penyaluran Gas Lpg 3 Kg Pada Pangkalan Gas Lpg”**. Shalawat dan salam tak lupa pula penulis sanjungkan kepada junjungan kita Nabi Muhammad SAW yang telah menerangi alam ini dengan cahaya keislaman dan ilmu pengetahuan.

Tugas Akhir ini merupakan salah satu syarat yang harus dipenuhi untuk memperoleh gelar Sarjana Komputer di Jurusan Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Syiah Kuala. Penulis menyadari bahwa penulisan Tugas Akhir ini tidak terlepas dari bantuan dan dorongan dari berbagai pihak, baik secara moril maupun materil. Pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Bapak Dr. Muhammad Subianto, S.Si, M.Si, selaku Ketua Jurusan Informatika Fakultas MIPA Universitas Syiah Kuala.
2. Bapak Irvanizam, S.Si, M.Sc, selaku Dosen Wali.
3. Bapak Rahmad Dawood, S.Kom, M.Sc., selaku dosen pembimbing I dan Bapak Kurnia Saputra, S.T, M.Sc, selaku dosen pembimbing II yang telah banyak memberikan bimbingan dan arahan kepada penulis dalam penyelesaian Proposal Penelitian Tugas Akhir ini.,
4. Seluruh Dosen di Jurusan Informatika Fakultas MIPA, yang tidak bisa disebutkan satu-satu, atas ilmu dan didikannya selama perkuliahan,
5. Staf administrasi Jurusan Informatika yang telah membantu proses administrasi.
6. Dengan tidak mengurangi rasa hormat penulis ucapkan terima kasih sebesar-besarnya kepada Ayahanda Saya Alm. Darnius dan Ibunda saya Ida Nursanti yang tak henti-henti mendukung, memberikan motivasi dan senantiasa mendoakan penulis dari awal masa studi hingga saat ini.
7. Sahabat seperjuangan Riki dan Yusran
8. Seluruh teman-teman jurusan Informatika Unsyiah angkatan 2014 sampai 2016 yang telah memberikan semangat kepada penulis dalam menyelesaikan Tugas Akhir ini.

Dengan segala kerendahan hati, penulis mengharapkan semoga tulisan ini dapat bermanfaat bagi para pembaca, peneliti dan bagi perkembangan ilmu pengetahuan.

Banda Aceh, Agustus 2018

Penulis

DAFTAR ISI

PENGESAHAN PROPOSAL	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR TABEL	vi
DAFTAR GAMBAR	1
BAB I PENDAHULUAN	2
1.1. Latar Belakang	2
1.2. Rumusan Masalah	4
1.3. Tujuan Penelitian	4
1.4. Manfaat Penelitian	4
BAB II TINJAUAN KEPUSTAKAAN	5
2.1. Struktur Penyaluran Gas LPG 3 Kg	5
2.2. Gas Subsidi LPG 3Kg	5
2.3. Android	6
2.4. Java	7
2.5. Servlet	7
2.6. Google App Engine	8
2.7. Aplikasi Hybrid (<i>Mobile Hybrid App</i>)	8
2.8. Ionic Framework	9
2.9. Apache Cordova	9
2.10. Angular	10
2.11. <i>Extreme Programming</i> (XP)	10
2.11.1. <i>Planning</i>	11
2.11.2. <i>Design</i>	12
2.11.3. <i>Coding</i>	12
2.11.4. <i>Testing</i>	12
2.11.5. Keunggulan <i>Extreme Programming</i>	13
2.12. <i>Test Plan</i>	13
2.13. <i>Usability Testing</i>	15
2.14. <i>Unit Testing</i>	15
BAB III METODOLOGI PENELITIAN	16
3.1. Tempat dan Waktu Penelitian	16
3.2. Alat dan Bahan	16
3.3. Metode Penelitian	17
3.3.1. Analisis Permasalahan	17
3.3.2. Studi Literatur	18
3.3.3. Pengumpulan Data	18
3.3.4. Perancangan dan Pembuatan Sistem	18

DAFTAR TABEL

Tabel 3.1	Jadwal Penelitian.	16
Tabel 3.2	Kebutuhan Non-Fungsional	20

DAFTAR GAMBAR

Gambar 2.1	Struktur Penyaluran Gas LPG	5
Gambar 2.2	Tabung Gas LPG	6
Gambar 2.3	Perbandingan Aplikasi Native & Hybrid	9
Gambar 2.4	Aktivitas Utama XP	11
Gambar 3.1	Diagram Alir Penelitian	17
Gambar 3.2	Diagram Use-Case	19

BAB I

PENDAHULUAN

1.1. Latar Belakang

Gas LPG 3 Kg sekarang telah menjadi kebutuhan utama bagi masyarakat di Indonesia sejak pemerintah melakukan konversi dari minyak tanah ke LPG (*Liquified Petroleum Gas*) 3 kilogram (Kg) yang dilakukan sejak 2007 dan telah mampu memberikan penghematan yang signifikan pada kas negara. Namun sekarang ketersediaan tabung gas LPG subsidi 3 Kg semakin berkurang menyebabkan Pertamina selaku distributor memberlakukan kebijakan kepada Pangkalan Gas untuk wajib mengisi *Logbook* Distribusi Tabung Gas 3 Kg. Pada *Logbook* memiliki seluruh informasi tercatat lengkap mulai dari nama pembeli, alamat, jumlah pembelian, jenis pembelian, dan data agen yang menjadi distributor. Sehingga pembeli yang ingin membeli gas subsidi ini harus menunjukkan kartu keluarga (KK) dan Kartu Tanda Penduduk (KTP) sehingga jumlah tabung yang di beli dapat dibatasi berdasarkan KK.

Dengan diberlakukannya kebijakan baru dalam hal penyaluran gas ini, ada beberapa permasalahan yang muncul seperti penerapan *Logbook* ini masih juga mengalami kendala karena pelaporan gas LPG 3 Kg masih dilakukan secara manual sehingga terkadang banyak data penyaluran gas yang salah atau tidak sesuai dengan stok di pangkalan, dalam hal registrasi pembeli baru masih dilakukan dengan cara mengisi formulir yang sangat banyak sehingga proses registrasi memerlukan waktu yang lama. Memang dengan kebijakan yang baru dapat meminimalisir potensi kecurangan dalam membeli tabung gas LPG 3 Kg karena pangkalan akan mengecek nomor KK (Kartu Keluarga) dari si pembeli, namun proses pengecekan tersebut terkadang memakan banyak waktu dan ada potensi terjadinya kesilapan pada saat pengecekan. Hal ini tentu saja akan menghambat pekerjaan dan membutuhkan waktu yang relatif lama sehingga seharusnya ada suatu pembaharuan sistem yang dilakukan sebagai langkah antisipasi.

Berdasarkan permasalahan tersebut, dibutuhkan sebuah aplikasi yang mampu mendaftarkan penyaluran gas LPG 3 Kg ke masyarakat. Sistem ini menggunakan platform Android untuk menjalankan aktivitas penyaluran tabung gas LPG seperti pembelian tabung, penerimaan tabung dari distributor, dan pelaporan gas 3 Kg. User dalam sistem ini adalah operator pangkalan gas LPG dan agen. Sedangkan, pembeli akan menggunakan KTP (Kartu Tanda Penduduk) sebagai alat identifikasi saat melakukan pembelian tabung dan operator akan melakukan verifikasi terhadap KTP dan melakukan pencatatan penjualan. Operator juga dapat melakukan register pembeli baru via aplikasi Android. Kebutuhan utama dari pengguna aplikasi ini

adalah informasi penyaluran gas LPG 3 Kg dapat diakses secara cepat, mudah dan akurat dari perangkat mobile dengan layanan internet di dalamnya. Untuk mendukung kebutuhan akses informasi tersebut, dibutuhkan suatu aplikasi yang inovatif. Sistem operasi Android menawarkan kemampuan untuk membangun aplikasi yang inovatif serta bersifat open source.

Android digunakan sebagai platform pada aplikasi ini karena beberapa alasan salah satunya adalah Android merupakan OS (*operating system*) yang paling disukai oleh banyak orang dengan harga yang tidak terlalu tinggi. Android juga memiliki kelebihan dalam hal *multitasking*, karena dapat menghubungkan dan mengerjakan beberapa aplikasi sekaligus. Android juga menyediakan fitur customisasi yang dapat memodifikasi smartphone sesuka hati. Dalam pengembangannya, aplikasi Android menggunakan bahasa pemrograman java yang di nilai oleh beberapa pengembang (*developer*) aplikasi *mobile* sebagai bahasa pemrograman yang kaku.

Namun saat ini, pengembangan aplikasi Android telah banyak mengalami kemajuan. Hadirnya Ionic yaitu *framework* yang dikhususkan untuk membangun aplikasi mobile hybrid dengan HTML5, CSS dan AngularJS. Ionic menggunakan Node.js, SASS, dan AngularJS sebagai engine-nya. Ionic dilengkapi dengan komponen-komponen CSS seperti *button*, *list*, *form*, *grids*, *tabs*, dan masih banyak lagi. Sehingga Ionic merupakan sebuah teknologi web yang bisa digunakan untuk membuat suatu aplikasi *mobile*. Karena hybrid maka aplikasi hanya dibuat sekali tetapi sudah bisa dirilis di lebih dari satu *platform* alias *cross-platform*.

Kebanyakan orang menginginkan suatu kemudahan dalam penggunaan aplikasi. Kemudahan dalam menggunakan aplikasi dapat diukur dengan adanya pengujian tingkat *usability*. Pengujian *usability* bertujuan untuk mengevaluasi produk aplikasi dengan secara langsung mengujinya kepada pengguna. *Usability testing* memiliki beberapa teknik evaluasi yang berbeda yaitu teknik *Think-Aloud*, *Shadowing Method*, *Coaching Method*, *Question Asking*, *Protocol*, *Teaching Method*, *Performance Measurement*, *Remote Testing* dan *Eye Tracker*. Diantara beberapa teknik tersebut terdapat teknik dengan mengukur performa keberhasilan dan kecepatan pengerjaan *task*(perintah) yaitu teknik *Performance Measurement*. Secara umum *usability* mengacu pada metode bagaimana meningkatkan kemudahan dalam menggunakan suatu produk selama proses desain (Nielsen, 2012). Hasil dari pengukuran tingkat *usability* nantinya bisa digunakan untuk mengembangkan aplikasi agar lebih baik dari sebelumnya.

Aplikasi ini diharapkan memiliki nilai *usability* yang baik, agar pengguna bisa dengan mudah menggunakan aplikasi dan dapat melayani kebutuhan pengguna dengan baik. Oleh karena itu fokus dari penelitian ini adalah untuk memberikan informasi penyaluran tabung gas LPG 3 Kg kepada pangkalan gas melalui

smartphone berbasis Android. Diharapkan aplikasi ini dapat membantu pihak pangkalan dan agen gas LPG untuk menyalurkan gas LPG subsidi 3 Kg.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, ada beberapa permasalahan yang akan dirumuskan pada penelitian ini, diantaranya:

1. Pangkalan Gas masih melakukan pendataan terhadap penyaluran tabung gas LPG 3 Kg secara manual.
2. Bagaimana pangkalan dapat melakukan pendataan tabung gas LPG 3 Kg yang terjual dengan mudah dan cepat.
3. Bagaimana agen gas LPG 3 Kg dapat melakukan pelaporan penyaluran gas LPG 3 Kg dengan mudah dan cepat.
4. Bagaimana memanfaatkan smartphone berbasis Android untuk melakukan pendataan dan pelaporan penyaluran tabung LPG 3 Kg.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang dan membuat aplikasi berbasis Mobile Hybrid yang dapat digunakan untuk mendata penyaluran tabung gas 3 Kg.
2. Menerapkan metode User Centered Design (UCD) dalam merancang aplikasi.
3. Memberikan fasilitas yang dapat menampilkan data penyaluran tabung gas LPG 3 Kg dan mencetak hasil pelaporan via perangkat *mobile*.
4. Menggunakan dan mengimplementasikan Google Cloud sebagai *Web service*.

1.4. Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Diharapkan aplikasi ini dapat membantu dalam semua proses terkait penyaluran tabung gas LPG 3 Kg.
2. Diharapkan aplikasi ini menjadi bahan pembelajaran untuk mengembangkan sebuah *mobile application* yang lebih baik dan berguna bagi masyarakat.
3. Diharapkan aplikasi ini dapat mengawasi penyaluran tabung gas 3 Kg dan mempermudah masyarakat yang berhak untuk mendapatkan gas 3 Kg.

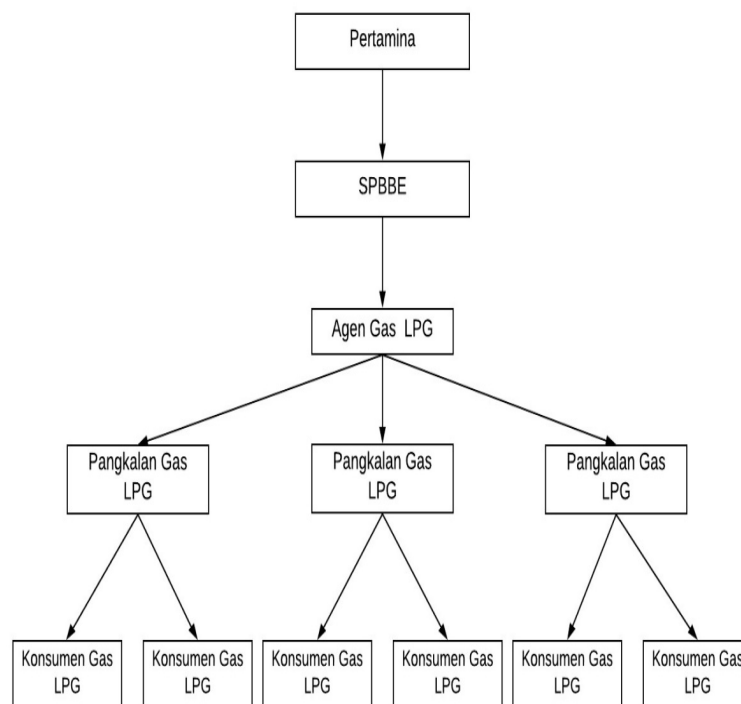
BAB II

TINJAUAN KEPUSTAKAAN

2.1. Struktur Penyaluran Gas LPG 3 Kg

Dalam pendistribusian gas elpiji ke masyarakat, sepenuhnya dilakukan oleh Pertamina dengan sistem *close loop supply chain*, yaitu suatu aliran produk mulai dari konsumen, kembali ke pabrik untuk diproses ulang kemudian kembali lagi ke konsumen sebagai barang baru.

Dalam alur distribusi LPG 3 Kg, yang pertama adalah berasal dari Depot LPG. Kemudian dari Depot LPG, jalur berikutnya disebut SPPBE (Stasiun Pengisian dan Pengangkutan Bulk LPG) yang dikelola oleh Pertamina dan pihak swasta, kemudian setelah itu paket LPG diterima oleh agen LPG dan selanjutnya sebagai ujung tombaknya disebut sub agen atau pangkalan LPG. Sub agen LPG inilah yang berhubungan langsung dengan pengecer, warung atau juga konsumen. seperti pada 2.1.



Gambar 2.1. Struktur Penyaluran Gas LPG

2.2. Gas Subsidi LPG 3Kg

Sejak Pemerintah melakukan konversi dari minyak tanah ke gas LPG yang dilakukan pada tahun 2007 menyebabkan kebutuhan untuk tabung gas LPG semakin meningkat dan gas subsidi LPG 3 kg yang ditawarkan pemerintah di awal periode

konversi, semakin berkurang. Ini dikarenakan jenis tabung ini dianggap lebih ekonomis di bandingkan dengan tabung gas dan banyak rumah tangga dan industri rumahan kecil yang masih memakai tabung gas jenis ini sebagai alat untuk memasak.



Gambar 2.2. Tabung Gas LPG

Maka dari itu Pertamina selaku distributor telah memberlakukan kebijakan kepada konsumen yang ingin membeli gas subsidi ini harus menunjukkan kartu keluarga (KK) dan Kartu Tanda Penduduk (KTP) sehingga jumlah tabung yang di beli dapat dibatasi berdasarkan KK.

2.3. Android

Android merupakan sistem operasi yang dibangun untuk perangkat mobile. Komponen-komponen dari sistem operasi Android ditulis dengan bahasa pemrograman C atau C++, akan tetapi aplikasi pengguna yang digunakan untuk Android ditulis dalam bahasa pemrograman Java (Ableson dkk., 2012). Android juga dapat diartikan sebagai sistem operasi perangkat seluler berbasis Linux yang menyediakan run time environment yang disebut dengan *Android Runtime* (ART) yang telah dioptimasi untuk perangkat dengan sistem memori yang kecil. Karena Android merupakan platform open source, maka setiap orang bebas untuk membuat dan mengembangkan suatu aplikasi (Supardi, 2011).

2.4. Java

Java pertama kali diluncurkan pada tahun 1995 sebagai bahasa pemrograman umum (*general purpose programming language*) dengan kelebihan dia bisa dijalankan di web browser sebagai applet. Sejak awal, para pembuat Java telah menanamkan visi mereka ke dalam Java untuk membuat piranti-piranti yang ada di rumah (*small embedded customer device*) seperti TV, telepon, radio, dan sebagainya agar dapat berkomunikasi satu sama lain. Karakteristik dari bahasa Java adalah sebagai berikut : (Utama, 2002)

- Memiliki struktur Syntax yang sederhana
- Sangat berorientasi objek (OOP) dengan implementasi yang sangat baik sehingga kita bukan hanya belajar bagaimana membuat program yang baik (reusable, scalable, dan maintainable) tetapi juga kita belajar bagaimana cara berfikir yang baik untuk mengenali struktur masalah yang sedang kita hadapi.
- *OpenPlatform, Write Once Run Anywhere* (WORA), portabel atau *multi platform*, program yang kita buat dapat dijalankan di Windows, Linux/Unix, Solaris, dan MacIntosh tanpa perlu diubah maupun di kompilasi ulang.
- Arsitekturnya yang kokoh dan pemrograman yang aman didukung oleh komunitas *Open Source*
- Bukan sekedar bahasa tapi juga platform sekaligus arsitektur. Java mempunyai portabilitas yang sangat tinggi.

2.5. Servlet

Servlet merupakan salah satu bentuk aplikasi berbasis web yang dikembangkan dengan bahasa Java yang berisi berupa class yang digunakan untuk menerima request dan memberi respon melalui protokol http. Servlet memiliki ekstensi .java dan merupakan subclass dari `HttpServlet`. Pada servlet terdapat dua method untuk mengolah request dari client. Kedua method itu adalah `doGet()` dan `doPost()`. (senasama, 2016)

- Method `doGet()` akan dijalankan jika client mengirimkan HTTP request dengan method GET. Contoh dari method GET, adalah jika user menginputkan sejumlah value pada sebuah tampilan website, maka value-value yang dikirimkan itu akan dimunculkan pada alamat URL web browser.
- Method `doPost()` akan dijalankan jika client mengirimkan HTTP response dengan method POST. Contoh dari method POST adalah jika user menginputkan sejumlah value pada sebuah tampilan website, maka

value-value yang dikirimkan itu tidak akan dimunculkan pada alamat URL web browser.

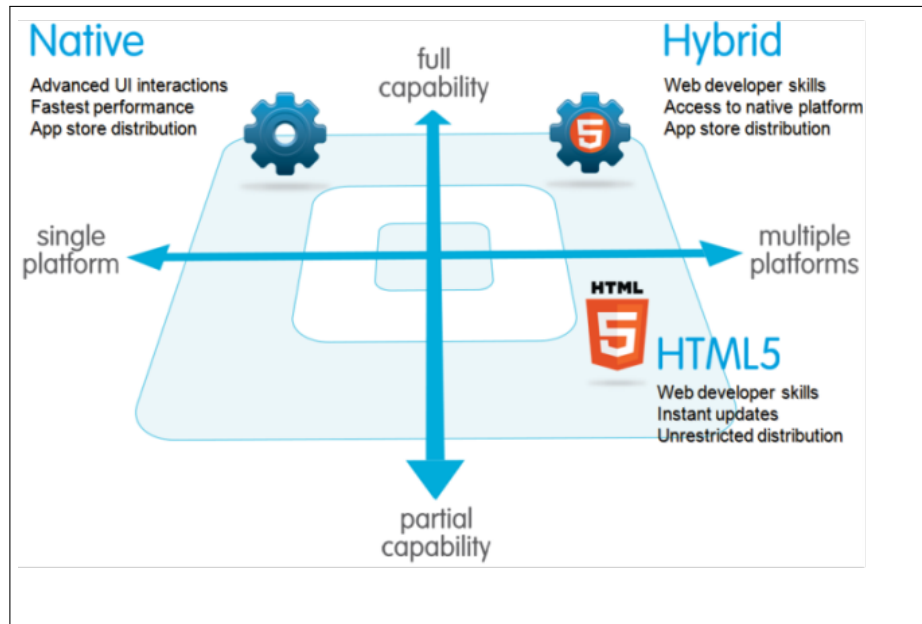
2.6. Google App Engine

Google App Engine (GAE) adalah layanan untuk mengembangkan dan hosting aplikasi Web di pusat data Google, yang termasuk ke dalam komputasi awan kategori Platform As Services (PaaS). Aplikasi web yang di jalankan pada GAE dijalankan di beberapa server untuk redundansi dan memungkinkan penskalaan sumber daya sesuai dengan persyaratan trafik tertentu. App Engine secara otomatis akan mengalokasikan sumber daya tambahan ke server untuk mengakomodasi peningkatan beban. Kelebihan dari GAE adalah sebagai berikut: (technopedia, 2016)

- Server yang tersedia tanpa memerlukan konfigurasi
- Power scaling yang menurunkan hingga ke tingkat "gratis" ketika saat penggunaan sumber daya minimal.
- Alat komputasi awan yang otomatis

2.7. Aplikasi Hybrid (*Mobile Hybrid App*)

Aplikasi hybrid adalah aplikasi web yang ditransformasikan menjadi kode native pada platform seperti iOS atau Android. Aplikasi hybrid biasanya menggunakan browser untuk mengijinkan aplikasi web mengakses berbagai fitur di device mobile seperti Push Notification, Contacts, atau Offline Data Storage. Beberapa tools untuk mengembangkan aplikasi hybrid antara lain Phonegap, Rubymotion dan lain-lain (Permana, 2016).



Gambar 2.3. Perbandingan Aplikasi Native & Hybrid

Keuntungan membangun aplikasi hybrid diantaranya pemeliharaan project menjadi semakin mudah jika dibandingkan dengan aplikasi native. Aplikasi hybrid juga, bisa dibangun secara cepat untuk keperluan cross platform dan dana yang bisa menjadi lebih hemat jika dibandingkan dengan native.

2.8. Ionic Framework

Ionic adalah *framework front-end* yang dikhususkan untuk membangun aplikasi *hybrid* dengan HTML5, CSS dan AngularJS. Ionic menggunakan Node.js SASS, AngularJS sebagai *engine*-nya. Ionic dilengkapi dengan komponen-komponen CSS seperti *button*, *list*, *card*, *form*, *grids*, *tabs*, dan masih banyak lagi. Jadi Ionic itu merupakan teknologi web yang bisa digunakan untuk membuat suatu aplikasi *mobile*. Karena *hybrid* maka aplikasi hanya dibuat satu kali tetapi sudah bisa dirilis di lebih dari satu *platform* dengan kata lain *cross-platform* (Wahyuni, 2017).

2.9. Apache Cordova

Seperti yang telah dijelaskan di atas, bahwa Ionic hanya menyediakan *framework Front-end* sedangkan untuk mengubahnya ke dalam *platform* Android dan IOS, Ionic menggunakan Apache Cordova. Apache Cordova adalah *platform* untuk membangun aplikasi *mobile native* menggunakan HTML, CSS dan JavaScript. Native mobile application yang didukung antara lain Android, iOS, Windows Phone dan Blackberry.

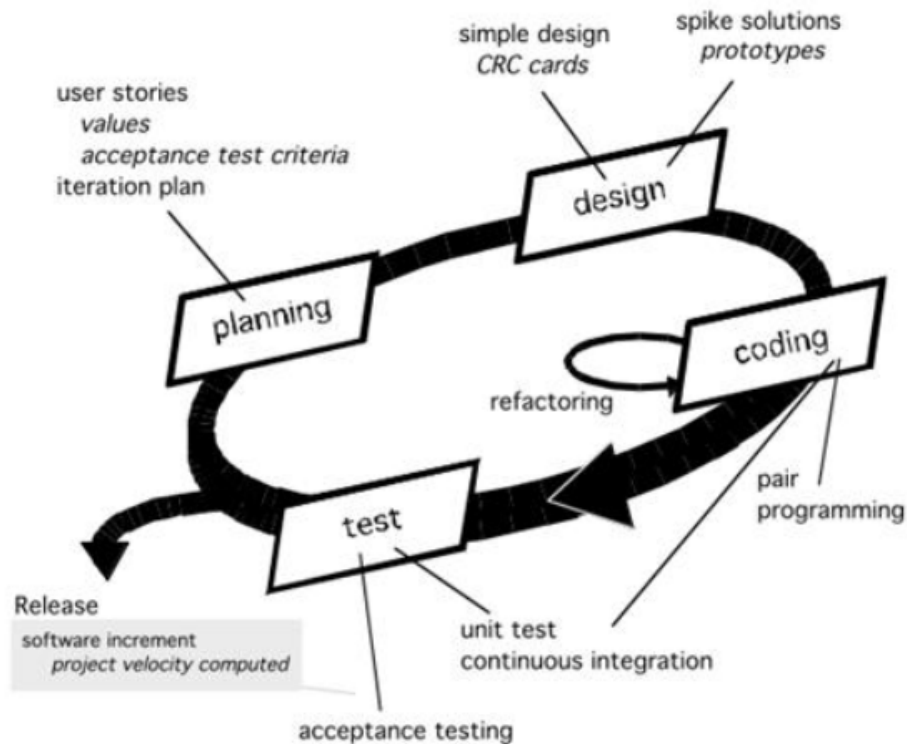
Apache Cordova berisi sekumpulan API (*Application Programming Interface*) untuk mengakses device dari perangkat mobile. Device itu antara lain kamera, GPS (*Global Positioning System*), storage dan lain-lain. Dengan menggunakan UI (*User Interface*) framework seperti jQuery Mobile, Dojo Mobile atau Sencha Touch, maka kita dapat mengakses API ini. Dengan kata lain kita dapat membangun aplikasi hanya menggunakan HTML, CSS dan Javascript.

2.10. Angular

Untuk melakukan implementasi logika, Ionic menggunakan teknologi framework javascript bernama Angular yang menawarkan performa dan respon cepat seperti aplikasi *native*. sebelumnya dikenal dengan nama AngularJS, sekarang dikenal dengan nama Angular (tanpa JS dibelakang). Angular yang merupakan versi terkini dari AngularJS tentu masih banyak peminatnya di dunia pemrograman. Angular telah mengalami banyak sekali perubahan dibandingkan pendahulunya AngularJS. Angular kini sudah side by side dengan framework Javascript modern lainnya seperti React, Vue dan lain-lainnya. Secara konsep Angular sudah lumayan matang, dengan mampu mengakomodir component based dan dengan bergabungnya Typescript milik Microsoft dan RxJS milik ReactiveX untuk mendukung kemapanan framework ini. Performa yang dihasilkan oleh Angular kini bisa disejajarkan dengan para kompetitor dikelasnya.

2.11. Extreme Programming (XP)

eXtreme Programming adalah sebuah model pengembangan sistem yang menyederhanakan berbagai tahapan proses pengembangan agar tercapainya peningkatan efisiensi dan fleksibilitas sebuah proyek pengembangan perangkat lunak. Model *Extreme Programming* ini mengedepankan proses pengembangan yang lebih responsive terhadap kebutuhan klien dibandingkan dengan model-model tradisional lainnya sambil membangun suatu perangkat lunak dengan kualitas yang lebih baik. *Extreme Programming* menawarkan sebuah disiplin baru dalam pengembangan perangkat lunak secara agile. Nilai dasar yang terkandung di dalam eXtreme Programming adalah : komunikasi (*communication*), kesederhanaan (*simplicity*), umpan balik (*feedback*), keberanian (*courage*) dan kualitas kerja (*quality work*). Menurut Pressman dalam bukunya yang berjudul Software Engineering, edisi keenam, proses *Extreme Programming* memiliki kerangka kerja yang terbagi menjadi empat konteks aktivitas utama. Empat konteks tersebut adalah *planning*, *design*, *coding* dan *testing*. Keempat aktivitas inilah yang akan menghasilkan sebuah perangkat lunak yang didasari dengan konsep.



Gambar 2.4. Aktivitas Utama XP

Dalam pembuatannya XP memiliki beberapa langkah operasional yang harus dilakukan seperti :

2.11.1. *Planning*

Planning atau perencanaan adalah proses yang dirancang untuk mencapai tujuan tertentu dan pengambilan keputusan untuk mencapai hasil yang diinginkan. Kebutuhan yang dibutuhkan pada tahap ini yaitu:

- Teknik pengumpulan data
- Analisis kebutuhan sistem
- Identifikasi aktor
- Identifikasi use case

Aktifitas planning pada model proses XP berfokus pada mendapatkan gambaran fitur serta fungsi dari perangkat lunak yang akan dibangun. Pada aktivitas ini dimulai dengan membuat kumpulan cerita atau gambaran yang diberikan klien yang kemudian akan menjadi gambaran dasar dari perangkat lunak.

Kumpulan tersebut nantinya dikumpulkan dalam sebuah indeks cerita dimana setiap poin dari indeks tersebut ditentukan prioritasnya untuk dibangun. Anggota tim dari pengembang aplikasi nantinya akan menentukan alur pengembangan aplikasi

dengan terlebih dahulu memulai mengembangkan tugas dengan resiko dan nilai prioritas yang tinggi terlebih dahulu. Dan seluruh tugas akan selesai dalam tenggat waktu dua minggu.

Selama proses pengembangan, klien dapat mengubah, memperkecil, membagi dan membuang setiap rencana dari aplikasi. Tim XP akan mempertimbangkan setiap perubahan yang diajukan client berikutnya akan mengubah setiap rencana dari pengembangan perangkat lunak.

2.11.2. Design

Aktifitas design dalam pengembangan aplikasi bertujuan untuk mengatur pola logika dalam sistem. Sebuah design yang baik dapat mengurangi ketergantungan antar setiap proses pada sebuah sistem. Dengan begitu, jika salah satu fitur pada sistem mengalami kerusakan, tidak akan mempengaruhi sistem secara keseluruhan.

Design pada model proses XP menjadi panduan dalam membangun perangkat lunak yang didasari dari cerita client sebelumnya. Dalam XP, proses design terjadi sebelum dan sesudah aktivitas coding berlangsung. Dimana aktivitas design terjadi secara terus-menerus selama proses pengembangan aplikasi berlangsung.

2.11.3. Coding

Setelah menyelesaikan pengumpulan cerita dan menyelesaikan *design* untuk aplikasi secara keseluruhan, XP lebih merekomendasikan tim untuk terlebih dahulu membuat modul unit tes yang bertujuan untuk melakukan uji coba setiap cerita yang didapat dari klien. Setelah berbagai unit tes selesai dibangun, barulah tim melanjutkan aktivitasnya ke penulisan coding aplikasi. XP menerapkan konsep *pair programming* dimana setiap tugas sebuah modul dikembangkan oleh dua orang programmer. XP beranggapan, 2 orang akan lebih cepat dan baik dalam menyelesaikan sebuah masalah. Selanjutnya, modul aplikasi yang sudah selesai dibangun akan digabungkan dengan aplikasi utama.

2.11.4. Testing

Tahapan uji coba pada XP sudah dilakukan juga pada saat tahapan sebelumnya yaitu *coding*. Pengujian perangkat lunak dimaksudkan untuk menguji semua elemen-elemen perangkat lunak yang dibuat apakah sudah sesuai dengan yang diharapkan. XP menerapkan perbaikan masalah kecil dengan sesegera mungkin akan lebih baik dibandingkan menyelesaikan masalah pada saat akan mencapai tenggat akhir. Oleh karena itu, setiap modul yang sedang dikembangkan akan terlebih dahulu mengalami pengujian dengan modul unit tes yang telah dibuat sebelumnya.

2.11.5. Keunggulan *Extreme Programming*

Keunggulan dari metode pengembangan perangkat lunak *eXtreme Programming* adalah sebagai berikut: (Michael., 2015)

- Meningkatkan kepuasan kepada klien.
- Membangun system dengan lebih cepat.
- Menjalinkan komunikasi yang baik dengan klien.
- Meningkatkan komunikasi dan sifat saling menghargai antar developer.

2.12. *Test Plan*

Test Plan adalah dokumen yang berisi definisi tujuan dan sasaran pengujian dalam lingkup iterasi (atau proyek), item-item yang menjadi target pengujian, pendekatan yang akan diambil, sumber daya yang dibutuhkan dan point untuk diproduksi. Dengan kata lain test plan dapat disebut sebagai perencanaan atau scenario untuk melakukan testing yang akan dilakukan baik oleh expert atau user umum. (Shanardi, 2017)

Tujuan umum membuat test plan secara umum adalah untuk memudahkan developer untuk melakukan testing agar testing yang dilakukan menjadi jelas sehingga hasilnya lebih berguna dan efisien. Tentunya dalam proses pembuatan test plan ini memerlukan beberapa langkah seperti berikut :

- *Test Plan Identifier*

Test Plan Identifier adalah bagian untuk menjelaskan secara singkat mengenai objek yang akan di test. Bisa berupa penjelasan narasi atau berbentuk tabel dengan kategori kategori tertentu. Informasi yang dijelaskan dapat berupa sekilas mengenai subjek testing, nama orang yang bertanggung jawab terhadap testing, penyusun test plan , tanggal dibuat test plan dan tanggal revisi, dan lain-lain.

- *Introduction*

Pada bagian introduction dibuat untuk menjelaskan secara narasi, mengenai testing yang akan dilakukan terhadap suatu objek testing. Bagian Introduction dapat dibuat lebih rinci dengan menambahkan sub bab apabila perlu untuk dibuat. Contoh subbab yang dapat dibuat antara lain :

- *Purpose* : untuk menjelaskan tujuan *testing* secara spesifik.
- *Background* : latar belakang mengapa *testing* dilakukan.
- *Scope* : sejauh mana *testing* dilakukan

- *Definition and acronyms* : Penjelasan mengenai singkatan dan istilah yang ada di dalam dokumen test plan.

- *Test Items*

Bagian test item menjelaskan mengenai daftar komponen komponen dalam objek testing yang akan di test satu per Satu.

- *Features to be tested*

Menjelaskan fitur fitur apa saja yang ada di dalam objek *testing*, namun fitur tersebut tidak akan di test pada saat pelaksanaan *testing* dan disertakan penjelasan singkat mengapa fitur tersebut tidak di test pada saat *testing*.

- *Item pass / fail criteria*

Berisi tentang kriteria-kriteria yang harus dipenuhi sebelum berlanjut ke fase berikutnya contoh :

- Jika suatu item di test sebanyak 10 kali dan 9 kali diantaranya berhasil namun ada 1 dimana benar benar gagal maka *item* tersebut dinyatakan sebagai gagal/ *fail*.
- Jika hasil dari suatu item sama dengan hasil yang diharapkan maka *item* tersebut dinyatakan berhasil/*pass*.
- *System Crash* akan dinyatakan sebagai *fail*,dsb.

- *Testing Task*

Menjelaskan kegiatan testing kepada pihak yang akan melaksanakan kegiatan tersebut.

- *Responsibilities*

Rincian pihak pihak yang akan bertanggung jawab terhadap suatu kegiatan task di dalam serangkaian kegiatan testing yang akan dilaksanakan

- *Schedule*

Ada beberapa tujuan dalam membuat schedule di dalam test plan, antara lain :

- Merincikan tolak ukur waktu pengerjaan *testing*.
- Estimasi waktu yang dibutuhkan untuk setiap *task*.
- Menjadwalkan *testing task* dan *test milestone*.
- Merincikan periode pemakaian *testing resource*. (Shanardi, 2017)

2.13. *Usability Testing*

Usability Testing atau tes produk merupakan metode riset untuk mengembangkan dan menyempurnakan produk baru maupun yang telah ada. Inti dari riset ini adalah mendudukkan pengguna/pelanggan di pusat, lalu mengambil pelajaran dari sana . maka dalam test ini, mutlak adanya pengamatan secara langsung (Nielsen, 2012)

Dengan mendokumentasikan pengalaman aktual para calon pengguna aplikasi/produk dapat dievaluasi , sebab dari uji ini diharapkan akan menangkap kekuatan dan kelemahan dari setiap aspek yang ada pada aplikasi itu sendiri.

2.14. *Unit Testing*

Unit Testing adalah metode verifikasi perangkat lunak di mana programmer menguji suatu unit program layak untuk tidaknya dipakai. Unit testing ini fokusnya pada verifikasi pada unit yang terkecil pada desain perangkat lunak (komponen atau modul perangkat lunak). Karena dalam sebuah perangkat lunak banyak memiliki unit-unit kecil maka untuk mengujinya biasanya dibuat program kecil atau main program) untuk menguji unit-unit perangkat lunak. Unit-unit kecil ini dapat berupa prosedur atau fungsi, sekumpulan prosedur atau fungsi yang ada dalam satu file jika dalam pemrograman terstruktur, atau kelas, bisa juga kumpulan kelas dalam satu package. (Feridi, 2016)

BAB III

METODOLOGI PENELITIAN

3.1. Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Jurusan Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Syiah Kuala Banda Aceh. Waktu yang diperlukan untuk melakukan penelitian ini kurang lebih selama lima bulan, yang dimulai dari bulan Juni 2018 sampai bulan Desember 2018.

Tabel 3.1. Jadwal Penelitian.

No	Keterangan	Bulan					
		Jul	Agt	Sep	Okt	Nov	Des
1	Studi literatur						
2	Penulisan Proposal						
3	Pengembangan Aplikasi						
4	Evaluasi Sistem						
5	Penulisan Laporan Akhir						

3.2. Alat dan Bahan

Alat yang digunakan pada penelitian ini adalah sebagai berikut :

a. Perangkat Keras (*Hardware*)

- 1 unit Laptop HP Intel(R) Core(TM) i7 6700HQ
- RAM 8 GB DDR3
- Harddisk 1 TB
- 1 unit Android Samsung S7

b. Perangkat Lunak (*Software*)

- Visual Studio Code
- Cordova 7.0.0
- Ionic Framework 3
- SDK Android 27.0.1

3.3. Metode Penelitian

Skema dari alur tahapan penelitian dapat dilihat pada Gambar 3.1



Gambar 3.1. Diagram Alir Penelitian

3.3.1. Analisis Permasalahan

Aplikasi ini merupakan aplikasi berbasis mobile yang berguna untuk membantu pangkalan gas LPG Subsidi 3 Kg dalam melakukan pelaporan penyaluran tabung gas subsidi. Aplikasi ini juga membantu agen gas LPG selaku distributor dan pengawasan dari pangkalan gas untuk melakukan rekap laporan penyaluran tabung gas 3 Kg. Aplikasi ini melakukan pelaporan penyaluran tabung gas subsidi dengan menggunakan NIK(Nomor Induk Kependudukan) dan no telepon sebagai data acuan yang valid. Setelah itu data tersebut langsung di kirim langsung ke agen gas LPG bersangkutan. Adapun fitur-fitur yang terdapat pada aplikasi ini adalah sebagai berikut

- Masuk ke dalam aplikasi.
- Melihat jadwal pasokan tabung.
- Mencatat penjualan tabung gas menggunakan NIK sebagai acuan.
- Mencetak kartu kendali untuk konsumen.
- Mengubah profile biodata user.

3.3.2. Studi Literatur

Studi literatur dilakukan dengan mencari jurnal baik nasional maupun internasional, buku, serta beberapa literatur elektronik yang diunduh dari internet yang terkait dengan penelitian ini. Studi literatur juga diperoleh dari penelitian sebelumnya yang berkaitan dengan penelitian. Studi literatur digunakan sebagai bahan referensi selama proses penelitian.

3.3.3. Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah data penyaluran tabung dan format pelaporan yang dipakai. Data tersebut didapatkan dari agen dan pangkalan yang menjadi objek penelitian.

3.3.4. Perancangan dan Pembuatan Sistem

Dalam merancang aplikasi ini, digunakan metode pengembangan perangkat lunak yaitu eXtreme Programming (XP). Metode XP fleksibel terhadap perubahan-perubahan sehingga cocok digunakan pada aplikasi ini. Berikut merupakan tahapan perancangan sistem yang akan dilakukan:

1. *Planning*

Pada tahap ini dikumpulkan kebutuhan awal user atau dalam XP dikenal dengan istilah *user requirement*. Hal ini dibutuhkan agar kebutuhan *output* sistem, dan fitur utama dari aplikasi yang dikembangkan terdefinisi dengan jelas. Selanjutnya dilakukan analisa kebutuhan awal terhadap aplikasi:

a. *User Story* User story adalah deskripsi singkat dan sederhana mengenai fitur atau fungsi dari software, dilihat dari sudut pandang pengguna. Berikut ini merupakan user story dari kasus penelitian ini.

- **ID: 01**

Sebagai pangkalan, saya ingin mencatat penjualan tabung ke dalam laporan logbook.

- **ID: 02**

Sebagai pangkalan, saya ingin mengetahui jadwal pasokan/penerimaan tabung gas

- **ID: 03**

Sebagai pangkalan, saya ingin melakukan login ke dalam aplikasi

- **ID: 04**

Sebagai pangkalan, saya ingin mendaftarkan konsumen baru.

- **ID: 05**

Sebagai agen, saya ingin melakukan rekap laporan dari masing-masing pangkalan milik saya.

- **ID: 06**

Sebagai agen, saya ingin menetapkan kuota tabung per bulan untuk masing-masing pangkalan.

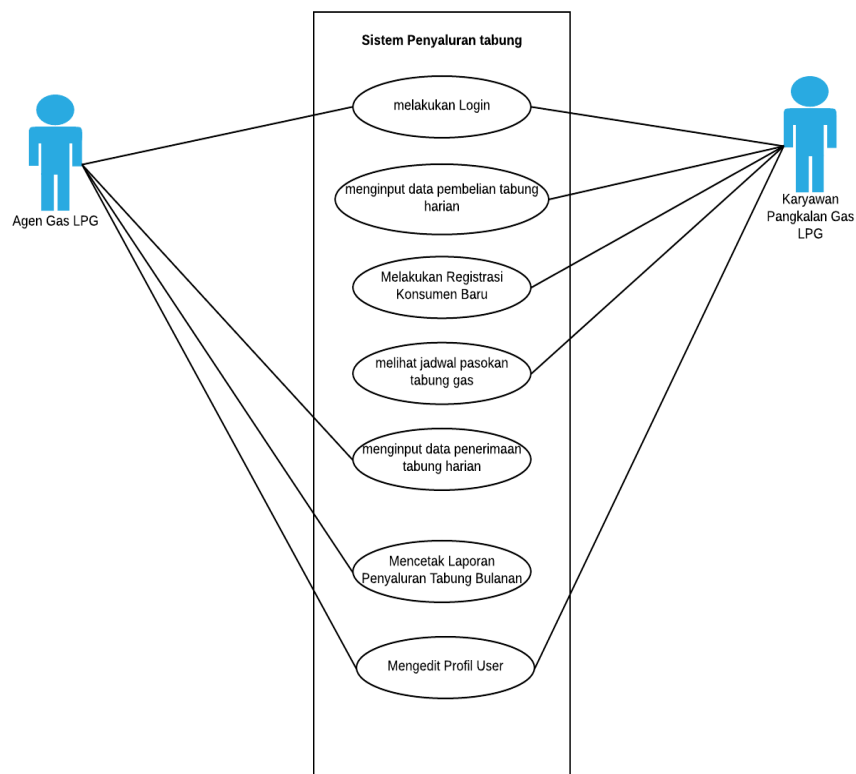
- **ID: 07**

Sebagai pangkalan, saya ingin mencetak kartu kendali konsumen untuk pangkalan kecil yang belum memiliki mesin cetak.

Penggunaan nomor unik ID pada user story dilakukan untuk memudahkan pengelolaan *user story* sekaligus membedakan satu *user story* dengan yang lainnya.

b. Kebutuhan Fungsional

Kebutuhan fungsional, dapat dimodelkan pada use case diagram. Use case diagram adalah diagram yang memodelkan perilaku dari sistem. Di dalam use case diagram terdapat sekumpulan use case, aktor dan hubungan antara use case dan aktor. Use case diagram dapat dilihat pada 3.2 di bawah ini.



Gambar 3.2. Diagram Use-Case

c. Kebutuhan Non Fungsional

Kebutuhan non-fungsional adalah batasan dalam layanan atau fungsi yang ditawarkan sistem. Dalam penelitian ini, kebutuhan non-fungsional yang digunakan ada dua macam, seperti pada Tabel 3.2.

Tabel 3.2. Kebutuhan Non-Fungsional

Jenis Kebutuhan	Deskripsi
<i>Usability</i>	Sistem yang dibuat harus dapat dipelajari dan digunakan dengan mudah oleh pengguna. Kemudahan-kemudahan tersebut diantaranya kemudahan dalam hal penggunaan sistem.
<i>Compatibility</i>	Sistem dapat diakses melalui beberapa versi sistem operasi Android.

2. *Design*

Tahap ini fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean. Representasi perangkat lunak yang dikembangkan digambarkan dalam sebuah pemodelan. Tahapan perancangan dilakukan berdasarkan dari hasil analisis kebutuhan perangkat lunak pada tahap sebelumnya. Tahapan desain ini meliputi perancangan Unified Modelling Language (UML), perancangan database dan perancangan desain tampilan (user interface).

3. *Coding*

Pada tahapan ini, dilakukan implementasi terhadap desain yang telah dibuat sebelumnya ke dalam bentuk program. Penulisan kode program dalam penelitian ini menggunakan bahasa pemrograman Typescript untuk aplikasi berbasis Android dan bahasa pemrograman PHP untuk web service yang akan digunakan oleh aplikasi untuk berkomunikasi dengan database.

4. *Testing*

Tahapan *testing* yang dilakukan diantaranya adalah sebagai berikut:

a. *Usability*

Pengujian *usability* digunakan untuk mengetahui seberapa mudah aplikasi dapat dijalankan oleh pengguna. Teknik pengujian *usability* yang digunakan adalah dengan memberikan kuesioner kepada setiap kelompok user yaitu agen gas LPG dan pangkalan gas LPG. Jenis pertanyaan yang ada pada kuesioner mengacu pada kuesioner SUS (*System Usability Scale*). Dalam pengujian *usability* ini, melibatkan sekitar 8 pengguna yang berbeda.

b. *Unit Testing*

Unit testing fokus pada verifikasi pada unit yang terkecil pada desain perangkat lunak (komponen atau modul perangkat lunak). Karena dalam sebuah perangkat lunak banyak memiliki unit-unit kecil maka untuk mengujinya biasanya dibuat program kecil atau main program) untuk menguji unit-unit perangkat lunak.

DAFTAR KEPUSTAKAAN

- Ableson, W., Sen, R., dan King, C. . (2012). *Android in Action*. Mobile technology. Manning.
- Feridi (2016). Pentingnya pengujian perangkat lunak - codepolitan.com. Diakses pada tanggal 25 Juni 2018.
- Michael. (2015). Extreme programming – michael – medium. Diakses pada tanggal 17 April 2018.
- Nielsen, J. (2012). Usability 101: Introduction to usability. Diakses pada tanggal 15 Maret 2018.
- Permana, Y. (2016). Ini dia perbedaan aplikasi native, hybrid atau web - codepolitan.com. Diakses pada tanggal 20 April 2018.
- senasama (2016). Pertemuan 5 (servlet) – senasama blog. Diakses pada tanggal 20 Juni 2018.
- Shanardi, A. (2017). Pengertian test plan. Diakses pada tanggal 20 Juni 2018.
- Supardi, Y. (2011). *Semua Bisa Menjadi Programmer Android Basic*. PT Elex Media Komputindo, Jakarta.
- technopedia (2016). What is google app engine (gae)? - definition from techopedia. Diakses pada tanggal 21 Juni 2018.
- Utama, A. (2002). *Berfikir Objek: Cara Efektif Menguasai Java*. IlmuKomputer.Com.
- Wahyuni, A. S. (2017). Apa itu ionic framework - codepolitan.com. Diakses pada tanggal 28 Mei 2018.