**BAB I**

**PENDAHULUAN**

**Latar Belakang Masalah**

Menurut (Laksmi, Gani, & Budiantoro, 2015), Surat merupakan salah satu sarana komunikasi dalam bentuk tertulis atau alat yang digunakan untuk menyampaikan pesan dari pihak pengirim ke penerima. Kegiatan saling berkirim surat dari seseorang kepada orang lain atau dari organisasi kepada organisasi disebut surat-menyurat atau korespondensi. Para pelakunya disebut koresponden.

Universitas Jambi (UNJA) merupakan perguruan tinggi negeri yang perkembangannya sangat pesat. Hal ini terwujud dengan adanya sinergi yang baik diantara setiap bagian rektorat UNJA, yaitu rektor, wakil rektor I (WR-I) bidang akademik, wakil rector II (WR-II) bidang administrasi, kepegawaian dan keuangan dan wakil rektor III (WR-III) bidang kemahasiswaan serta fakultas, program studi dan semua pihak terkait baik pihak dalam maupun luar UNJA. Sinergi yang baik ini diupayakan dengan adanya komunikasi, terutama melalui surat. Pengelolaan surat yang yang dilakukan oleh UNJA berkaitan dengan rektorat, fakultas, program studi, departemen lain, mahasiswa dan pihak eksternal UNJA.

Berdasarkan Surat Edaran Rektor Universitas Jambi, Nomor 2 Tahun 2020 tentang Jenis Tata Naskah Dinas di Lingkungan Universitas Jambi. Ada beberapa jenis surat atau tata naskah dinas yang ada di universitas jambi, diantaranya, naska dinas peraturan (peraturan, surat edaran, prosedur operasional standar administrasi pemerintahan), naskah dinas penetapan (keputusan), naskah dinas penugasan (instruksi, surat perintah dan surat tugas), naskah dinas korespondensi(nota dinas, memo,surat dinas, surat undangan, surat pengantar), naskah dinas khusus (nota kesepahaman, perjanjian kerjasama, surat kuasa, berita acara, surat keterangan, surat pernyataan, pengumuman, laporan dan notula) dan naskah dinas elektronik.

Adapun surat atau naskah-naskah tersebut sudah dibuatkan sistem untuk mempermudah dan memperlancar dalam prosesnya, akan tetapi sistem yang telah ada hanya berbasis website saja, sehingga masih banyak timbul masalah-masalah, seperti keterbatasan waktu dan tempat karena tidak mungkin setiap saat pegawai yang mengurus surat berada di depan komputer, adakalahnya mereka sedang berada dalam perjalanan, sehingga hal tersebut akan menjadi penghambat dalam proses persuratan.

Berdasarkan latar belakang di atas maka dirancang sebuah sistem administrasi persuratan berbasis mobile pada universitas jambi untuk mempermudah dalam melakukan proses persuratan, sehingga akan menjadi lebih efekif dan efisien dalam prosesnya.

**Tujuan**

1. Membuat rancangan sistem informasi pengelolaan persuratan berbasis android berdasarkan hasil analisis sistem.
2. Mengimplementasikan rancangan sistem informasi persuratan yang telah dibangun pada UNJA.
3. Melakukan pengujian sistem informasi persuratan pada UNJA agar sesuai dengan keinginan dari pengguna sistem di UNJA.

**Manfaat**

**BAB II**

**LANDASAN TEORI**

**Sistem Operasi Android**

Android adalah sistemoperasi yang bersifat open source. Sistem operasi ini memiliki lisensi Apache yang sangat terbuka dan bebas. Android awalnya dikembangkan oleh Android Inc., kemudian diakuisisi oleh Google pada tahun 2005 sehingga menjadi salah satu bagian dari produk Google [1]. Sifat open sourceini membuat daya tarik tersendiri bagi para pengembang aplikasi perangkat bergerak untuk mencoba menciptakan aplikasi hasil karyanya. Hal ini membuat sistem operasi Android menjadi sistem operasi yang sangat populer bagi berbagaiprodusen perangkat bergerak untuk meraih konsumen [2].

Menurut Supardi (2014, 2)[3], Android merupakan sebuah sistem operasi perangkat mobile berbasis linux yang mencangkup sistem operasi, middleware, dan aplikasi. Beberapa pengertian lain dari Android, yaitu :

1. Merupakan platform terbuka (Open Source) bagi para pengembang (Programer) untuk membuat aplikasi.
2. Merupakan sistem operasi yang dibeli Google Inc. dari Android Inc.
3. Bukan bahasa pemrograman, tetapi hanya menyediakan lingkungan hidup atau run time environment yang disebut DVM (Dalvik Virtual Machine) yang telah dioptimasi untuk alat/device dengan sistem memori yang kecil
4. Struktur AplikasiAndroid

Struktur aplikasi Android atau fundamental aplikasi ditulis dalam bahasa pemrograman Java. Kode Java dikompilasi bersama dengan resource file yang dibutuhkan oleh aplikasi. Dimana prosesnya di package oleh tools yang dinamakan apt tools ke dalam paket Android. Sehingga menghasilkan file dengan ekstensi apk. File apk ini yang disebut dengan aplikasi, dan nantinya dapat dijalankan pada peralatan mobile. Ada empat komponen pada aplikasi Android, yaitu :

* 1. Activities

Activities merupakan komponen untuk menyajikan tampilan pemakai (user interface) kepada pengguna.

* 1. Service

Service merupakan komponen yang tidak memiliki tampilan pemakai(user interface),tetapi service berjalan secara backgrounds.

* 1. Broadcast Receiver

Broadcast Receiver merupakan komponen yang berfungsi menerima dan bereaksi untuk menyampaikan notifikasi.

* 1. Content Provider

Content Provider merupakan komponen yang membuat kumpulan aplikasi data secara spesifik, sehingga bisa digunakan aplikasi lain.

1. Tools

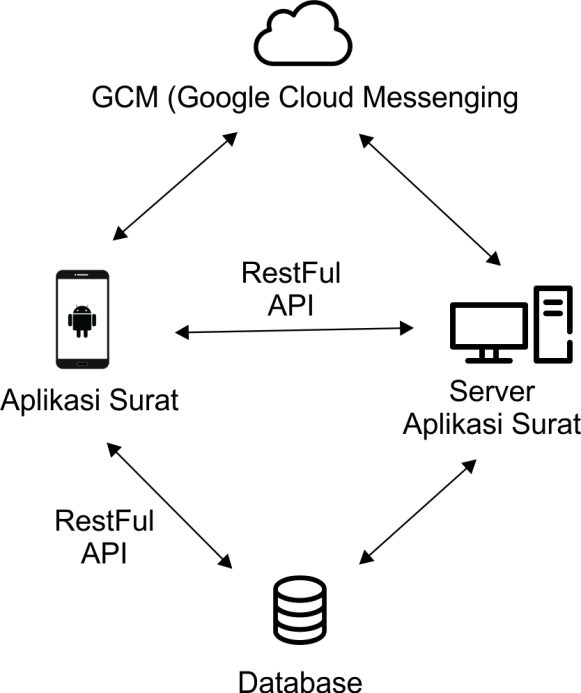
Pembangunan Android Untuk membangun sebuah sistem operasi Android dapat menggunakan Mac, Windows PC, ataupun Linux. Tools yang dibutuhkan gratis dan dapat didownload dari web. Berikut adalah beberapa tools yang digunakan untuk membangun aplikasi android.

* 1. JDK(Java Development Kit)
  2. Android SDK
  3. ADT (Android Development Tools)

***RESTfulAPI***

RESTfulAPI adalah sebuah web service yang diimplementasikan dengan HTTP dan prinsip REST (Representational State Transfer) [4], dan sumber dayanya tersimpan di penyimpanan data. Web service ini digunakan sebagai media pertukaran data antara sisi klien dengan sisi server yang menyimpan basis data. Adapun cara kerja RESTful API, dimana sebuah client mengirimkan sebuah data atau request melalui HTTP Request dan kemudian server merespon melalui HTTP Response.

1. Komponen dari HTTP Request :
   1. Verb, HTTP method yang digunakan misalnya GET, POST, DELETE, PUT dll.
   2. Uniform Resource Identifier (URI) untuk mengidentifikasikan lokasi resource pada server.
   3. HTTP Version, menunjukkan versi dari HTTP yang digunakan, contoh HTTP v1.1.
   4. Request Header, berisi metadata untuk HTTP Request. Contoh, type client/browser, format yang didukung oleh client, format dari body pesan, seting cache dll.
   5. Request Body, konten dari data.
2. Komponen dari HTTP Response :
   1. Status/Response Code, mengindikasikan status server terhadap resource yang direquest. misal : 404, artinya resource tidak ditemukan dan 200 response OK.
   2. HTTP Version, menunjukkan versi dari HTTP yang digunakan, contoh HTTP v1.1.
   3. Response Header, berisi metadata untuk HTTP Response. Contoh, type server, panjang content, tipe content, waktu response, dll
   4. Response Body, konten dari data yang diberikan.



Gambar 1 : Arsitektur Sistem Persuratan

***Google Cloud Messaging(GCM)***

*Google Cloud Messaging (GCM)* adalah sebuah servis yang dikembangkan oleh Google Inc yang dapat membantu para pengguna dalam mengirimkan data dari server ke aplikasi Android. Pesan yang dikirimkan oleh GCM biasanya memberikan informasi mengenai suatu data baru dalam aplikasi. Sebagai contoh ketika ada email baru yang masuk maka akan muncul *notifikasi* ke perangkat Android. GCM servis akan melakukan pengaturan semua hal yang berhubungan dengan antrian pesan atau *notification* dan pengirimannya ke perangkat *mobile* yang menjadi target [5].

Layanan GCM menangani semua aspek antrian pesan dan pengiriman ke aplikasi target Android berjalan pada perangkat target. GCM memungkinkan aplikasi android untuk mengirimkan pesan kepada server untuk membroadcast sebuah notifikasi kepada seluruh *client* yang ada. Hanya membutuhkan sebuah *account* Gmail maka akan langsung dapat menggunakan fasilitas GCM ini.

**BAB III**

**METODELOGI**

Pengembangan aplikasi persuratan pada universitas jambi, baik sistem mobile maupun sistem web ini mengunakan proses *SDLC* (*Software Development Life Cycle*) dalam bentuk model air terjun. Model Air terjun, atau yang kadang dinamakan siklus hidup klasik adalah pendekatan pengembangan perangkat lunak yang sistematis dan berurutan (sekuensial), yang dimulai dengan spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (planning), pemodelan (modelling), konstruksi (construction), serta penyerahan sistem/perangkat lunak ke para pelanggan/pengguna (deployment), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak lengkap yang dihasilkan. [6]

**Analisis Masalah dan Usulan Solusi**

Tahapan pertama yang dilakukan adalah tahapan perencanaan, dimana dalam tahapan ini peneliti mencari dan mengumpulkan kebutuhan secara lengkap yang kemudian dianalisis. Seperti yang telah diketahui bahwa Universitas Jambi merupakan salah satu perguruan tinggi negeri yang terletak di Provinsi Jambi. Sebagai instansi pendidikan tentunya membutuhkan sistem informasi persuratan yang digunakan untuk mengatur kegiatan surat-menyurat agar menjadi lebih cepat dan efektif, serta efisien dan akurat untuk meningkatkan standar mutu pendidikan.

Universitas Jambi telah menggunakan sistem dalam menjalankan proses persuratannya. Sistem yang telah ada sudah baik, akan tetapi sistemnya masih berbasis web (desktop), sehingga sistem tersebut hanya dapat diakses dengan menggunakan perangkat komputer saja, untuk itu diperlukannya sistem yang dapat diakses dengan mudah, kapan dan dimana saja.

## Analisis Kebutuhan Sistem

Analisis kebutuhan sistem sangat diperlukan dalam mendukung kinerja aplikasi, apakah aplikasi yang dibuat telah sesuai dengan kebutuhan atau belum, karena kebutuhan sistem akan mendukung tercapainya tujuan suatu aplikasi.

1. Analisis Kebutuhan Fungsional (Functional Requirement)

Kebutuhan fungsional berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Dari hasil analisis yang telah dilakukan, maka perlu dibuat sistem yang mampu:

* Sistem mampu menampilkan surat masuk
* Sistem mampu menampilkan surat keluar
* Sistem mampu mengirimkan surat

1. Analisis Kebutuhan Non Fungsional (Non Functional Requirements)

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan sistem. Kebutuhan non-fungsional meliputi ketersedian perangkat keras, perangkat lunak dan pengguna. Analisis kebutuhan yang diperlukan dalam perancangan sistem ini antara lain sebagai berikut :

1. Kebutuhan Perangkat Keras

Agar sebuah sistem dapat berjalan dengan baik dan mempunyai kemampuan yang memadai. Perangkat keras yang dibutuhkan dalam pembuatan aplikasi ini adalah :

* PC ( *Personal Computer* ) / Laptop
* Prosessor Intel® Core™ i7
* Ram 8 Gb
* SSD 256 Gb

1. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan mendukung dalam pembuatan dan pengoperasian program aplikasi ini adalah sebagai berikut:

* Sistem operasi Windows 10
* Flutter SDK (Version 10)
* Android Studio (Version 4.0.1)
* Visual Studio Code (Version 1.53.2)
* Android SDK (Version 30.0.3)
* Java SE Development Kit
* Bahasa Dart

1. Kebutuhan Pengguna (User)

Pengguna yang akan berinteraksi secara langsung dengan sistem ini adalah Pegawai Universitas Jambi.

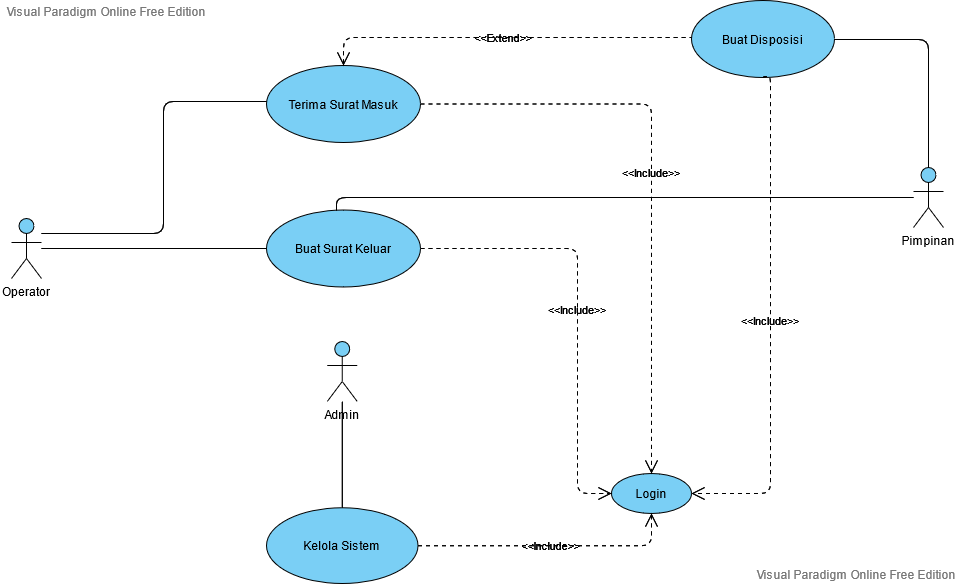
**Model Fungsional Perangkat Lunak**

Pemodelan perangkat lunak menggunakan model Unified Modelling Language (UML). UML didefenisikan sebagai solusi notasi diagram untuk menggambarkan artefak pada perancangan sistem berorientasi objek. Dengan UML kita bisa memvisualisasikan, menetapkan, membuat dan mendokumentasikan aplikasi software kita. Saat sistem software menjadi lebih besar dan lebih kompleks, kita perlu untuk mengatur kekompleksan tersebut dan dalam arti menyederhanakannya sehingga kita bisa lebih mengerti. (Dharwiyanti : 2003)[7].

Seperti bahasa-bahasa lainnya, UML Mendefinisikan notasi dan syntax/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML syntaxmendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Diagram-diagram yang terdapat di dalam pemodelan UML adalah sebagai berikut :

1. Diagram Use Case

Diagram Use Case menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa”yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem. Untuk mengetahui gambaran interaksi antara sistem dan aktor yang terlibar pada sistem yang dibuat, maka digambarkan dengan menggunakan use case diagram sebagai berikut :



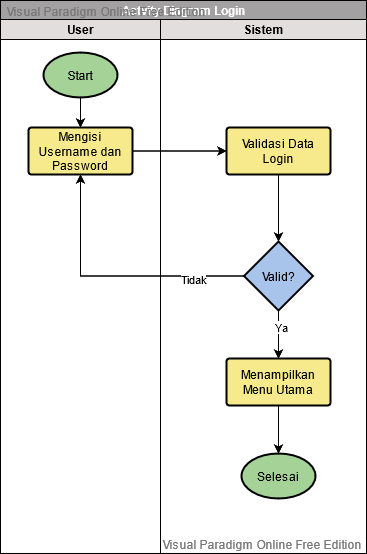
Gambar 2 : Usecase Diagram

1. Diagram Aktivitas

Diagram aktivitas menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Diagram aktivitas juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Berikut adalah diagram aktivitas dari sistem informasi surat masuk dan surat keluar yang dibuat:

* Diagram Aktivitas Login

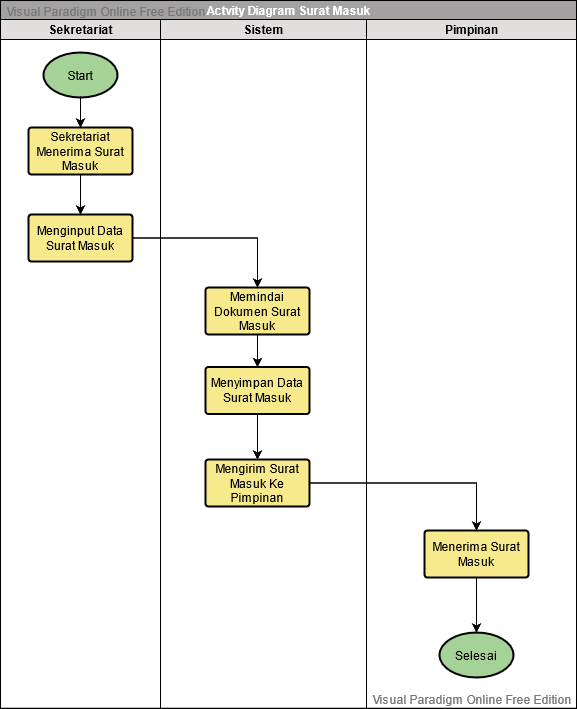
Diagram aktivitas login menjelaskan aliran kerja aktor pengguna pada proses login. Diagram aktivitas login dapat dilihat pada gambar berikut ini :



Gambar 3 : Diagram Aktivitas Login

* Diagram Aktivitas Terima Surat Masuk

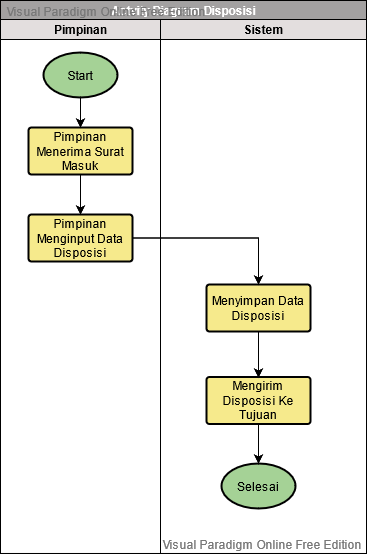
Diagram aktivitas terima surat masukmenjelaskan aliran kerja aktor pengguna pada proses terima surat masuk. Diagram aktivitas terima surat masuk dapat dilihat pada gambar berikut ini :



Gambar 4 : Diagram Aktivitas Terima Surat Masuk

* Diagram Aktivitas Buat Disposisi

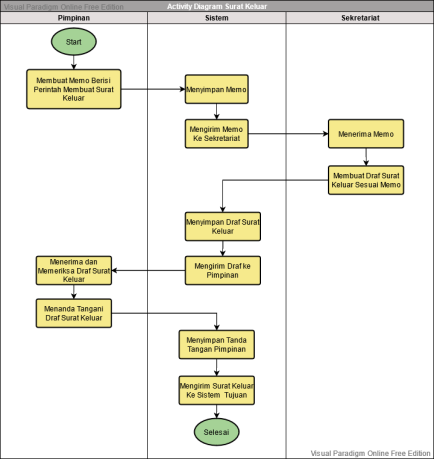
Diagram aktivitas buat disposisi menjelaskan aliran kerja aktor pada proses buat disposisi. Diagram aktivitas buat disposisi dapat dilihat pada gambar berikut ini :



Gambar 5 : Diagram Aktivitas Buat Disposisi

* Diagram Aktivitas Buat Surat Keluar

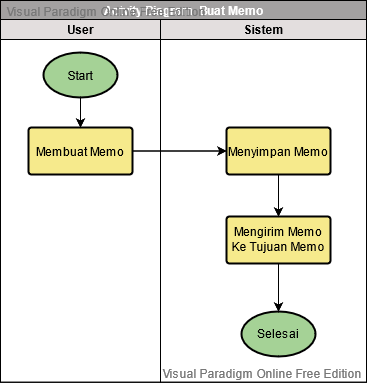
Diagram aktivitas buat surat keluar menjelaskan aliran kerja aktor pada proses buat surat keluar. Diagram aktivitas buat surat keluar dapat dilihat pada gambar berikut ini :



Gambar 6 : Diagram Aktivitas Buat Surat Keluar

* Diagram Aktivitas Buat Memo

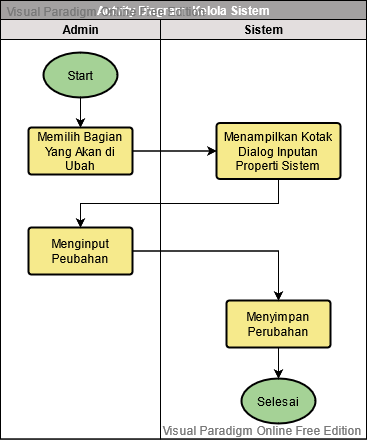
Diagram aktivitas buat memo menjelaskan aliran kerja actor pada proses buat memo. Diagram aktivitas buat memo dapat dilihat pada gambar berikut ini :



Gambar 7 : Diagram Aktivitas Buat Memo

* Diagram Aktivitas Kelola Sistem

Diagram aktivitas kelola sistem menjelaskan aliran kerja aktor pada proses kelola sistem. Diagram aktivitas kelola administrasi properti sistem dapat dilihat pada gambar berikut ini :



Gambar 8 : Diagram Aktivitas Kelola Sistem

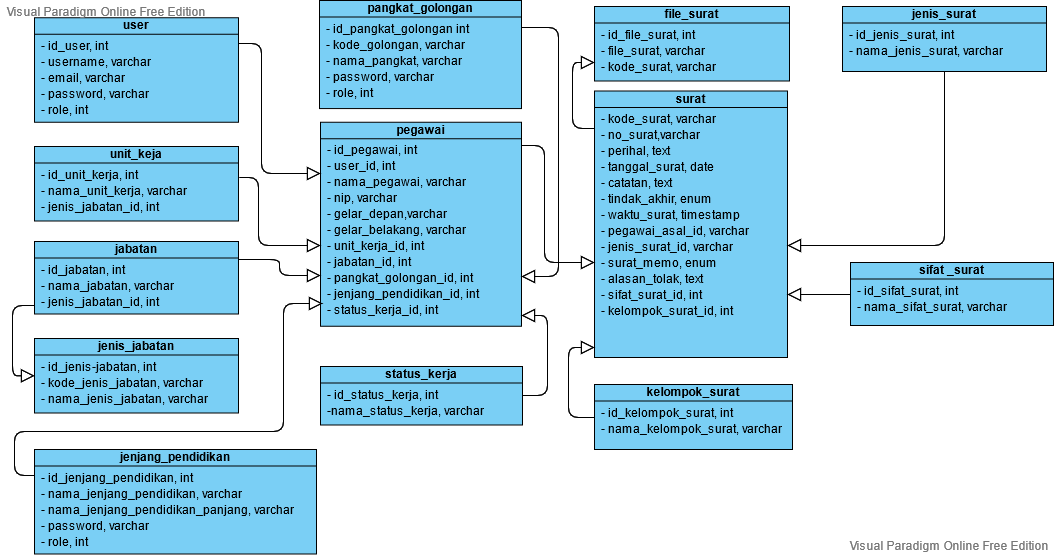
1. Diagram Kelas

Diagram kelas adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

Diagram kelas menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk Memanipulasi keadaan tersebut (metoda/fungsi). Diagram kelas menggambarkan struktur dari deskripsi class, package, dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Diagram kelas memiliki tiga area pokok:

1. Nama (dan stereotype)
2. Atribut
3. Metoda

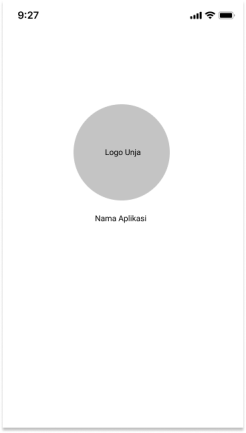
Berikut adalah diagram kelas dari sistem informasi persuratan yang dibuat :



Gambar 9 : Diagram Kelas

**Perancangan Antarmuka Perangkat Lunak**

Perancangan antarmuka perangkat lunak merupakan proses pembuatan antarmuka yang akan digunakan untuk berinteraksi antara pengguna dengan perangkat lunak. Rancangan antarmuka yang dibuat sebaik mungkin sehingga aplikasi yang dikembangkan bersifat user friendly.

****

Gambar 10 : Tampilan Rancangan Antarmuka *Splash*



Gambar 11 : Tampilan Rancangan Antarmuka Onboarding

****

Gambar 12 : Tampilan Rancangan Antarmuka Form Login

**DAFTAR PUSTAKA**

[1] Mohamad Sani. (2010, Oktober) Tutorial Pengantar Pengembangan Aplikasi Android.[Online]. <http://referensi.dosen.narotama.ac.id/files/2011/12/Pengantar-Membuat-Aplikasi-Android.pdf>

[2] Okto Hendrik Sinaga, Shelti E. S. Teresa, and M. Jamil. (2010, Oktober) Sheltiteresaes-just another wordpress.com site. [Online]. <http://sheltiteresaes.files.wordpress.com/2011/10/android.docx>

[3]Supardi, Yuniar. 2014. Semua Bisa Menjadi Programer Android –Case Study Jakarta : PT Elex Media Komputindo.

[4] Roger L. Costello. XFront-tutorials and articles on XML and web technologies. [Online].http://www.xfront.com/REST-Web-Services.html

[5] Google Inc. 2013. Google Cloud Messaging Overview. http://developer.android.com/google/gcm/gcm.html (diakses tanggal 29 Maret 2014).

[6] Pressman, Roger S. 2002. Rekayasa Perangkat Lunak : Pendekatan Praktisi.Terjemahan CN Harnaningrum. Software Engineering : Practitioner’s Approach. 1997. Yogyakarta : ANDI.

[7] Lupa dapusnya