

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pesatnya perkembangan teknologi di abad ke-20 ini sangat banyak mempengaruhi kehidupan manusia dalam melakukan berbagai kegiatan, salah satunya di dalam dunia pendidikan. Salah satu teknologi yang cukup dikenal adalah barcode. Pada masa ini, penggunaan barcode sudah tidak asing lagi dalam industri di seluruh dunia. Hal ini memudahkan kegiatan pendidikan dalam mengembangkan ilmu pengetahuan. Barcode menyimpan data spesifik seperti kode produksi, nomor identitas, dan lain-lain sehingga sistem komputer dapat mengidentifikasi informasi yang di kodekan dalam barcode dengan mudah. Hal ini dapat dimanfaatkan dalam efisiensi kegiatan belajar pada sekolah untuk memberikan informasi atau memanfaatkan waktu secara lebih efisien dan efektif melalui barcode tersebut. Seiring dengan berkembangnya teknologi yang begitu pesatnya, penggunaan barcode kini mulai digantikan dengan *Qr code*. *Qr code* adalah image dua dimensi yang merepresentasikan suatu data, terutama data berbentuk teks. *Qr code* merupakan evolusi dari barcode yang awalnya satu dimensi menjadi dua dimensi. *Qr code* memiliki kemampuan menyimpan data yang lebih jauh besar daripada barcode. Dengan penggunaan *Qr code* ini, sebuah sekolah atau organisasi dapat menyajikan informasi yang berkaitan dengan presensi, data siswa, jumlah pelajaran, dan informasi mengenai kepentingan sekolah yang akan mereka gunakan selanjutnya secara digital. Dengan demikian, mereka dapat memaksimalkan waktu presensi. Dengan bertumpu pada penelitian yang dilakukan Elin Herlinah Penerapan QR Code Untuk Sistem Presensi Siswa SMP Berbasis Web.

Peranan yang cukup penting berkembangnya teknologi dan informasi didalam dunia pendidikan yaitu presensi, dalam hal ini presensi merupakan suatu hal yang pokok menjadi parameter bagi seorang guru menilai keaktifan siswa/i di sekolah, presensi juga memiliki peran penting dalam memperhitungkan siswa, layak atau tidaknya siswa tersebut untuk naik kelas maupun tidak. Selain itu siswa memulai presensi pada saat sebelum, sesudah, dan bahkan pada saat pelajaran berlangsung, yang mana ini sangat tidak efektif dan cenderung mengganggu jalannya proses pembelajaran dikelas, sehingga tidak sesuai dengan ekspektasi yang harusnya siswa itu diberi waktu presensi selama yang ditentukan oleh guru, dan setelah itu siswa yang datang terlambat tidak diperbolehkan untuk melakukan presensi, oleh karena itu

dibutuhkanlah suatu sistem yang dapat memenuhi kebutuhan tersebut sehingga presensi dapat lebih mengoptimalkan proses belajar mengajar di sekolah, terutama di SMP Negeri 9 Jakarta. Untuk proses kegiatan presensi berjalan dengan optimal dan efisien, maka penulis membuat aplikasi tersebut.

## **1.2 Rumusan Masalah**

Permasalahan dalam penulisan tugas akhir atau eksperimen ini adalah dalam pembuatan aplikasi ini.

1. Bagaimana merancang aplikasi ini supaya berfungsi dengan semestinya.
2. Bagaimana pengguna bisa menggunakan kamera telephon untuk *scanning Qr code*.
3. Bagaimana cara supaya guru bisa menggunakan aplikasi pada *web* untuk melakukan presensi.

## **1.3 Batasan Masalah**

Ruang lingkup permasalahan pada eksperimen ini hanya terbatas pada hal berikut :

1. Aplikasi presensi hanya bisa untuk presensi siswa.
2. Siswa hanya bisa melakukan presensi namun tidak bisa melihat jumlah presensi.
3. Siswa harus terhubung dengan internet.

## **1.4 Tujuan Penelitian**

Pada eksperimen atau penelitian yang dilakukan saat ini bertujuan untuk :

1. Merancang dan membangun sistem presensi yang lebih kompeten.
2. Meningkatkan efektivitas presensi yang dilakukan oleh sekolah.
3. Membuat saya berkembang dalam merancang dan membangun software yang bisa berguna untuk saya dan orang lain kedepannya.

## **1.5 Manfaat Penelitian**

Manfaat pada penelitian yang akan didapatkan untuk proyek akhir ini ialah :

1. Memberikan pengalaman dan wawasan yang luas tentang merancang dan membangun aplikasi ..
2. Memberikan kemudahan untuk pengguna aplikasi pada penelitian ini.

## 1.6 Metode Penelitian

Metode penelitian yang digunakan pada untuk penelitian tugas akhir adalah metoder air terjun atau yang disebut dengan *waterfall method*.

1. Analisa Kebutuhan

Langkah ini merupakan Analisa terhadap kebutuhan sistem , seperti pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara ataupun studi literatur.

2. Desain sistem

Tahap dimana melakukan penuangan pikiran dan perancangan sistem terhadap solusi dari permasalahan yang ada dengan perangkat pemodelan seperti diagram alir data.

3. Penulisan kode program

Penulisan kode program atau *coding* adalah penerjemahan dalam bahasa yang bisa dikenali oleh komputer. yang biasa dilakukan oleh programmer.

4. Pengujian Program

Tahap dimana sistem yang baru diuji kemampuan dan efektivitasnya sehingga didapatkan kekurangan dan kelemahannya untuk sistem dan akan dikaji ulang.

5. Operasi dan Pemeliharaan

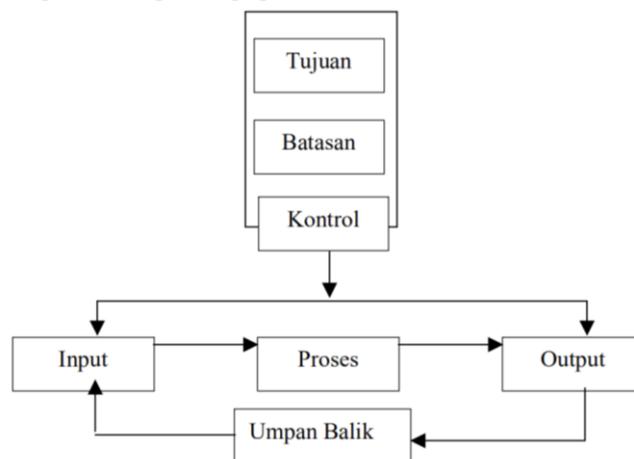
Pada tahap operasi dan pemeliharaan , yang nantinya bisa digunakan oleh pengguna secara baik dan bisa dilakukan pemeliharaan untuk perbaikan *software* atau perangkat

## BAB II

### LANDASAN TEORI

#### 2.1 Sistem Informasi

Pengertian sistem menurut beberapa ahli yaitu, menurut (Tata Sutabri,2012) “pada dasarnya sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu”[1]. Diagram sistem dapat dilihat pada gambar 2.1 dibawah ini:



Gambar 2.1 Elemen-elemen sistem (Andri Kristanto,2003)

Menurut (kristanto, 2003:2). Dari gambar di atas bisa dijelaskan sebagai berikut : tujuan, batasan, dan kontrol sistem akan berpengaruh pada *input*, proses, dan *Output*. *Input* dalam sistem akan diproses dan diolah sehingga menghasilkan *output*, dimana *output* tersebut akan dianalisis dan akan menjadi umpan balik bagi si penerima. Kemudian dari umpan balik ini akan muncul segala macam pertimbangan untuk *input* selanjutnya. Selanjutnya siklus ini akan berlanjut dan berkembang sesuai dengan permasalahan yang ada[2].

#### 2.2 Presensi

Presensi dalam Kamus Besar Bahasa Indonesia (KBBI) berarti kehadiran, artinya adalah suatu cara atau metode untuk menandakan seseorang hadir atau tidak. Sedangkan sistem menurut Kamus Besar Bahasa Indonesia (KBBI) adalah perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas. Maka Sistem presensi merupakan sebuah sistem yang digunakan untuk mengetahui, mencatat maupun merangkum daftar kehadiran setiap anggota instansi tersebut.

Sistem presensi mencatat identitas anggota instansi dan waktu keluar masuk anggotanya. Menurut (Wirahadi, 2019) "Sistem presensi mempunyai kebutuhan penting dalam suatu instansi dimana data yang didapat dari proses pengisian kehadiran bisa digunakan ataupun diolah menjadi salah satu proses penilaian baik atau buruknya kinerja seseorang"[3]. Pada suatu instansi pendidikan seperti universitas, perguruan tinggi maupun sekolah, data dari proses presensi juga sangat berguna sebagai tolak ukur kedisiplinan suatu pegawai ataupun staf pengajar, yang kemudian data tersebut akan dijadikan sebagai salah satu acuan kedisiplinan dalam bekerja.

Dewasa ini perkembangan sistem presensi pun beragam tergantung kebutuhan dan keefektifan dari instansi masing-masing. Adapun diantaranya yaitu :

1. Sistem presensi menggunakan tanda tangan, yang menggunakan tanda tangan diselebar kertas atau menggunakan alat lain sebagai identifikasi kehadiran.
2. Sistem presensi menggunakan *scan barcode*, yang menggunakan kamera *Handphone* untuk melakukan *scan barcode* yang telah disediakan atau ditempel dibagian lokasi kerja tertentu.
3. Sistem presensi menggunakan *fingerprint*, yang menggunakan sidik jari atau *fingerprint* yang dicocokkan pada alat *fingerprint scanner* yang biasanya diletakkan didaerah lokasi pekerjaan sebagai identifikasi kehadiran.
4. Dan sistem presensi lain sebagainya.

### **2.3 Sistem Kehadiran Siswa Menggunakan *Qr code***

Sistem kehadiran yang telah memanfaatkan teknologi *Qr code* telah banyak dikembangkan pada saat ini dengan berbagai jenis perangkat dan metode-metode yang digunakan. Pada penelitian yang dilakukan oleh Norhikmah Dkk (Norhikmah, Azizah, Laili, 2016) menggunakan *Qr code* dalam kehadiran menjadi lebih cepat dan efisien hanya dalam waktu 15 detik/*idcard*, dibandingkan dengan menggunakan tanda tangan diatas kertas selama 30 detik/orang[4].

Penelitian lain yang dilakukan oleh Yustim Dkk (Yustim, Violina, Purnama, 2016) menggunakan *Qr code* pada sistem kehadiran dapat mempercepat dan mempermudah dalam memasukkan data kehadiran mahasiswa dalam kegiatan Proses Belajar Mengajar, dapat menghasilkan informasi yang *up-to-date* terhadap kehadiran mahasiswa serta dapat melakukan monitoring kehadiran mahasiswa. Dan dari kedua penelitian tersebut masih ada beberapa kekurangan atau menjadi Batasan dalam penelitian seperti tidak adanya sistem keamanan dari proses kehadiran baik itu

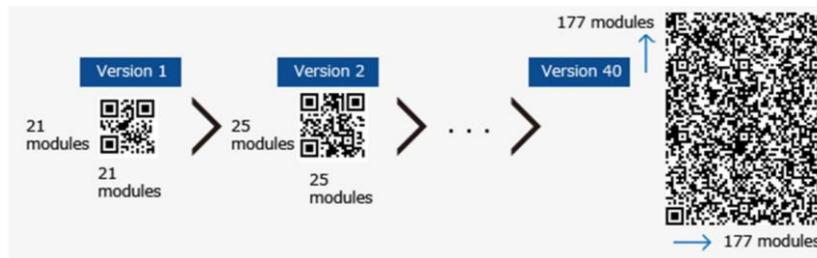
pada saat melakukan *scanning Qr code* sampai dalam pengirim data kehadiran, dan juga tidak ada autentikasi terhadap mahasiswa selaku *user* dari aplikasi tersebut. Dalam penelitian ini, sistem kehadiran siswa akan dibangun menggunakan *Qr code* yang setiap siswa memiliki *user id* yang digunakan untuk pembeda dari siswa yang lainnya[5].

## 2.4 Qr code

Pada penelitian ini hal utama berfokus pada *Qr code* untuk bahan utama pembuatan aplikasi. *Quick Response Code* sering di sebut *Qr code* atau Kode QR adalah semacam simbol dua dimensi yang dikembangkan oleh Denso Wave yang merupakan anak perusahaandari Toyota sebuah perusahaan Jepang pada tahun 1994. Menurut (Denso,2011)Tujuan dari *Qr code* ini adalah untuk menyampaikan informasi secara cepat dan juga mendapat tanggapan secara cepat. Pada awalnya *Qr code* digunakan untuk pelacakan bagian kendaraan untuk manufacturing. Namun sekarang, telah digunakan untuk komersil yang ditujukan pada pengguna telepon 29 seluler. *Qr code* adalah perkembangan dari barcode atau kode batang yang hanya mampu menyimpan informasi secara horizontal sedangkan *Qr code* mampu menyimpan informasi lebih banyak, baik secara horizontal maupun vertikal. *Qr code* biasanya berbentuk persegi putih kecil dengan bentuk geometris hitam, meskipun sekarang banyak yang telah berwarna dan digunakan sebagai brand produk. Informasi yang dikodekan dalam *Qr code* dapat berupa URL, nomor telepon, pesan SMS, V-Card, atau teks apapun (Ashford,2010). *Qr code* telah mendapatkan standarisasi internasional *SO/IEC18004* dan Jepang *JIS-X- 0510*[6] .

### 2.4.1 Versi Qr code

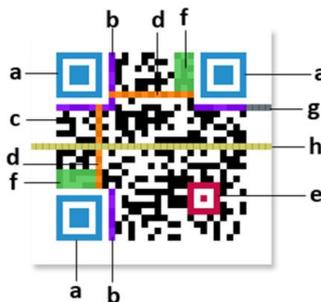
*Qr code* dapat menghasilkan 40 versi yang berbeda dari versi 1 (21 x 21 modul) sampai versi 40 (177 x 177 modul). Tingkatan Versi *Qr code* 1 dan 2 berbeda 4 modul berlaku sampai dengan versi 40. Setiap versi memiliki konfigurasi atau jumlah modul yang berbeda. Modul ini mengacu pada titik hitam dan putih yang membentuk suatu *Qr code*. Setiap versi *Qr code* memiliki kapasitas maksimum data, jenis karakter dan tingkat koreksi kesalahan. Jika Jumlah data yang ditampung banyak maka modul yang akan diperlukan dan menjadikan *Qr code* menjadi lebih besar. Versi dari *Qr code* dapat dilihat pada gambar 2.2 dibawah ini :



Gambar 2.2 Versi *Qr code*(qrcode.com)

### 2.4.2 Anatomi *Qr code*

*Qr code* mempunyai anatomi yang terdiri dari pattern atau model dengan kotak hitam yang bentuknya tersusun sesuai dengan informasi didalamnya. Anatomi *Qr code* dapat dilihat pada gambar 2.3 dibawah ini :



Gambar 2.3 Anatomi *Qr code* (Ariadi 2011)

Penjelasan mengenai anatomi dari *Qr code* Menurut (Ariadi 2011)[7]:

1. *Finder Pattern* berfungsi untuk identifikasi letak *Qr code*.
2. *Format Information* berfungsi untuk informasi tentang *error correction level* dan *mask pattern*.
3. *Data* berfungsi untuk menyimpan data yang dikodekan.
4. *Timing Pattern* merupakan pola yang berfungsi untuk identifikasi koordinat pusat *Qr code*, berbentuk modul hitam putih.
5. *Alignment Pattern* merupakan pola yang berfungsi memperbaiki penyimpangan *Qr code* terutama *distorsi non linier*.
6. *Version Information* adalah versi dari sebuah *Qr code*.
7. *Quiet Zone* merupakan daerah kosong di bagian terluar *qrCode* yang mempermudah mengenali pengenalan *qr* oleh sensor *CCD*.
8. *Qr code version* adalah versi dari *Qr code* yang digunakan.

## 2.5 Mengoreksi Kesalahan Qr code

*Qr code* mampu mengoreksi kesalahan dan pengembalian data dalam pembacaan kode apabila *Qr code* kotor atau rusak. M, Ada 4 tingkatan koreksi kesalahan dalam *Qr code*[6] yang dapat dilihat pada tabel 2.1 dibawah ini :

**Tabel 2.1 Level Koreksi *Qr code***

Level Koreksi Kesalahan	Jumlah Perkiraan Koreksi
L	7%
M	15%
Q	25%
H	30%

Semakin tinggi tingkat koreksi kesalahan semakin besar juga versi *Qr code*. Faktor lokasi dan lingkungan operasi perlu di timbangkan dalam menentukan level *Qr code*. Level Q dan H baik digunakan di pabrik yang kotor, sedangkan L untuk tempat yang bersih. Level yang sering digunakan adalah level M dengan perkiraan koreksi mencapai 15% (qrcode.com).

## 2.6 Manfaat Qr code

Beberapa manfaat yang terdapat pada *Qr code* antara lain:

1. Kapasitas tinggi dalam menyimpan data Sebuah *Qr code* tunggal dapat menyimpan sampai 7.089 angka.
2. Ukuran yang kecil Sebuah *Qr code* dapat menyimpan jumlah data yang sama dengan barcode 1D dan tidak memerlukan ruang besar.
3. Dapat mengoreksi kesalahan Tergantung pada tingkat koreksi kesalahan yang dipilih, data pada *Qr code* yang kotor atau rusak sampai 30% dapat diterjemahkan dengan baik.
4. Banyak jenis data *Qr code* dapat menangani angka, abjad, simbol, karakter bahasa Jepang, Cina atau Korea dan data biner.
5. Kompensasi distorsi *Qr code* tetap dapat dibaca pada permukaan melengkung atau terdistorsi.
6. Kemampuan menghubungkan Sebuah *Qr code* dapat dibagi hingga 16 simbol yang lebih kecil agar sesuai dengan ruang.

Simbol-simbol kecil yang dibaca sebagai kode tunggal apabila di scan menurut urutan.

## 2.7 Android

Menurut (Herlinah,Musliadi,2019)Android merupakan sistem operasi yang banyak digunakan pada perangkat bergerak yang dewasa ini sangat terkenal dan populer digunakan pada ponsel cerdas. Android juga merupakan platform pemrograman yang dikembangkan oleh Google untuk ponsel cerdas dan perangkat seluler lainnya, missal tablet. Android bisa berjalan di beberapa macam perangkat yang dikembangkan oleh banyak vendor ponsel cerdas yang berbeda. Android menyertakan paket pengembangan perangkat lunak untuk penulisan kode asli dan perakitan modul perangkat lunak dalam membuat aplikasi bagi Pengembang Android [8]. Logo dari *Android* dapat dilihat pada gambar 2.4 dibawah ini :



Gambar 2.4 Android logo

Android juga mempunyai komponen yaitu :

1. Activities

Suatu *activity* akan menyajikan *user interface* (UI) kepada pengguna sehingga pengguna dapat melakukan interaksi untuk menjalankan fungsi tertentu. Sebuah aplikasi Android bisa jadi hanya memiliki satu *activity*, tetapi umumnya aplikasi memiliki banyak *activity* tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut.

2. Service,

*Service* tidak memiliki *Graphic User Interface* (GUI), tetapi *service* berjalan secara *background* untuk melakukan operasi-operasi yang *longrunning* (proses yang memakan waktu cukup lama) atau melakukan operasi untuk proses *remote*.

3. Broadcast Reciever

*Broadcast Reciever* berfungsi menerima dan bereaksi untuk menyiapkan *notifikasi*. *Broadcast Reciever* tidak memiliki *user interface (UI)* tapi 33 memiliki sebuah *activity* untuk merespon informasi yang diterima atau kepada pengguna. *Broadcast Receiver* hanyalah pintu gerbang menuju komponen lain dan memang dirancang untuk hanya melakukan kerja seminimal mungkin.

#### 4. Content Provider

*Content Provider* membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. *Content providers* juga berguna untuk membaca dan menulis data yang berstatus *private* dan tidak dibagikan ke suatu aplikasi.

## 2.8 IntelliJ IDE

Perangkat lunak ini digunakan untuk membuat sistem pada aplikasi android yang sudah mendukung beberapa bahasa program seperti javascripct, java dan lain – lain, intellij ide digunakan, karena lebih ringan dibanding perangkat lunak sejenisnya.

Menurut (HF,Irwanto 2018:20) IntelliJ IDEA merupakan IDE (Integrated Development Environment), yaitu alat pengembang terpadu dari JetBrains. Di dalamnya terdapat fasilitas untuk programmer mengembangkan program / aplikasi. IDE ini hadir dari JetBrains yang masuk dalam bisnis development tool selama 15 tahun terkahir dengan sukses besar. IntelliJ IDEA adalah IDE bagi para profesional dan hadir dalam dua edisi yaitu edisi gratis (community) dan edisi ultimate yang menargetkan pengguna para developer enterprise. Edisi gratis hadir dengan banyak fitur untuk membangun aplikasi Android serta aplikasi JVM. Sebenarnya, platform pengembangan Android resmi Google Android Studio didasarkan pada edisi komunitas gratis IntelliJ IDEA. Edisi Ultimate hadir dengan serangkaian fitur paling modern untuk mengembangkan aplikasi *web* dan Java EE enterprise [9].

## 2.9 Javascript

Javascript menurut (Sunyoto,2007:17) “adalah bahasa scripting yang populer di internet dan dapat bekerja di sebagian besar browser popoler seperti Internet Explorer (IE), Mozilla Firefox, Netscape dan Opera. Kode Javascript dapat disisipkan dalam halaman *web* menggunakan tag script” [10].

Beberapa hal tentang Javascript:

1. Javascript didesain untuk menambah interaktif suatu *web*
2. Javascript merupakan sebuah bahasa scripting.
3. Bahasa scripting merupakan bahasa pemrograman yang ringan.
4. Javascript berisi baris kode yang dijalankan di komputer (*web browser*).
5. Javascript biasanya disisipkan (*embedded*) dalam halaman HTML.
6. Javascript adalah bahasa interpreter (yang berarti skrip dieksekusi tanpa proses kompilasi).

Bahasa program ini sangat fleksibel karena selain bisa untuk digunakan dalam aplikasi bahasa ini juga dapat digunakan sebagai bahasa program untuk pembuatan *web*.

## 2.10 Database

Dilihat dari segi bahasa maka database juga dapat disebut basis data. Suatu data dapat dimaksudkan informasi yang diperoleh dan disimpan sedangkan basis adalah semacam perkumpulan atau tempat berkumpul. Maka dapat diambil kesimpulan bahwa arti dari database adalah kumpulan data atau informasi yang diperoleh dan selanjutnya disimpan dalam suatu media, umumnya adalah di computer.

Sedangkan menurut Indrajani :

1. Sebuah kumpulan data yang berkaitan secara logis serta merupakan penjelasan dari data tersebut yang dibuat dengan tujuan agar menemukan data yang dibutuhkan oleh suatu perusahaan dan organisasi. Basis data juga dapat diartikan sebagai kumpulan data yang saling terintegrasi karena basis data dibuat untuk digunakan oleh banyak pengguna, memegang data operasional dan juga penjelasan tentang data tersebut, dan menghindari duplikasi data [11].
2. Sebuah kumpulan elemen data yang terintegrasi serta berhubungan secara logika. Basis data menggabungkan berbagai catatan yang tadinya disimpan dalam file yang terpisah ke dalam suatu elemen data[11].

## 2.11 PostgreSQL

PostgreSQL (obe *and hsu* 2016) dalam buku postgresql: *up and running* “adalah sebuah sistem basis data yang disebarluaskan secara bebas menurut perjanjian lisensi *bsd*”[12]. Piranti lunak ini merupakan salah satu basis data yang paling banyak digunakan saat ini, selain *mysql* dan *oracle*. PostgreSQL menyediakan fitur yang berguna untuk replikasi basis data. fitur-fitur yang disediakan postgresql antara lain

db mirror, pgpool, slony, pgcluster, dan lain-lain.berikut ada tiga *tools* utama dalam pada postgresql yang mendukung penggunaan postgresql bagi developer dan logo untuk PostgreSQL dapat dilihat pada gambar 2.2 dibawah ini :



Gambar 2.5 PostgreSQL logo

1. Psql

Psql adalah sebuah tampilan pada *command-line* yang berisi perintah tertulis (query) pada postgresql.

2. Pgadmin

pgadmin adalah aplikasi grafis yang digunakan sebagai alat untuk menggunakan postgresql. pgadmin merupakan aplikasi desktop dan dapat terhubung ke berbagai server postgresql walaupun dengan sistem operasi yang berbeda-beda.

3. Phppgadmin

phppgadmin adalah alat untuk menggunakan postgresql yang berjalan di *web* seperti phpmyadmin untuk mysq

## 2.12 UML (Unified Modeling Language)

Menurut Rosa Dan Shalahuddin (2015:133) "*UML (Unified Modelling Language)* adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek" [13]..

Jadi UML bisa diartikan sebagai bahasa visual untuk menggambarkan definisi – definisi tentang kebutuhan, membuat analisi dan desain sertai membuat arsitektur dalam penulisan pemrograman yang berorientasikan objek dengan menggunakan teks yang mendukung.

### 2.12.1 Use case Diagram

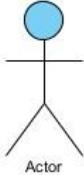
Menurut Rosa dan Shalahuddin (2015: 155) "*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (behavior)

sistem informasi yang akan dibuat”[13]. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut Aktor dan *use case*.

1. Aktor merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram *use case* : Simbol Diagram *Use case* pada tabel 2.2 dibawah ini:

**Tabel 2.1 Usecase diagram**

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling tertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>Use case</i>.</p>
<p>Aktor/actor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang: biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama aktor.</p>
<p>Asosiasi/association</p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use</i></p>

	case atau <i>use case</i> memiliki interaksi dengan aktor.
<p>Ekstensi/extend</p> <p style="text-align: center;">&lt;&lt;extend&gt;&gt;</p> 	<p>Relasi <i>use case</i> tambahan tersebut <i>use case</i> dinamakan <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>
<p>Generalisasi/generalization</p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>

### 2.12.2 Activity Diagram

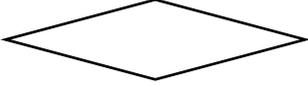
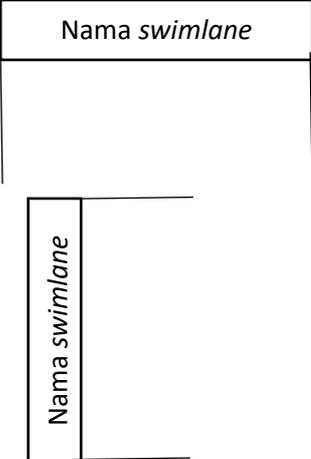
Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak[13].

Menurut Rosa dan Shalahuddin (2015: 161) Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
2. Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kamus ujinya
4. Rancangan menu yang ditampilkan pada perangkat lunak

Berikut adalah simbol-simbol yang ada pada diagram aktivitas dalam tabel 2.3 dibawah ini :

**Tabel 2.3 Activity diagram**

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

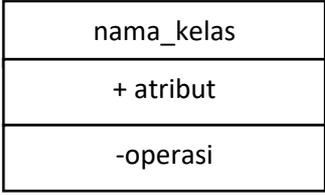
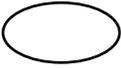
### 2.12.3 Class Diagram

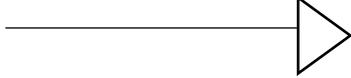
Menurut Rosa dan Shalahuddin (2015: 141) “diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut pola dan metode atau operasi” [13].

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas .
2. Operasi atau metode adalah fungsi yang dimiliki oleh suatu kelas.

Berikut adalah simbol-simbol yang ada pada diagram kelas yang sudah tersedia pada tabel 2.4 dibawah ini:

**Tabel 2.4 Activity diagram**

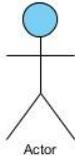
Simbol	Deskripsi
<p>Kelas</p> 	<p>Kelas pada struktur system</p>
<p>Antarmuka / <i>interface</i></p>  <p>Nama_interfacec</p>	<p>Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek</p>
<p>Asosiasi / <i>association</i></p> 	<p>Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i></p>
<p>Asosiasi berarah / <i>directed Association</i></p> 	<p>Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i></p>

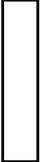
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spealisasi (umum – khusus)
Kebergantungan / <i>dependency</i> 	Kebergantungan antar kelas
Agregasi / <i>aggregation</i>	Relasi antarkelas dengan makna semua-bagian ( <i>whole – part</i> )

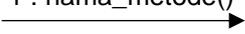
#### 2.12.4 Sequence Diagram

Menurut Rosa dan Shalahuddin (2015: 165) “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek”[13].Berikut adalah simbol-simbol yang ada pada diagram sekuen yang sudah tersedia pada tabel 2.5 dibawah ini :

**Tabel 2.5 Activity diagram**

Simbol	Deskripsi
Aktor  atau <div style="border: 1px solid black; padding: 5px; display: inline-block;"> <u>Nama aktor</u> </div>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari Aktor adalah gambar orang, tapi belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda.

<p>Garis hidup / <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek.</p>
<p>Objek</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p><u>Nama aktor</u></p> </div>	<p>Menyatakan objek yang berinteraksi Pesan.</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya, misalnya :</p> <p>1 : <i>log in ()</i> 2: <i>cekStatusLog in ()</i> 3: <i>open ()</i></p> <p>Maka <i>cekStatusLog in ()</i> dan <i>open ()</i> dilakukan di dalam metode <i>log in ()</i></p> <p>Aktor tidak memiliki waktu aktif</p>
<p><code>&lt;&lt;create&gt;&gt;</code></p> 	<p>objek yang lain, arah panah mengarah pada objek yang dibuat.</p>

<p>Pesan tipe call</p> <p>1 : nama_metode()</p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p> <p>nama_metode()</p> <p>Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.</p>
---	---

### 2.13 React Native

React Native adalah framework open source besutan facebook yang dibuat setelah facebook sebelumnya membuat react.js, React.JS sendiri merupakan sebuah library dari facebook yang dapat digunakan untuk membangun antarmuka pengguna (UI). Jadi, react native adalah framework open source untuk membuat aplikasi multi-plaatform (android, ios dan windows platform "dalam tahap pengembangan") dengan bahasa javascript, sesuai dengan deskripsi di situs resminya "Learn once, write anywhere"[14].

Menurut (B.Eisenman,2015) React Native adalah salah satu framework untuk membuat aplikasi mobile dengan menggunakan kode JavaScript. Framework React Native memiliki seperangkat komponen bagi platform iOS dan Android untuk membangun aplikasi mobile dengan tampilan yang benar-benar seperti native [15]. Dengan menggunakan Framework React Native, kita dapat merender *User Interface* untuk platform iOS dan Android. React Native ini adalah framework open source, yang bisa kompatibel dengan platform lain seperti Windows atau tvOS dalam waktu dekat.

### 2.14 Node JS

Node.js adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis *web* dan ditulis dalam sintaks bahasa pemrograman JavaScript. Bila selama

ini kita mengenal JavaScript sebagai bahasa pemrograman yang berjalan di sisi client / browser saja, maka Node.js ada untuk melengkapi peran JavaScript sehingga bisa juga berlaku sebagai bahasa pemrograman yang berjalan di sisi server, seperti halnya PHP, Ruby, Perl, dan sebagainya. Node.js dapat berjalan di sistem operasi Windows, Mac OS X dan Linux tanpa perlu ada perubahan kode program.

Menurut (Sheba, 2017:8). Berbeda dengan bahasa pemrograman sisi server pada umumnya yang bersifat sinkronis atau blocking, Node.js bersifat asinkronis atau non-blocking sebagaimana halnya JavaScript bekerja. Node.js berjalan dengan basis event (event-driven). Maksud dari blocking secara sederhana adalah bahwa suatu kode program akan dijalankan hingga selesai, baru kemudian beralih ke kode program selanjutnya[16].

## 2.15 Java

Menurut Suyanto (2015:2) Java diciptakan oleh suatu tim yang dipimpin oleh Patrick Naughton dan James Gosling dalam suatu proyek dari sun microsystem yang memiliki kode green dengan tujuan untuk menghasilkan bahasa komputer sederhana yang dapat dijalankan di peralatan sederhana dengan tidak terikat pada arsitektur tertentu.[17] mulanya disebut oak, tetapi karena oak sendiri merupakan nama dari bahasa pemrograman komputer yang sudah ada, maka sun mengubahnya menjadi java.sun kemudian meluncurkan browser dari java yang disebut hot java yang mampu menjalankan applet. Setelah itu teknologi java diadopsi oleh Netscape yang memungkinkan program java dijalankan di browser netscape yang kemudian diikuti Internet Explore. Karena keunikan dan kelebihanannya, teknologi java mulai menarik banyak vendor seperti IBM, Symantec, Inprise, dll. Sun merilis versi awal java secara resmi pada awal tahun 1996 yang kemudian terus berkembang hingga muncul jdk 1.1 kemudian jdk 1.2 yang mulai disebut sebagai versi java2 karena banyak mengandung peningkatan dan perbaikan. Perubahan utama adalah swing yang merupakan teknologi GUI (Graphical User Interface) yang mampu menghasilkan windows yang portable. Dan pada tahun 1998-1999 lahirlah teknologi J2EE (Java 2 Enterprise Edition).

## 2.16 Blackbox Testing

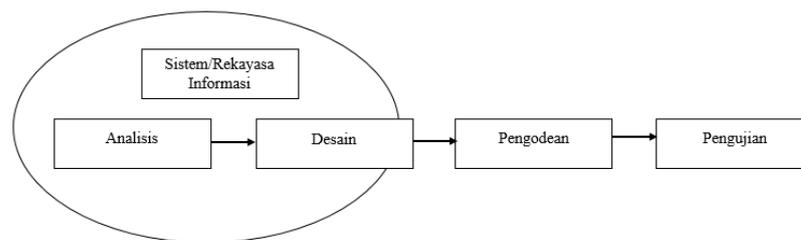
*Blackbox testing* adalah tahap yang digunakan untuk menguji kelancaran program yang telah dibuat. Pengujian ini penting dilakukan agar tidak terjadi kesalahan alur program yang telah dibuat. Menurut Rosa dan Salahuddin "*Blackbox testing* yaitu

menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program”[13].

Jadi *blackbox testing* merupakan pengujian terhadap fungsionalitas atau kegunaan sebuah aplikasi yang telah dibuat. Sehingga bisa mengetahui hasil dari sebuah sistem apakah berjalan dengan baik atau masih memiliki masalah untuk diperbaiki.

## 2.17 Model SDLC Air Terjun

Menurut Rosa & Shalahuddin Model SDLC air terjun (*Waterfall*) sering juga disebut model sekuensi linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (*support*)[18]. Berikut adalah gambar model air terjun pada gambar 2.6 dibawah ini :



Gambar 2.6 Model Ilustrasi waterfall (Rosa & Shalahuddin, 2016)

1. Analisis kebutuhan perangkat lunak  
Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.
2. Desain  
Desain perangkat lunak adalah proses multilangkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan .
3. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus adaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

