BAB II

LANDASAN TEORI

2.1 Android (Sistem Operasi)

Android adalah sebuah sistem operasi yang ditargetkan untuk *smartphone* dan tablet. Android dibuat oleh Android, Inc. yang kemudian pada tahun 2005 dibeli dan dikembangkan oleh Google, Inc. sampai seperti sekarang. (verizonwireless.com)

2.2 Android Studio

Android Studio adalah Integrated Development Environment (IDE) resmi dari Android untuk pengembangan aplikasi Android, diresmikan pada tanggal 16 Mei 2013 di Google I/O conference.

Gambar 2.1 Logo Android Studio
Sumber: https://en.wikipedia.org/wiki/Android_Studio

Android Studio berbasis IntelliJ IDEA dari JetBrains dan menggunakan Bahasa Java. Selain *code editor* dan *tools* pengembangan IntelliJ yang kuat, Android Studio menawarkan lebih banyak fitur yang meningkatkan produktivitas ketika membangun aplikasi Android, seperti:

1. *Build system* berbasis Gradle yang fleksibel.

2. *Emulator* yang cepat dan kaya fitur.
3. Lingkungan terpadu yang dapat digunakan untuk mengembangkan aplikasi untuk semua perangkat Android.
4. Alat pengujian dan *framework* yang ekstensif.
5. Instant Run untuk menggabungkan perubahan pada aplikasi yang sedang berjalan tanpa membangun APK baru.
6. Lint untuk menangkap kinerja, kegunaan, kompatibilitas versi dan masalah lainnya.
7. *Template* kode dan integrasi GitHub untuk membantu membangun fitur aplikasi umum dan *import* contoh kode.
8. Mendukung C++ dan NDK.
9. *Built-in support* untuk Google Cloud Platform.

(developer.android.com)

INSTITUT BISNIS & INFORMATIKA stikom SURABAYA

2.3 Java

Aplikasi Android dikembangkan menggunakan bahasa Java. Sampai sekarang, itu adalah satu-satunya pilihan Anda untuk membuat aplikasi dasar. Java adalah bahasa pemrograman yang sangat populer yang dikembangkan oleh Sun Microsystems (sekarang dimiliki oleh Oracle). Dikembangkan lama setelah C dan C++, Java menggabungkan banyak fitur canggih dari bahasa-bahasa yang kuat sementara menangani beberapa kelemahan mereka. Namun, bahasa pemrograman hanya sekuat library mereka. Library ini ada untuk membantu pengembang membangun aplikasi.

Beberapa fitur Java:

1. Mudah dipelajari dan dipahami.
2. Didesain untuk menjadi *platform-independent* dan aman dengan menggunakan mesin *virtual*.
3. Pemrograman *object-oriented.*

(code.tutsplus.com)

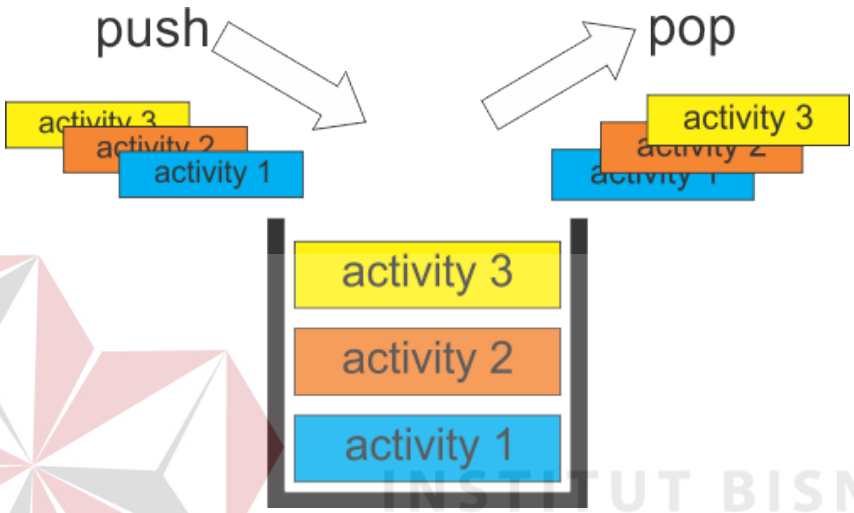
2.4 XML (*eXtensible Markup Language*)

Extensible Markup Language (XML) adalah seperangkat aturan untuk pengkodean dokumen dalam bentuk yang dapat dibaca oleh mesin (*machine-readable*). XML adalah format populer untuk berbagi data di internet. *Website* yang sering memperbarui konten mereka, seperti situs berita atau *blog*, sering memberikan XML *feed* sehingga program eksternal dapat mengikuti perubahan konten (developer.android.com).

2.5 *Activity*

Activity adalah komponen aplikasi yang menyediakan layar dimana pengguna dapat berinteraksi untuk melakukan sesuatu. Sebuah aplikasi android biasanya memiliki banyak *activity* (Zapata, 2015: 16). Setiap *activity* diberikan sebuah jendela untuk *user interface*-nya. Biasanya, salah satu kegiatan dalam aplikasi ditetapkan sebagai *Main Activity*. *Main Activity* adalah *activity* yang ditampilkan pertama kali kepada pengguna ketika aplikasi dijalankan. Setiap *activity* kemudian dapat memulai *activity* lain untuk melakukan tindakan yang

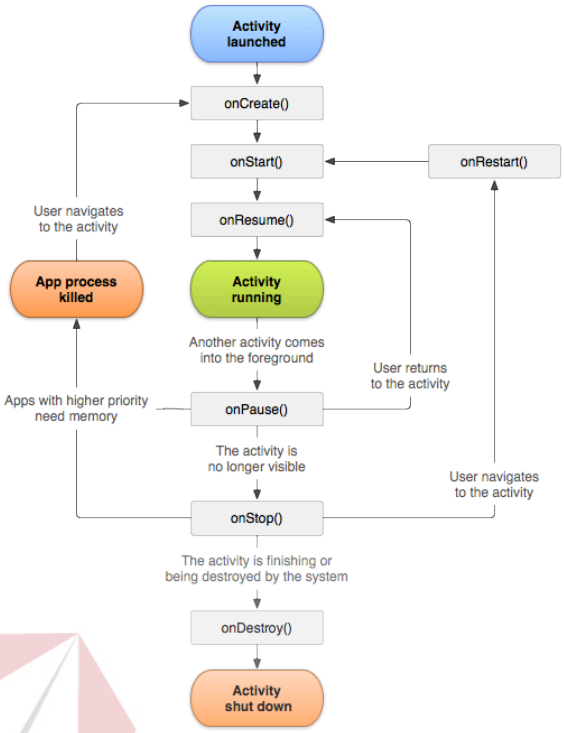
berbeda. Setiap kali *activity* baru dimulai, *activity* sebelumnya dihentikan, tetapi sistem mempertahankan *activity* dalam sebuah tumpukan (*stack*). *Stack activity* memiliki mekanisme *Last In First Out*. Untuk lebih jelas tentang mekanisme *Last In First Out*, perhatikan Gambar 2.2 dibawah.



Gambar 2.2 Mekanisme *Stack Last In First Out*

Ketika *activity* baru dimulai, *activity* akan didorong (*push*) ke dalam stack. Jadi, ketika pengguna selesai dengan *activity* saat ini dan menekan tombol Kembali (*Back*), *activity* tersebut akan dikeluarkan (*pop*) dan dihancurkan (*destroy*) kemudian *activity* sebelumnya akan dilanjutkan (*resume*) (developer.android.com). Jadi *activity* akan ditumpuk dengan urutan yang pertama muncul dan ditindih dengan *activity* yang muncul kedua, dan seterusnya, begitupun ketika akan menampilkan kembali *activity* sebelumnya, *activity* yang tampil saat ini akan dihancurkan dan ditampilkan *activity* dengan urutan sebelumnya dalam *stack*. *Activity* memiliki siklus hidup yang dapat dilihat pada Gambar 2.3.

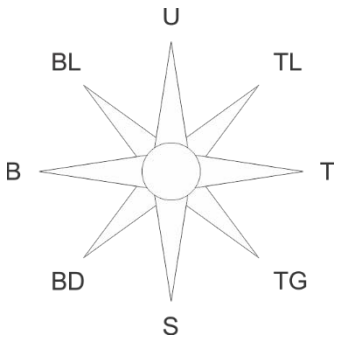
INSTITUT BISNIS & INFORMATIKA stikom SURABAYA



Gambar 2.3 Siklus Hidup *Activity*
Sumber: https://developer.android.com/guide/components/activities.html

2.6 Mata Angin

Mata angin adalah sebuah acuan arah yang biasa digunakan dalam sistem navigasi, dan kompas. Secara umum, mata angin memiliki 8 arah, dengan 4 arah sebagai arah utama.



Gambar 2.4 Arah Mata Angin

Berikut adalah arah mata angin secara umum beserta derajatnya:

1. Utara (U) : 0°
2. Timur laut (TL) : 45°
3. Timur (T) : 90°
4. Tenggara (TG) : 135°
5. Selatan (S) : 180°
6. Barat daya (BD) : 225°
7. Barat (B) : 270°
8. Barat laut (BL) : 315°

2.7 *Minimap*

Menurut yang dikemukakan oleh Adams (2013), *Minimap* adalah suatu peta kecil, yang diperbarui secara dinamis, umumnya ditampilkan pada sudut layar dalam mode permainan primer. Pemain menjadi titik tengah pada *minimap*, dengan orientasi yang berubah sesuai dengan ke arah mana pemain menghadap biasa disebut radar screen. Untuk contoh *minimap* dapat dilihat pada Gambar 2.5.

Gambar 2.5 *Minimap* Pada Permainan Battlefield 4

INSTITUT BISNIS & INFORMATIKA stikom SURABAYA

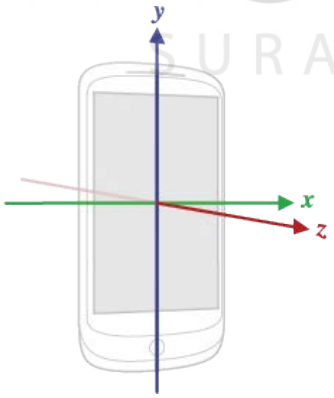
2.8 *Accelerometer Sensor*

Accelerometer Sensor atau Sensor *Accelerometer* dapat membaca percepatan perangkat dalam 3 sumbu sensor. Pengukuran ini dilaporkan dalam X, Y dan Z. *Accelerometer* bekerja dengan sistem koordinat standar. Untuk memperjelas tentang koordinat standar, perhatikan Gambar 2.6. Jika perangkat diletakkan pada posisi datar maka:

a. Jika perangkat digerakkan ke arah kanan pengguna, maka nilai percepatan X positif, dan jika perangkat digerakkan ke arah kiri pengguna, maka nilai percepatan X negatif

b. Jika perangkat digerakkan menjauh dari pengguna, maka nilai percepatan Y positif, dan jika perangkat digerakkan mendekati pengguna, maka nilai percepatan Y negatif

c. Jika perangkat diangkat keatas dengan percepatan A m/s^2 maka nilai percepatan Z adalah A + 9.81



Gambar 2.6 Sistem Koordinat Standar Pada *Smartphone*
Sumber: https://source.android.com/devices/sensors/sensor-types.html

Secara umum, *accelerometer* adalah sensor yang tepat digunakan untuk mengetahui gerak perangkat. Hampir setiap perangkat *mobile* yang menggunakan sistem

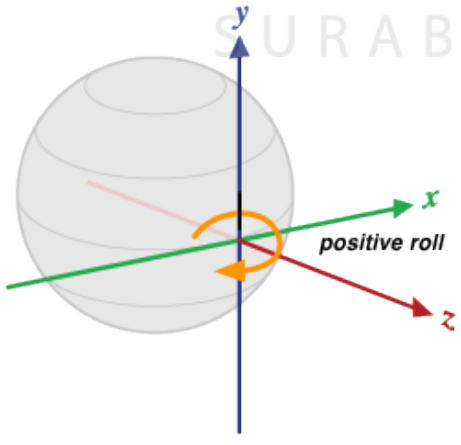
operasi Android, baik *smartphone* ataupun tablet memiliki *accelerometer*, ditambah *accelerometer* menggunakan daya sepuluh kali lebih kecil dibandingkan dengan sensor gerakan yang lain (developer.android.com). Dalam Tugas Akhir ini sensor *accelerometer* digunakan sebagai pendeteksi gerakan.

2.9 *Orientation Sensor*

Orientation Sensor atau Sensor Orientasi dapat membaca perubahan sudut perangkat. Pengukuran ini dilaporkan dalam X, Y, dan Z.

a. Nilai **X** adalah *azimuth*, yaitu selisih sudut antara arah utara magnetik dan sumbu **X**, dengan mengikuti sumbu **Z**. Nilai *azimuth* berkisar antara 0 dan 360, dengan 0=Utara, 90=Timur, 180=Selatan, 270=Barat.

b. Nilai **Y** adalah *pitch*, yaitu putaran dengan sumbu **X** sebagai pusatnya. Nilai *pitch* berkisar antara -180 dan 180, bernilai positif apabila sumbu **Z** bergerak ke arah sumbu **Y**.



Gambar 2.7 Sumbu X, Y, Dan Z Pada *Orientation Sensor*
Sumber: https://source.android.com/devices/sensors/sensor-types.html

c. Nilai **Z** adalah *roll*, yaitu putaran dengan sumbu **Y** sebagai pusatnya. Nilai *roll* berkisar antara -90 dan 90, bernilai positif apabila sumbu **X** bergerak menjauh dari sumbu **Z**

(developer.andoid.com)

Dalam Tugas Akhir ini sensor orientasi digunakan untuk mengetahui arah hadap pengguna.

INSTITUT BISNIS & INFORMATIKA stikom SURABAYA