**APLIKASI SMART ABSEN MENGGUNAKAN**

***QR CODE SCANNING* DAN *LOCATION***

***BASED SERVICE (LBS)***

**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat

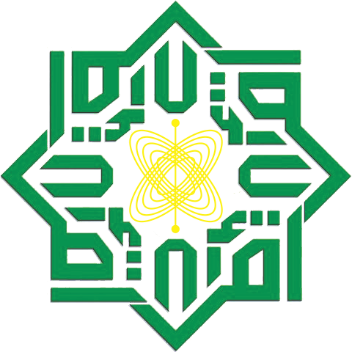
Untuk Memperoleh Gelar Sarjana

Teknik Pada Jurusan Teknik Informatika

Oleh :

**RADINAL DWIKI N**

**11451101637**

****

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU**

**2018**

# PENDAHULUAN

## Latar Belakang

Perkembangan teknologi telah meningkatkan kualitas hidup manusia. Kemajuan teknologi memberikan kontribusi yang signifikan di berbagai bidang seperti kesehatan, nutrisi, agrikultur, industri, transportasi, dan juga pendidikan. Di UIN SUSKA RIAU khususnya Fakultas Sains dan Teknologi, mahasiswa sudah tidak asing lagi dalam hal penggunaan teknologi. Penggunaan teknologi selama proses perkuliahan dapat meningkatkan kualitas proses pembelajaran.

Teknologi tidak hanya dimanfaatkan sebagai fasilitas pendukung selama proses pembelajaran, namun juga sebagai pendukung sistem administrasi mahasiswa di kampus. Tetapi hingga saat ini masih terdapat permasalahan terkait operasional sistem absensi mahasiswa untuk kegiatan perkuliahan di kampus. Absensi mahasiswa yang masih dilakukan secara manual dengan menandatangani daftar hadir setiap memasuki jadwal perkuliahan mata kuliah tertentu masih kurang efektif. Dalam proses absensi mahasiswa masih terjadi kecurangan seperti daftar hadir mahasiswa yang diisi oleh mahasiswa lain. Selain itu, rekapitulasi data absensi di akhir perkuliahan yang dilakukan secara manual oleh dosen juga cukup menyita banyak waktu karena jumlah mahasiswa yang tidak sedikit. Permasalahan yang terjadi pada pengoperasian pendataan absensi manual tersebut menjadi masalah yang wajib diselesaikan sebab kehadiran mahasiswa merupakan salah satu tolok ukur kedisiplinan mahasiswa selama proses perkuliahan. Oleh sebab itu perlu dikembangkan aplikasi dengan sistem yang secara otomatis dapat memverifikasi kehadiran mahasiswa saat jam perkuliahan hingga pengolahan data absensi.

Aplikasi Smart Absen dengan menggunakan *QR Code Scanning* dan *Location Based Service* (LBS) dapat menjadi terobosan terbaru untuk menggantikan proses absensi manual mahasiswa di kampus. Aplikasi ini memanfaatkan perkembangan teknologi pada alat komunikasi. Penggunaan *QR Code Scanning* dan *Location Based Service* (LBS) dalam sistem absensi juga menjadikan proses absensi mahasiswa lebih efektif dan praktis. Selain itu, aplikasi ini juga dapat mengatasi kecurangan mahasiswa dalam mengisi daftar absensi karena verifikasi kehadiran dilakukan berdasarkan lokasi mahasiswa secara *real time*.

Perangkat telepon seluler yang saat ini biasa disebut dengan istilah *smartphone* adalah alat komunikasi yang dapat dibawa kemana pun. Kini telepon seluler tak hanya terbatas untuk melakukan panggilan suara dan mengirim pesan berbasis teks saja, namun juga sudah dilengkapi dengan berbagai fitur yang membantu pekerjaan manusia. Berbagai fitur canggih yang hampir ada di setiap jenis telepon seluler adalah fitur *chatting*, *browsing*, *video call, barcode scanner, location service*. *Smartphone* yang paling banyak digunakan di dunia *smartphone* dengan *operating system* Android, Android memiliki *market share* 85% dari penggunaan *smartphone* di dunia (IDC, 2017).

*Location Based Service* (LBS) didefinisikan sebagai layanan yang menggabungkan antara lokasi perangkat mobile dengan informasi lain sehingga menyediakan nilai tambah untuk pengguna (Schiller & Voisard, 2004). Dengan layanan berbasis lokasi ini sangat memungkinkan untuk melakukan pengolahan data berupa lokasi dari setiap pengguna Android. Sehingga teknologi LBS ini dapat dijadikan solusi dalam membantu absen konvensional. Selain LBS, *QR Code* juga merupakan salah satu inovasi dalam bidang teknologi yang cukup populer beberapa tahun terakhir. *QR Code* atau *Quick Response Code* adalah kode matrik 2D yang menyimpan sejumlah data yang besar sebagai perbandingan ke *barcode* 1D dan dapat dibacadalam waktu singkat menggunakan *smartphone* (Tiwari, 2017).

Penerapan teknologi LBS dan *QR Code* memungkinkan dihasilkannya sebuah aplikasi yang membantu mahasiswa untuk melakukan absensi hanya dengan melalui *smartphone* Android. Aplikasi ini dirancang dengan model *client-server*, disamping mudah dalam menangani *database* juga relatif lebih sederhana untuk dikembangangkan dan diimplementasikan. Aplikasi terdiri dari *font-end* dan *back-end*, versi *mobile* *application* atau *front-end* akan dioperasikan oleh mahasiswa, dosen dan versi *web application* atau *back-end* akan dioperasikan oleh admin.

Aplikasi yang terpasang di perangkat seluler mahasiswa secara otomatis akan membaca lokasi dari mahasiswa tersebut. Data lokasi terbaru mahasiswa kemudian dibandingkan dengan data lokasi yang diperbolehkan untuk absen. Jika lokasinya sesuai maka menu scan *QR Code* akan terbuka, setelah dosen menampilkan *QR Code* di depan kelas, mahasiswa melakukan *scan QR Code,* aplikasi melakukan proses validasi dan mengirimkan data absensi ke server dengan status *pending*. Dosen melakukan *checkout* terhadap daftar mahasiswa yang berhasil tervalidasi, dosen menekan tombol kirim dan data absensi terkirim dengan status *ok.* Sehingga Aplikasi ini hanya memperbolehkan pengguna untuk melakukan absensi di kampus saja yang dalam penelitian ini adalah Fakultas Sains dan Teknologi UIN SUSKA RIAU.

Aplikasi ini juga memiliki fitur untuk membantu mahasiswa dalam melihat kehadiran dosen dan pegawai di kampus, dengan menggunakan *Reverse Geocoding* dari Google Maps API dilakukan penerjemahan koordinat lokasi dosen dan pegawai tersebut agar menjadi nilai lokasi yang mudah dibaca seperti nama kota. Untuk mendukung penelitian ini, maka perlu adanya pemaparan dari penelitian terkait dengan aplikasi yang dibangun seperti:

Penelitian oleh Sanga Son dan Yongtae Shin tahun 2016 dengan judul “*Design of Smart Shopping Application Using Barcode Scanning and Location Based Coupon Service*, 2016 IEEE”. Pada penelitian ini dibangun sebuah aplikasi yang dapat menambahkan produk ke dalam keranjang belanja online dengan cara barcode scanning menggunakan kamera smartphone kemudian melakukan pemesanan dengan toko online yang terhubung (Son & Shin, 2016)

Penelitian oleh Swastikasari dkk tahun 2017 dengan judul “*Design of E-KOST* : *an Android-Based Mobile Application Using Location Based Service*, 2017 IEEE”. Pada penelitian ini dibangun sebuah aplikasi dengan teknologi *Location Based Service* untuk membantu mahasiswa Universitas Kristen Satya Wacana dalam menemukan kost terdekat dan membantu pemilik bisnis kost dalam mencari penyewa kost (Swastikasari dkk, 2017)

Penelitian oleh Soewito dkk tahun 2017 dengan judul “*Smart Mobile Attendance System Using Voice Recognition and Fingerprint on Smartphone*, 2017 IEEE”. Pada penelitian ini dibangun sebuah aplikasi yang menggunakan *smartphone* untuk memverifikasi absensi karyawan. Aplikasi menyediakan dua pilihan untuk melakukan verifikasi, yaitu *fingerprint* dan *voice recognition*. Dihasilkan *false positive* pada *fingerprint verification* sebesar 95% dan *false negative* dari *voice recognition* sebesar 5,88% (Soewito dkk, 2017).

Penelitian oleh Raghav Apoorv dan Puja Mathur tahun 2016 dengan judul “*Smart Attendance Management using Bluetooth Low Energy and Android*, 2016 IEEE”. Pada penelitian ini mengimplementasikan teknologi *Bluetooth low energy* pada sistem manajemen absensi. Menggunakan teknologi *Bluetooth Low Energy of beacons* yang terhubung dengan aplikasi Android. Aplikasi ini mengumpulkan data dari sensor dan disimpan berdasarkan tanggal. Aplikasi ini memudahkan guru dalam mencatat dan menganalisa absensi murid-murid (Apoorv & Mathur, 2016).

Penelitian oleh Wasum dkk tahun 2015 dengan judul “*Application Search School Location Method Using Location Based Service (LBS) based on J2ME*, 2015 IEEE”. Pada penelitian ini dibangun sebuah aplikasi untuk memudahkan masyarakat mengetahui lokasi sekolah terdekat dengan lokasi pengguna. (Wasum dkk, 2015)

Dengan latar belakang tersebut, maka akan dilakukan penelitian dengan judul “Rancang Bangun Aplikasi Smart Absen Menggunakan *QR Code Scanning* dan *Location Based Service* (LBS)”.

## Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat diambil rumusan masalah yaitu Bagaimana Merancang Bangun Aplikasi Smart Absen Menggunakan *QR Code Scanning* dan *Location Based Service* (LBS).

## Batasan Masalah

Agar penelitian tetap fokus pada masalah yang akan dibahas, maka dibutuhkan batasan masalah. Batasan masalah penelitian ini adalah sebagai berikut:

1. Sistem ini tidak mencakup absensi harian dosen, namun hanya absensi matakuliah saja, dosen memberikan *qr-code* dengan cara ditampilkan melalui proyektor di depan kelas.
2. Sistem ini menggunakan Bot Telegram untuk memberikan *early warning* kepada mahasiswa yang memiliki absen melebihi 2.
3. Observasi lokasi menggunakan Google Maps
4. Data lokasi yang digunakan adalah data di kampus Uin Suska Riau.

## Tujuan Penelitian

Berdasarkan rumusan masalah yang disebutkan di atas, maka tujuan dari penelitian ini adalah menghasilkan Aplikasi Smart Absen Menggunakan *QR Code Scanning* dan *Location Based Service* (LBS).

## Sistematika Penulisan

Laporan Tugas Akhir ini terdiri dari enam bab dengan sistematika penulisan sebagai berikut :

**BAB I PENDAHULUAN**

Berisikan tentang latar belakang permasalahan, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Bagian ini membahas teori-teori yang mendukung dalam proses pengerjaan Tugas Akhir ini.

**BAB III METODOLOGI PENELITIAN**

Bab ini berisi tentang rangkaian tahapan dalam penelitian, tahapan pengumpulan data, analisis kebutuhan sistem, perancangan perangkat lunak, implementasi, pengujian dan waktu penelitian.

**BAB IV ANALISA DAN PERANCANGAN**

Bab ini membahas analisa langkah kerja dan perancangan aplikasi yang akan dikembangkan sesuai dengan metodologi yang digunakan.

**BAB V IMPLEMENTASI DAN PENGUJIAN**

Bab ini membahas implementasi dan pengujian hasil penelitian sesuai yang tertera pada rumusan masalah dan pembahasan.

**BAB VI PENUTUP**

Pada bab ini membahas mengenai kesimpulan yang dapat diambil dari penelitian dan saran yang di peroleh untuk pengembangan lebih lanjut.

# LANDASAN TEORI

## Sistem Absensi

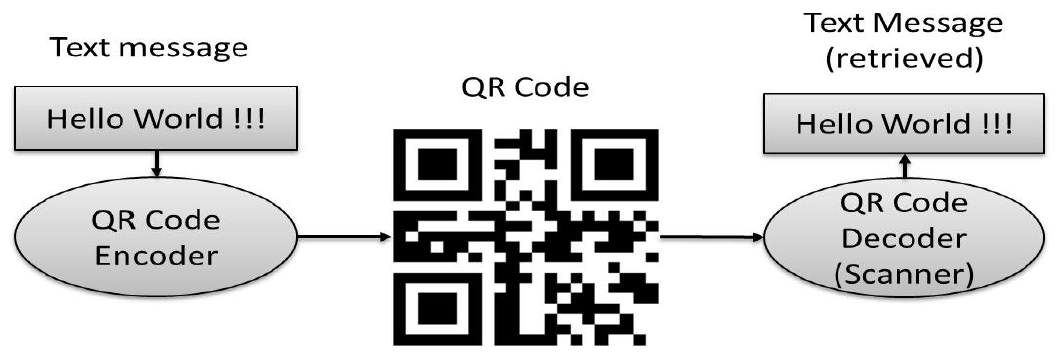
Sistem absensi atau sistem kehadiran sudah dikenal sejak lama, mulai dari sistem manual menggunakan buku catatan absensi hingga sistem yang otomatis dan rumit seperti sistem absen *biometric* atau lebih dikenal absen *fingerprint*. Namun kedua sistem tersebut sama-sama memiliki kelemahan, terutama dalam mengontrol pegawai yang sedang bekerja diluar kantor (Soewito dkk, 2017).

## *QR Code*

Menurut (Sutheebanjard & Premchaiswadi, 2010) dikutip oleh (Tiwari, 2017) *QR Code* adalah *barcode* tipe matrik atau kode dua dimensi yang dapat menyimpan informasi dan dapat dibaca oleh *smartphone*. *QR* merupakan kepanjangan dari *Quick Response* yang artinya kode tersebut harus di *decode* dalam waktu singkat. Kode terdiri dari modul hitam yang disusun dalam bentuk pola persegi dengan latar belakang berwarna putih. Informasi yang dikodekan adalah teks, URL. *QR Code* dirancang untuk dapat dipindai secara cepat, popularitas teknologi *QR Code* bertambah dengan cepat di seluruh dunia. Saat ini, telepon genggam dengan kamera sudah dapat digunakan untuk mengenali *QR Code*.

Bermula pada tahun 1994, Denso Wave yang merupakan cabang dari perusahaan Toyota berhasil menciptakan teknologi *QR Code*. Awalnya teknologi ini digunakan untuk memantau stok pada pabrik kendaraan. Ide yang muncul dari pengembangan *QR Code* ini adalah keterbatasan kapasitas teknologi *barcode* yang hanya bisa memuat 20 karakter alfanumerik.

Teknologi yang dulunya hanya sebatas memantau stok di pabrik kendaraan kini sudah digunakan di banyak bidang serperti komersil, label barang di toko, dan aplikasi-aplikasi yang ditujukan untuk pengguna *smartphone*. Pengguna dapat membuka URL, menerima teks setelah memindai *QR Code*. dengan menggunak situs atau aplikasi *QR Code generator* , pengguna bisa menghasilkan dan mencetak *QR Code* milik mereka sendiri untuk dipindai dan digunakan orang lain (Tiwari, 2017).



Gambar 2. Cara kerja *QR Code* (Tiwari, 2017)

Gambar diatas menjelaskan bagaiman prinsip kerja *QR Code*. Teks biasa atau URL di *encode* oleh *QR Code encoder* dan menghasilkan *QR Code* apabila data ingin diakses atau didapatkan kembali, lakukan proses *decode* dengan *QR Code decoder* atau *scanner*.

## *Location Based Service* (LBS)

*Location Based Service* dapat didefinisikan sebagai layanan yang mengintegrasikan sebuah lokasi atau posisi perangkat *mobile* dengan informasi lainnya sehingga memberikan nilai tambah bagi pengguna (Schiller & Voisard, 2004).

*Location Based Service* merupakan layanan berbasis lokasi yang menggambarkan teknologi untuk menemukan posisi perangkat yang sedang kita gunakan. Ada dua unsur utama LBS yakni (Safaat, 2014):

1. *Location Manager* (API *Maps*)

Menyediakan *tools/source* untuk LBS, *Application Programming Interface* (API) Maps menyediakan fasilitas untuk menampilkan, memanipulasi *maps*/peta beserta *feature-feature* lainnya seperti tampilan satelit, *street* (jalan), maupun gabungannya. Paket ini berada pada com.google.Android.maps.

1. *Location Providers* (API *Location*)

Menyediakan teknologi pencarian lokasi yang digunakan oleh *device*/perangkat. API *Location* berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi *real-time.* API *location* berapa pada paket Android yaitu dalam paket Android.location. Dengan *Location Manager*, kita dapat menentukan lokasi kita saat ini, *Track* gerakan/perpindahan, serta kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan.

## Android

Pada tahun 2000, sebelum produk ponsel cerdas beredar di pasaran, Google mengakuisi perusahaan bernama Android. Perusahaan tersebut merupakan pendatang baru berfokus untuk pengembangan sistem operasi embedded dan software untuk perangkat mobile (Istiyanto, 2013).

### Sejarah Android

Android adalah sistem operasi *mobile* berbasis Linux yang mencakup sistem operasi, middleware dan aplikasi. Android memberikan platform terbuka yang dapat dikembangkan oleh para *developer* untuk membuat aplikasi sendiri, aplikasi ini dapat digunakan diberbagai perangkat *mobile*. Android dapat berkembang pesat karena memiliki platform yang sangat lengkap baik itu sistem operasinya, Aplikasi dan Tool Pengembangan, Market aplikasi Android serta dukungan yang tinggi dari komunitas *Open Source* di dunia, sehingga Android dapat berkembang pesat di dunia baik dari segi teknologi maupun dari segi jumlah *device* yang ada di dunia.

Pada saat peluncuran pertama Android, 5 November 2007, Android dan *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka atau *open source* pada perangkat seluler. Di sisi lain, Google merilis kode-kode Android di bawah lisensi Apache, yakni lisensi perangkat lunak dan open platform perangkat seluler.

Sekitar September 2007, Google mempubilkasikan salah satu jenis smartphone yang bersistem operasi Android, yakni Nexus One. Sebuah smartphone yang diproduksi oleh HTC Corporation dan sudah tersedia di pasaran pada 5 Januari 2010. Pada tanggal 9 Desember 2008, anggota program kerja Android sudah mulai bertambah, ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc (Safaat, 2014).

### Arsitektur Android

Secara garis besar Arsitektur Android adalah sebagai berikut (Safaat, 2014):

1. Applications dan Widgets

*Applications* dan *Widgets* adalah layer yang berhubungan dengan aplikasi saja, yang mana biasanya kita lakukan proses download aplikasi, proses instalasi dan proses menjalankan aplikasi tersebut. Pada layer ini terdapat aplikasi inti yang ditulis dalam bahasa pemrograman java yakni klien email, program SMS, peta, kalender, kontak, browser.

1. Applications Framework

*Appilcation Frameworks* adalah layer di mana para pengembang melakukan pengembangan aplikasi yang akan dijalankan di sistem operasi Android, karna aplikasi yang dirancang akan dibuat pada layer ini seperti *content-providers* yang berupa sms dan panggilan telpon.

Komponen-komponen penyusun layer ini adalah:

1. *Views*
2. *Content Provider*
3. *Resource Manager*
4. *Notification Manager*
5. *Activity Manager*
6. Libraries

*Libraries* adalah layer di mana fitur-fitur Android berada, seperti pemutaran audio dan video, manajemen tampilan. Biasanya para pengembang mengakses *libraries* untuk menjalankan aplikasinya. Layer ini berjalan di atas kernel, meliputi berbagai libari C/C++ inti seperti Libc dan SSL.

1. Android Run Time

*Android Run Time* adalah layer di mana terdapat implementasi Linux di dalam prosesnya. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membentuk kerangka dasar sistem operasi Android. Dalam layer ini terdapat dua bagian yaitu *Core Libraries* yaitu *libraries* untuk menterjemahkan bahasa java/c, dan *Dalvik Virtual Machine* yang berfungsi sebagai virtual masin berbasis register dimana pengembangan yang mampu membuat linux kernel untuk melakukan threading dan manajemen tingkat rendah.

1. Linux Kernel

*Linux Kernel* adalah layer dimana inti dari sistem operasi Android berada. Isi dari layer ini adalahfile-file sistem yang mengatur, *system processing, resource, drivers,* dan sistem-sistem operasi Android lainnya. Android menggunalan linux kernel release 2.6.

### Versi Android

Versi-versi Android yang pernah dirilis adalah sebagai berikut (Safaat, 2014):

1. Android versi 1.1

Android versi pertama dirilis pada 9 Maret 2009. Android ini dilengkapi dengan pembaruan estetis pada aplikasi, jam, alarm, *voice searching,* pengiriman pesan melalui Gmail, dan pemberitahuan email.

1. Android versi 1.5

Google kembali merilis pada pertengahan Mei 2009, telpon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Fitur pada perilisan kedua ini sudah banyak bertambah, seperti merekam dan menonton video dengan modus kamera, mengupload video ke Youtube dan gambar ke Pucasa langsung dari telepon, Bluetooth A2DP, otomatis terhubung ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat menyesuaikan sistem.

1. Android versi 1.6 (*Donut*)

Android versi 1.6 dirilis pada September 2009 dengan menampilkan proses pencarian yang lebih baik dibandingkan sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur pada galeri untuk memilih foto yang akan dihapus, kamera, camcorder dan galeri yang diintegrasikan, CDMA/EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine, dial kontak, *text to change speech*(tidak tersedia pada semua ponsel). Pengadaan resolusi VWGA.

1. Android versi 2.0/2.1 (*Eclair*)

Android ini dirilis pada 3 Desember 2009, perubahan yang terjadi adalah pengoptimalan hardware, pengingkatan *Google Maps* 3/1/2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 M, digital Zoom, dan Bluetooth 2.1.

1. Android versi 2.2 (*Froyo: Frozen Yoghurt*)

Android ini dirilis pada bulan Mei 2010, fitur yang tersedia pada Android ini sudah komplek yakni:

1. Kerangka aplikasi memungkinkan penggunaan dan penghapusan komponen yang tersedia.
2. *Dalvik Virtual Machine dioptimalkan untuk perangkar mobile.*
3. Grafik: grafik di 2D dan grafis di 3D berdasarkan libraries OpenGL.
4. *SQLite:* untuk penyimpanan data
5. Mendukung media: audio, video, berbagai format gambar.
6. GSM, Bluetooth, EDGE, 4G dan WiFi (hardware independent)
7. Kamera, *Global Positioning System* (GPS), kompas, dan *acceleromete* (tergantung hardware)
8. Android versi 2.3 (*Gingerbread*)

Android versi ini diluncurkan pada Desember 2010, beberapa revisi dari versi sebelumnya adalah kemampuan sebagai berikut:

1. *SIP-based VoIP*
2. *Near Field Communications (NFC)*
3. Gyroscope dan sensor
4. *Multiple cameras support*
5. *Mixable audio effects*
6. *Download manager*
7. Android versi 3.0 (*Honeycomb*)

Android versi 3.0/3.1 dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada Android versi ini terlihat berbeda karna didesain untuk tablet. *Honeycomb* juga mendukung *multi processor* dan juga akselerasi perangkat keras (*hardware*) untuk grafis. Motorola Xoom merupakan tablet pertama yang menjalankan *Honeycomb*.

1. Android versi 4.0 (*Ice Cream Sandwich*)

Android versi 4.0 telah diperkenalkan pada Mei 2011, dan resmi dirilis pada tanggal 19 Oktober 2011. Dengan penyempurnaan fitur Android *Honeycomb* sebelumnya seperti pengunci dengan pengenal muka (*facial recognition unlock*), *control* dan pengawasan penggunaan data jaringan (*network data usage monitoring*), sinkronisasi kontak dengan jaringan sosial (*social networking contacts*), kualitas foto lebih tinggi, *offline email searching*, dan berbagai informasi menggunakan NFC (*Near Field Communication*). Samsung Galaxy Nexus merupakan *smartphone* pertama yang menggunakan Android *Ice Cream Sandwich*.

1. Android versi 4.1 (*Jelly Bean*)

Android versi 4.1 dirilis pada Juli 2012. Terjadi beberapa perubahan pada Android versi ini, seperti peningkatan *input keyboard*, desain baru dan penyempurnaan pada fitur pencarian yang sudah disempurnakan UI dan *voice search* yang lebih cepat. Diwaktu yang sama Google merilis *Google Now*, *Google Now* memberikan informasi yang tepat pada waktu yang sekarang. Salah satu fitur andalannya adalah, informasi cuaca, lalu lintas dan hasil pertandingan. Google Nexus 7 merupakan *smartphone* pertama yang menggunakan Android *Jelly Bean*.

1. Android versi 4.4 (*KitKat*)

Android versi 4.4. dikenalkan oleh Google pada 3 September 2013, dan dirilis pada 31 Oktober 2013. Versi ini menghadirkan performa system tertinggi dengan mengoptimalkan memori dan menyempurnakan layar sentuh agar merespon lebih cepat dan akurat. Anda dapat mendengarkan musik saat menjelajah web atau bermain game balapan terbaru, lakukan semua dengan lancar”.

1. Android versi 5.0 (*Lollipop*)

Android versi 5.0 diperkenalkan pertama kali dengan *codename* Android L pada 25 Juni 2014 selama presentasi keynote pada konferensi penegembang Google I/O. Pada versi ini, sudah banyak penyempurnaan fitur seperti *Material design*, notifikasi, baterai, keamanan, berbagi perangkat, konektivitas, dan banyak lagi.

1. Android versi 6.0 (*Marshmallow*)

Android versi 6.0 memperkenalkan model *permission* yang didesain ulang, sekarang hanya ada delapan jenis *permission*, dan aplikasi tidak secara otomatis meminta *permission* saat aplikasi diinstall namun *permission* dimintasaat aplikasi dijalankan. Android *Marshmallow* juga memiliki skema manajemen daya baru yang bernama Doze, sehingga versi ini akan lebih sedikit menggunakan *resource*.

1. Android versi 7.0 (*Nougat*)

Android versi 7.0 pertama kali dirilis sebagai pratinjau pengembang pada tanggal 9 Maret 2016. Rilis terakhir dari Android versi ini adalah tanggal 22 Agustus 2016. Pada versi ini sudah terdapat setelan Bahasa multilokal, dapat menjalankan dua aplikasi sekaligus, enkripsi berbasis file dan lain-lain.

1. Android versi 8.0 (*Oreo*)

Android versi 8.0 pertama kali dirilis sebagai *preview* pengembang pada tanggal 21 Maret 2017. Pratinjau pengembang terakhir dirilis pada tanggal 24 Juli 2017. Slogan dari versi ini adalah “*Smarter, faster, more powerful and sweeter than ever”.* Android *Oreo* 2x lebih cepat, dilengkapi dengan fitur *Autofill* dan *permission* pengguna memungkinkan *Login* lebih cepat pada aplikasi dan lain-lain.

## *Mock Location Data*

Sebagai pengembang aplikasi, tentu saja diperlukan pengujian untuk melihat apakah model yang dibuat benar-benar bisa mendapatkan lokasi pengguna. Hal ini sangat mudah dilakukan bila menggunakan perangkat Android asli. Jika tidak memiliki perangkat Android dan tetap ingin mencoba hal tersebut, fitur *location-based* masih bisa dicoba dengan menggunakan *emulator* Android.

*Mock location* atau lokasi tiruan dapat digunakan dengan memilih opsi *mock location* yang sudah disediakan di menu *developer options,* atau dengan menggunakan *geo command* pada konsol *emulator*. *Mock location* bekerja dengan cara memasukkan data lokasi ke data lokasi GPS, sehingga dibutuhkan GPS\_PROVIDER untuk menjalankan fitur dari *mock location* tersebut (Developer, 2018).

## Telegram

Telegram adalah aplikasi pesan yang berfokus pada kecepatan dan keamanan, aplikasi ini sangat cepat, sederhana, dan gratis. Telegram dapat digunakan di semua perangkat dalam waktu yang sama. Aplikasi ini menerima pesan dalam bentuk teks, foto, video dan file.

## *Unified Modelling Language* (UML)

Pemodelan adalah gambaran dari realita yang sederhana dan dituangkan kedalam bentuk pemetaan dengan aturan tertentu. UML (*Unified Modeling Language*) merupakan suatu alat bantu untuk pemodelan, definisi dari UML itu sendiri adalah bahasa visual untuk pemodelan dan komunikasi dari sebuah sistem dalam bentuk diagram dan teks-teks pendukung (Rosa & Shalahuddin, 2013).

UML terdiri dari 13 macam diagram yang dikelompokkan ke dalam 3 kategori sebagai berikut :

1. *Structure Diagram* (*Static View*)
   1. *Class diagram*
   2. *Object diagram*
   3. *Component diagram*
   4. *Composite structure diagram*
   5. *Package diagram*
   6. *Deployment diagram*
2. *Behavior Diagram* (*Functional View*)
   1. *Use case diagram*
   2. *Activity Diagram*
   3. *State machine diagram*
3. *Interaction Diagram* (*Dynamic View*)
   1. *Sequence diagram*
   2. *Communication diagram*
   3. *Timing diagram*
   4. *Interaction overview diagram*

### *Use Case* Diagram

*Use Case Diagram* menjelaskan apa yang sistem dapat lakukan melalui pemodelan fitur-fitur sistem yang terlihat dan berguna bagi aktor. *Use Case* tidak menjelaskan bagaiman alur kerja sistem atau bagaimana pengimplementasian sistem. Penggunaan *Use Case* adalah untuk menjelaskan sistem, lingkunganya, dan hubungan antara sistem dan lingkunganya. Sehingga prilaku sistem dapat dijelaskan melalui *Use Case* (Fikry dkk, 2015)*.*

Tabel 2.1 Simbol-simbol Use Case Diagram

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan fungsi** |
| 1 | Aktor  Hasil gambar untuk simbol aktor uml png | Aktor merepresentasikan semua yang berinteraksi dengan sistem. Aktor bisa berupa orang, mesin, atau sistem lain. |
| 2 | *Use case*  Hasil gambar untuk simbol use case uml png | *Use Case* adalah urutan transaksi yang dilakukan oleh sistem, menghasilkan hasil yang terukur untuk aktor. |
| 3 | Asosiasi  Hasil gambar untuk line png | Mengilustrasikan interaksi antara aktor dan *Use Case* dengan cara mengirimkan stimulan antara satu dengan lainnya. |

### Sequence Diagram

*Sequence Diagram* menjelaskan aliran eksekusi dari setiap aliran yang ada pada *Use Case*. Sehingga memungkinkan deskripsi tekstual dari prilaku yang ada di dalam *Use Case* diterjemahkan menjadi operasi pada *Class Diagram*.

Skenario adalah instans dari *Use Case*, berisi uraian kejadian yang terjadi selama proses eksekusi sistem. Skenario yang dihasilkan dari setiap *Use Case* bervariasi sesuai kebutuhan. Skenarion ini dimodelkan dengan *Sequence Diagram*. Objek-objek dari *Sequence Diagram* dapat diidentifikasi dengan melihat kata benda yang terdapat di dalam *Use Case* dan skenario (Fikry dkk, 2015).

Tabel 2.2 Simbol-simbol Sequence Diagram

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan fungsi** |
| 1 | Aktor  Hasil gambar untuk simbol aktor uml png | Aktor merepresentasikan semua yang berinteraksi dengan sistem. Aktor bisa berupa orang, mesin, atau sistem lain. |
| 2 | *Lifeline* | Objek *entity*, antarmuka yang saling berinteraksi. |
| 3 | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |
| 4 | *Boundary* | Menggambarkan sebuah penggambaran dari form. |
| 5 | *Control* | Menggambarkan penghubung antara boundary dengan tabel. |
| 6 | *Entity* | Menggambarkan hubungan kegiatan yang akan dilakukan. |

### Class Diagram

*Class Diagram* menjelaskan struktur sistem dari segi pendefinisan kelas-kelas yang akan dibuat untuk pembangunan sistem. Dalam kelas terdapat atribut dan metode. Denifisi atribut adalah variabel-variabel yang dimiliki oleh sutau kelas, sementara metode adalah fungsi-fungsi atau *method* yang dimiliki kelas tersebut (Fikry dkk, 2015).

Tabel 2.3 Simbol-simbol Class Diagram

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan fungsi** |
| 1 | *Class* | Kumpulan dari objek-objek yang berbagi atribut serta operasi yang sama. |
| 2 | Generalisasi  Hasil gambar untuk line png | Hubungan antara objek anak *(descendent)b*erbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*). |
| 3 | *Nary Association* | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |

### Activity Diagram

*Activity Diagram* menjelaskan *workflow* (aliran kerja) dari sebuah sistem atau proses bisnis. Perlu diperhatikan bahwa *Activity Diagram*  menggambarkan alur kerja sistem bukan apa yang dilakukan aktor.

Tabel 2.4 Simbol-simbol Activity Diagram

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan** |
| 1 | *Start State* | *Start state* adalah kondisi awal dari sebuah *object*terjadinya perubahan keadaan. Start state digambarkan dengan sebuah lingkaran solid. |
| 2 | *End State* | *End state* adalah gambaran ketika objek berhenti memberi respon terhadap sebuah event. *End state* digambarkan dengan lingkaran solid di dalam sebuah lingkaran kosong. |
| 3 | *State/Activities* | *State* atau *activities* merepresentasikan kondisi dari sebuah entitas, dan digambarkan dengan segiempat yang pinggirnya. |
| 4 | *Fork* (Percabangan) | *Fork* atau percabangan adalah pemisalah beberapa aliran konkuren dari sebuah aliran tunggal. |
| 5 | *Join* (Penggabungan) | *Join* atau penggabungan merupakan pengabungan beberapa aliran konkuren dalam seuatu aliran tunggal. |
| 6 | *Decision* | *Decision*adalah suatu logika aliran konkuren yang mempunyai dua cabang aliran konkuren. |

### Deployment Diagram

## Penelitian Terkait

Tabel 2.4 berikut menjelaskan mengenai peneltian-penelitan sebelumnya tentang Absensi dan *Location Based Service* (LBS) :

Table 2.4 Penelitian Terkait

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Peneliti | Judul Penelitian | Metode | Hasil |
| 1. | Sanga Son dan Yongtae Shin , 2016 | Design of Smart Shopping Application Using Barcode Scanning and Location Based Coupon Service | Location Based Service(LBS) | Pada penelitian ini dirancang sebuah aplikasi yang dapat menambahkan produk ke dalam keranjang belanja online dengan cara barcode scanning menggunakan kamera smartphone kemudian melakukan pemesanan dengan toko online yang terhubung |
| 2. | Swastikasari, M. M., Sediyono, E., & Ardjo, A. S. (2017) | Design of E-KOST : an Android-Based Mobile Application Using Location Based Service. | Location Based Service(LBS) | Pada penelitian ini dirancang sebuah aplikasi yang dapat menghubungkan antara pemilik kost dan calon penyewa kost. Dengan menggunakan teknologi *Location Based Service*, calon penyewa kost dapat menemukan kost terdekat sesuai dengan harga yang diinginkan.­­­­­­ |
| 3. | Benfano Soewito, Ford Lumban Gaol, Echo Simanjuntak, Fergyanto E. Gunawan, 2017 | Smart Mobile Attendance System Using Voice Recognition and Fingerprint on Smartphone | Fingerprint Matching Based on Minutiae and Texture Features, Voice Recognitioin | Pada penelitian ini dibangun sebuah aplikasi yang menggunakan smartphone untuk memverifikasi absensi karyawan. Menggunakan fingerprint dan voice recognition. Dihasilkan 95% false positive pada fingerprint verification dan 5,88% false negative dari voice recognition. |
| 4. | Raghav Apoorv dan Puja Mathur | Smart attendance management using Bluetooth low energy and Android | Bluetooth low energy | Pada penelitian ini mengimplementasikan teknologi *Bluetooth low energy* pada sistem manajemen absensi. Menggunakan teknologi *Bluetooth Low Energy of beacons* yang terhubung dengan aplikasi Android. Aplikasi ini mengumpulkan data dari sensor dan disimpan berdasarkan tanggal. Aplikasi ini memudahkan guru dalam mencatat dan menganalisa absensi murid-murid. |
| 5. | Wasum, Andreas Setiyono, Raimundus Sedo, Sholeh Hadi Pramono, 2015 | Application search school location method using location based services (LBS) based on J2ME | Global Positioning System(GPS) Sensor Barometrics dan Sensor Magnetics | Pada penelitian ini dirancang sebuah aplikasi untuk memudahkan masyarakat mengetahui lokasi sekolah terdekat dengan lokasi pengguna. Aplikasi berbasis J2ME, studi kasus cirebon. Aplikasi ini menggunakan metode *location based service* dalam menyediakan informasi lokasi dan posisi geografis sekolaj. Aplikasi ini mengugnakan teknologi terbaru sehingga memudahkan pengguna untuk mencari informasi dan promosi edukasi. Ada tiga pilihan sekolah yani SD SMP dan SM, pengujian aplikasi menggunakan Nokia X2 dan Nokia C6. |

# METODOLOGI PENELITIAN

Metodologi penelitian menjelaskan rencana kerja yang akan dilakukan dalam penelitian agar hasil yang didapatkan sesuai dengan yang diharapkan. Gambar berikut ini menjelaskan mengenai tahapan-tahapan yang akan dilakukan dalam penelitian ini :

MULAI

STUDI LITERATUR

OBSERVASI

WAWANCARA

BUKU,

JURNAL,

ARTIKEL

Mendapatkan informasi mengenai sistem absensi Fakultas Sains dan Teknologi berupa data-data seperti alur proses absensi, pengalaman pengguna selama menggunakan sistem absensi fingerprint.

ANALISA KEBUTUHAN

DESAIN

PENGKODEAN PROGRAM

PENGUJIAN

KESIMPULAN DAN SARAN

SELESAI

Gambar 3. Tahapan Penelitian

## Pengumpulan Data

Langkah awal yang harus dilakukan untuk menentukan data apa saja yang dibutuhkan pada penelitian adalah dengan mengumpulkan data. Pada penelitian ini pengumpulan data dilakukan dengan teknik studi literatur, wawancara dan observasi. Studi literatur dilakukan dengan membaca teori dan penelitian terkait. Wawancara dilakukan kepada dosen, pegawai dan mahasiswa Fakultas Sains dan Teknologi sebagai pengguna. Observasi dilakukan dengan mengamati langsung kegiatan absensi *fingerprint* pada dosen dan absensi manual pada mahasiswa saat jam perkuliahan. Sehingga akan diperoleh bagian sistem mana yang dianggap baik dan dianggap kurang baik.

## Pembuatan Sistem

Dalam pembuatan perangkat lunak ini akan menggunakan model SDLC *Waterfall* (air terjun) karna *Waterfall* cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah dan *Waterfall* juga mempermudah dalam merancang dan membangun aplikasi dengan jangka waktu yang singkat dan jelas. Tahapan untuk melakukan pembuatan perangkat lunak dilakukan secara sekuensial atau terurut dan saling berkesinambungan. Berikut adalah tahapan dalam pembuatan perangkat lunak :

### Analisa Kebutuhan

Tahap pengumpulan kebutuhan pengguna akan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar pengguna paham dengan perangkat lunak yang dibutuhkan. Data yang didapat pada penelitian ini adalah hasil wawancara dan observasi, data akan diolah untuk mendapatkan solusi dari masalah yang ada. Spesifikasi kebutuhan perangkat lunak pada tahap analisa akan didokumentasikan. Analisa pembangunan sistem menggunakan metode UML diantaranya : *Usecase Diagram*, *Usecase Scenario*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

### Desain

Pada tahap desain akan dilakukan proses pembuatan desain program perangkat lunak berdasarkan kebutuhan perangkat lunak dari tahap analisis. Sehingga tahap desain sistem akan menghasilkan representasi desain dari tahap analisis kebutuhan. Desain perangkat lunak yang dihasilkan pada tahap ini akan didokumentasikan. Metode pendekatan sistem yang adalah pendekatan dengan *Object Oriented* dengan menggunakan *Object Oriented Analysis* (OOA) dan *Object Oriented Design* (OOD) yang akan divisualisasikan dengan *Unified Modeling Language* (UML). Model desain UML yang akan digunakan adalah *Usecase Diagram*, *Usecase Scenario*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

### Pengkodean program

Pengkodean program dilakukan dengan mentranslasikan desain ke dalam program perangkat lunak. Sehingga akan diperoleh hasil program komputer sesuai dengan rancangan pada tahap desain yang sudah dilakukan. Tahap pengkodean program (*coding*) akan dilakukan pada aplikasi *mobile* dan *web*, aplikasi *mobile* dalam kasus ini adalah Android yang akan diprogram menggunakan bahasa Java dan aplikasi web yang akan diprogram menggunakan bahasa PHP, *database* MySQL sebagai sistem manajemen basis data.

### Pengujian

Pada tahap pengujian dilakukan pengujian terhadap perangkat lunak guna memastikan fungsi-fungsi yang ada pada perangkat lunak berjalan dengan semestinya. Pengujian akan dilakukan terhadap perangkat lunak dan pengguna, yakni dengan cara *Black Box* dan *User Acceptance Test* (UAT). Sehingga akan meminimalisir error dan akan diperoleh keluaran yang dihasilkan sesuai dengan yang diinginkan

### Kesimpulan dan Saran

Tahapan kesimpulan dan saran ini merupakan langkah terakhir dari penelitian ini. Akan diperoleh kesimpulan dari penelitian yang dilakukan ini dan akan disebutkan saran-saran yang membangun terhadap penelitian ini agar menimbulkan penelitan baru yang dapat memperbaiki penelitian sebelumnya.

# DAFTAR PUSTAKA

Apoorv, R., & Mathur, P. (2016). Smart attendance management using Bluetooth Low Energy and Android. *2016 IEEE Region 10 Conference (TENCON)*, 1048–1052. https://doi.org/10.1109/TENCON.2016.7848166

Developer, A. (2018). Location Strategies. Retrieved February 5, 2018, from https://developer.android.com/guide/topics/location/strategies.html

Fikry, M., Yusra, & Afrianty, I. (2015). *Analisa dan Perancangan Berorientasi Objek*. Pekanbaru: CV. Asa Riau.

International Data Center. (2017). Smartphone OS Market Share, 2017 Q1. Retrieved February 1, 2018, from https://www.idc.com/promo/smartphone-market-share/os

Istiyanto, J. E. (2013). *Pemrograman Smart Phone Menggunakan SDK Android dan Hacking Android*. Yogyakarta: GRAHA ILMU.

Rosa, A. S., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek*. Bandung: Informatika.

Safaat, N. (2014). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.

Schiller, J., & Voisard, A. (2004). *Location-Based Service*. San Francisco: Morgan Kaufmann.

Soewito, B., Gaol, F. L., Simanjuntak, E., & Gunawan, F. E. (2017). Smart mobile attendance system using voice recognition and fingerprint on smartphone. *Proceeding - 2016 International Seminar on Intelligent Technology and Its Application, ISITIA 2016: Recent Trends in Intelligent Computational Technologies for Sustainable Energy*, 175–180. https://doi.org/10.1109/ISITIA.2016.7828654

Sutheebanjard, P., & Premchaiswadi, W. (2010). QR-Code Generator. *Knowledge Engineering, 2010 8th International Conference on ICT and*. https://doi.org/10.1109/ICTKE.2010.5692920

Swastikasari, M. M., Sediyono, E., & Ardjo, A. S. (2017). Design of E-KOST: An Android-based mobile application using location based service (Study case: SWCU’S students). *2017 International Conference on Innovative and Creative Information Technology (ICITech)*, 1–9. https://doi.org/10.1109/INNOCIT.2017.8319135

Tiwari, S. (2017). An introduction to QR code technology. *Proceedings - 2016 15th International Conference on Information Technology, ICIT 2016*, *1*, 39–44. https://doi.org/10.1109/ICIT.2016.38