

## BAB II

### LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Dalam landasan teori ini akan diuraikan tentang hasil penelitian lainnya yang pernah dilakukan dan mempunyai kaitan topik dengan penelitian yang penulis lakukan, diantaranya :

1. Penelitian yang dilakukan oleh Muhamad Nurul Huda pada tahun 2019 dalam penelitian yang berjudul “Aplikasi *Tracking* Paket Berbasis *Android*”, dimana pada penelitian tersebut membahas bagaimana mengembangkan sebuah *system* yang memiliki fungsi mendeteksi keberadaan/koordinat dari *GPS Tracker* yang diterapkan kedalam *mobile platform*.
2. Penelitian yang dilakukan oleh Ramos Somya pada tahun 2018 dalam penelitian yang berjudul “Sistem *Monitoring* Kendaraan Secara *Real Time* Berbasis *Android* menggunakan Teknologi *CouchDB* di PT.Pura Barutama”, dimana pada penelitian ini membahas bagaimana cara membuat *system* monitoring kendaraan secara *real time* yang dapat diakses langsung oleh masing-masing unit produksi dengan menggunakan *Android*.
3. Penelitian yang dilakukan oleh Pasha Dani Pamungkas, Dharu Arseno,S.T.,M.T., Edwar,S.T.,M.T. pada tahun 2018 dalam penelitian yang berjudul “Analisa Perbandingan Sinkronisasi Berbasis Sistem *Global Positioning System* (GPS) dan *Global Navigation System* (GLONASS)” dimana pada penelitian tersebut membahas perbandingan antara GPS dan GLONASS yang mana pada penelitian tersebut GPS lebih baik dari pada GLONASS.
4. Penelitian yang dilakukan oleh I Made Oka Widyantara, I Gede Agus Krisna Warmayana, Linawati pada tahun 2015 dalam penelitian yang berjudul “Penerapan Teknologi *GPS Tracker* Untuk Identifikasi Kondisi *Traffik Jalan Raya*” dimana pada penelitian tersebut membahas tentang cara kerja *GPS Tracker* yang dimana

GPS Tracker memiliki fungsi untuk mengetahui posisi kendaraan secara real time.

5. Penelitian yang dilakukan oleh Nhort Hosea Hutagaol, Hendra Kurniawan, S.kom., M.Sc.Eng, Ferdi Chahyadi, S.Kom., M.Cs pada tahun 2017 dalam penelitian yang berjudul “Aplikasi Absensi Tenaga Marketing Menggunakan Metode Location Based Service (LBS) Berbasis Android” dimana pada penelitian tersebut membahas bagaimana cara kerja *Location Base Service* (LBS) dan membahas bagaimana Mengimplementasikan *Location Base Service* (LBS) kedalam *Mobile Platform*.

## 2.2 *Geolocation Maps*

*Geolocation* yaitu kemampuan untuk menentukan posisi suatu objek atau seorang manusia menggunakan koordinat geografisnya (garis lintang dan garis bujur) dalam suatu peta melalui berbagai teknologi yang tersedia (Anwar & Afwani, 2016).

## 2.3 *GPS (Global Position System)*

GPS adalah singkatan dari *Global Positioning System*, yang merupakan sistem navigasi dengan menggunakan teknologi satelit yang dapat menerima sinyal dari satelit (Alfeno & Cipta, 2017). Cara kerja GPS secara logik ada 5 langkah:

1. Memakai perhitungan “*triangulation*” dari satelit.
2. Untuk perhitungan “*triangulation*”, GPS mengukur jarak menggunakan *travel time* sinyal radio.
3. Untuk mengukur *travel time*, GPS memerlukan akurasi waktu yang tinggi.
4. Untuk perhitungan jarak, kita harus tahu dengan pasti posisi satelit dan ketinggian pada orbitnya.
5. Terakhir harus mengoreksi *delay* sinyal waktu perjalanan di *atmosfer* sampai diterima *receiver*.

## **2.4 Lock GPS**

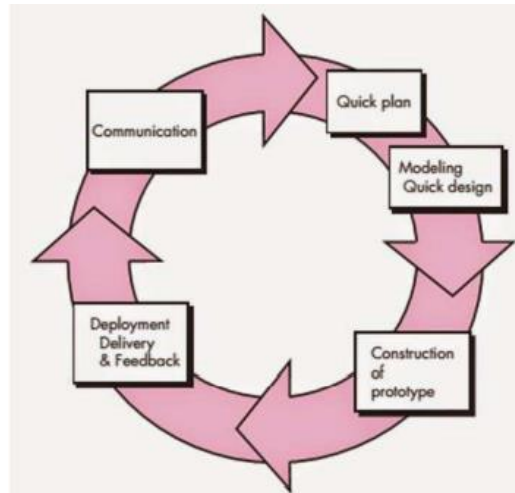
*Lock GPS* yaitu kondisi dimana *GPS receiver* sudah terhubung dengan sinyal satelit dan mendapatkan sinyal informasi dari satelit. Sebuah *GPS receiver* harus mengunci sinyal minimal tiga satelit untuk menghitung posisi 2D (*latitude* dan *longitude*) dan jalur pergerakan. Jika *GPS receiver* dapat menerima empat atau lebih satelit, maka dapat menghitung posisi 3D (*latitude, longitude dan altitude*). Jika sudah dapat menentukan posisi *user*, selanjutnya *GPS* dapat menghitung informasi lain, seperti kecepatan, arah yang dituju, jalur, tujuan perjalanan, jarak tujuan, dan masih banyak lagi (Akbar & Prabowo, 2015).

## **2.5 Javascript**

Menurut Suyanto (2009) *Javascript* adalah bahasa yang ditempatkan pada kode HTML (*HypertextMarkupLanguage*). Bahasa ini kemampuan dokumen HTML (*HypertextMarkupLanguage*) menjadi lebih luas sebagai mana contoh digunakan untuk *validasi* pada formulir sebelum diproses ke tahap berikutnya. dan dapat membuat interaktif dan juga bisa untuk menambah desain *web*. *Javascript* mempunyai macam library atau framework *front end* yaitu *Angular, VueJS, dan React*.

## **2.6 Metode Prototyping**

Metode *prototyping* merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi pengguna secara cepat. Berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai. *Prototyping* tersebut akan dievaluasi oleh pelanggan/pemakai dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak (Susanto & Andriana, 2016). Tahapan model pengembangan *prototyping* ini dapat dilihat pada gambar 2.1 berikut:



**Gambar 2.1** Model Pengembangan *Prototyping*

Sumber : (Ramos Somya, 2018)

## 2.7 *API (Application Programming Interface)*

*(API) Application Programming Interface* adalah sebuah bahasa dan pesan format yang digunakan oleh program aplikasi untuk berkomunikasi dengan sistem operasi atau program pengendalian lainnya seperti sistem manajemen database (DBMS) atau protokol komunikasi. *(API) Application Programming Interface* diimplementasikan dengan menulis panggilan fungsi dalam program, yang menyediakan hubungan ke subrutin yang diperlukan untuk eksekusi (Bachtiar Arief Pahlevy, 2016).

## 2.8 *JSON (JavaScript Object Notation)*

Menurut Dwija Wisnu Brata (2015) *JSON (JavaScript Object Notation)* adalah merupakan format untuk memasukan data kedalam sebuah variable struktur data yang universal, dalam artian bisa digunakan dalam berbagai bahasa pemrograman. Hampir semua bahasa pemrograman mendukung *JSON (JavaScript Object Notation)* dalam berbagai format.

Hal ini memungkinkan format data yang dapat dipertukarkan menggunakan bahasa pemrograman juga menggunakan dasar dari struktur *JSON (JavaScript Object Notation)*. *JSON* merupakan bagian dari bahasa pemrograman javascript.

## 2.9 **Pengujian Sistem**

Menurut Pressman (2012) pengujian perangkat lunak adalah elemen kritis dari jaminan perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi,

desain dan pengodean. Selain itu, pengujian juga dapat diartikan sebagai sebuah proses eksekusi suatu program dengan maksud menentukan kesalahan.

## **2.10 Pengujian**

Menurut Iqbal (2016), Pengujian sistem yaitu:

ISO/IEC 25010 merupakan model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang *software engineering. Product quality* ini juga digunakan untuk tiga model kualitas yang berbeda untuk produk perangkat lunak antara lain:

1. Kualitas dalam model penggunaan,
2. Model kualitas produk, dan
3. Data model kualitas

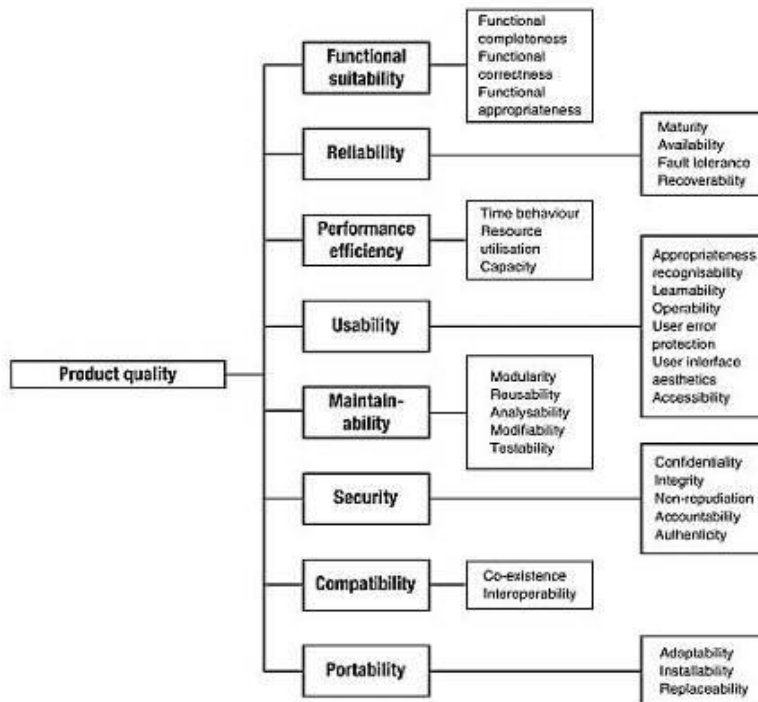
Model kualitas produk terdiri dari delapan karakteristik yang berhubungan dengan sifat statis perangkat lunak dan sifat dinamis dari sistem komputer. Model ini berlaku untuk sistem komputer dan produk perangkat lunak. Karakteristik yang didefinisikan oleh kedua model tersebut relevan untuk semua produk perangkat lunak dan sistem komputer. Karakteristik dan subkarakteristik memberikan terminologi yang konsisten untuk menentukan, mengukur dan mengevaluasi kualitas sistem dan perangkat lunak. Mereka juga menyediakan seperangkat karakteristik kualitas yang sesuai dengan persyaratan kualitas yang dapat dibandingkan untuk kelengkapan (ISO /IEC, 2011).

## **2.11 ISO/IEC 25010**

ISO/IEC 25010 adalah tahap yang digunakan untuk menguji kelancaran program yang telah dibuat. Pengujian ini penting dilakukan agar tidak terjadi kesalahan alur program yang telah dibuat.

ISO/IEC merupakan standar yang digunakan oleh dunia internasional untuk melakukan evaluasi atau penguku-ran kualitas dari perangkat lunak. ISO/IEC yang digunakan dalam penelitian ini adalah versi 25010 yang merupa-kan versi lanjutan dari ISO/IEC 9126 dengan penambahan beberapa struktur dan bagian dari standar model kualitas. Secara keseluruhan ISO/IEC 25010 memiliki 8 karakteristik untuk mengukur kualitas perangkat lunak secara me-nyeluruh, antara lain *portability, performance efficiency, reliability, security usability,*

*maintainability, compatibil-ity, dan functional suitability.* Berikut adalah tampilan dari 8 karakteristik ISO/IEC 25010 pada Gambar 2.2 :



**Gambar 2.2** Pengujian ISO 25010

**Sumber:** Fadli dan Siti Rochimah (2019)

Menurut Gunawan dan Trianto (2017), karakteristik ISO/IEC 25010 adalah:

### 1. **Functional Suitability**

Sejauh mana perangkat lunak mampu menyediakan fungsi yang memenuhi kebutuhan yang dapat digunakan dalam kondisi tertentu. Karakteristik ini dibagi menjadi beberapa karakteristik yaitu.

- a. *Functional completeness*, sejauh mana fungsi yang disediakan mencakup semua tugas dan tujuan pengguna secara spesifik.
- b. *Functional correctness*, sejauh mana produk atau sistem menyediakan hasil yang benar sesuai kebutuhan.
- c. *Functional appropriateness*, sejauh mana fungsi yang disediakan mampu memfasilitasi penyelesaian tugas dan tujuan tertentu.

### 2. **Compatibility**

Sejauh mana sebuah produk, sistem atau komponen dapat bertukar informasi dengan produk, sistem atau komponen dan/atau menjalankan fungsi

lain yang diperlukan secara bersamaan ketika berbagi perangkat keras dan environment perangkat lunak yang sama. Karakteristik ini dibagi menjadi 2 karakteristik yaitu.

- a. *Co-existence*, sejauh mana produk atau sistem dapat menjalankan fungsi yang dibutuhkan secara efisien sementara berbagi sumber daya dengan produk atau sistem yang lain tanpa merugikan produk atau sistem tersebut.
- b. *Interoperability*, sejauh mana dua atau lebih produk, sistem atau komponen dapat bertukar informasi dan menggunakan informasi tersebut.

### **3. Usability**

Sejauh mana sebuah produk atau sistem dapat digunakan oleh user tertentu untuk mencapai tujuan dengan efektif, efisiensi, dan kepuasan tertentu dalam konteks penggunaan. Karakteristik ini terbagi menjadi beberapa karakteristik yaitu.

- a. *Appropriateness recognizability*, sejauh mana pengguna dapat mengetahui apakah sistem atau produk sesuai kebutuhan mereka.
- b. *Learnability*, sejauh mana produk atau sistem dapat digunakan oleh pengguna untuk mencapai tujuan tertentu yang belajar menggunakan sistem atau produk dengan efisien, efektif, kebebasan dari resiko dan kepuasan dalam konteks tertentu.
- c. *Operability*, sejauh mana produk atau sistem mudah dioperasikan dan dikontrol.
- d. *User error protection*, sejauh mana produk atau sistem melindungi pengguna terhadap membuat kesalahan.
- e. *User interface aesthetics*, sejauh mana antarmuka pengguna dari produk atau sistem memungkinkan interaksi yang menyenangkan dan memuaskan pengguna.
- f. *Accessibility*, sejauh mana produk atau sistem dapat digunakan oleh semua kalangan untuk mencapai tujuan tertentu sesuai konteks penggunaan.

#### **4. Reliability**

Sejauh mana sebuah sistem, produk atau komponen dapat menjalankan fungsi tertentu dalam kondisi tertentu selama jangka waktu yang ditentukan. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Maturity*, sejauh mana produk atau sistem mampu memenuhi kebutuhan secara handal di bawah keadaan normal.
- b. *Availability*, sejauh mana produk atau sistem siap beroperasi dan dapat diakses saat perlu digunakan.
- c. *Fault tolerance*, sejauh mana produk atau sistem tetap berjalan sebagaimana yang dimaksud meskipun terjadi kesalahan pada perangkat keras atau perangkat lunak.
- d. *Recoverability*, sejauh mana produk atau sistem mampu dapat memulihkan data yang terkena dampak secara langsung dan menata ulang kondisi sistem seperti yang diinginkan ketika terjadi gangguan.

#### **5. Security**

Sejauh mana sebuah produk atau sistem melindungi informasi dan data sehingga seseorang atau sistem lain dapat mengakses data sesuai dengan jenis dan level otorisasi yang dimiliki. Karakteristik ini terbagi menjadi beberapa karakteristik yaitu.

- a. *Confidentiality*, sejauh mana produk atau perangkat lunak memastikan data hanya bisa diakses oleh mereka yang berwenang untuk memiliki akses.
- b. *Integrity*, sejauh mana produk atau perangkat lunak mampu mencegah akses yang tidak sah untuk memodifikasi data.
- c. *Non-repudiation*, sejauh mana peristiwa atau tindakan dapat dibuktikan telah terjadi, sehingga tidak ada penolakan terhadap peristiwa atau tindakan tersebut.
- d. *Accountability*, sejauh mana tindakan dari suatu entitas dapat ditelusuri secara unik untuk entitas.



- e. *Authenticity*, sejauh mana identitas subjek atau sumber daya dapat terbukti menjadi salah satu yang diklaim.

## **6. *Portability***

Sejauh mana keefektifan dan efisiensi sebuah sistem, produk atau komponen dapat dipindahkan dari satu perangkat keras, perangkat lunak atau digunakan pada lingkungan yang berbeda. Karakteristik ini dibagi menjadi beberapa karakteristik yaitu.

- a. *Adaptability*, sejauh mana produk atau sistem dapat secara efektif dan efisien disesuaikan pada perangkat lunak, perangkat keras dan lingkungan yang berbeda.
- b. *Installability*, sejauh mana produk atau sistem dapat berhasil dipasang atau dihapus dalam lingkungan tertentu.
- a. *Replaceability*, sejauh mana produk atau sistem dapat menggantikan produk atau sistem lain yang ditentukan untuk tujuan yang sama pada lingkungan yang sama.

## **7. *Performance Efficiency***

Kinerja relatif terhadap sumber daya yang digunakan dalam kondisi tertentu. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Time behaviour*, sejauh mana respon dan pengolahan waktu produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
- b. *Resource utilization*, sejauh mana jumlah dan jenis sumber daya yang digunakan oleh produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
- c. *Capacity*, sejauh mana batas maksimum parameter produk atau sistem dapat memenuhi persyaratan.

## **8. *Maintainability***

Sejauh mana keefektifan dan efisiensi dari sebuah produk atau sistem dapat dirawat. Karakteristik ini terbagi menjadi beberapa subkarakteristik yaitu.

- a. *Modularity*, sejauh mana sistem terdiri dari komponen terpisah sehingga perubahan atau modifikasi pada salah satu komponen

tersebut memiliki dampak yang kecil terhadap komponen yang lain.

- b. *Reusability*, sejauh mana aset dapat digunakan lebih oleh satu sistem atau digunakan untuk membangun aset lain.
- c. *Analyzability*, tingkat efektivitas dan efisiensi untuk mengkaji dampak perubahan pada satu atau lebih bagian-bagian produk atau sistem, untuk mendiagnosis kekurangan atau penyebab kegagalan produk, untuk mengidentifikasi bagian yang akan diubah.
- d. *Modifiability*, sejauh mana produk atau sistem dapat dimodifikasi secara efektif dan efisien tanpa menurunkan kualitas produk yang ada.
- e. *Testability*, tingkat efektivitas dan efisiensi untuk membentuk kriteria uji dari produk, sistem atau komponen dan uji dapat dilakukan untuk menentukan apakah kriteria tersebut telah terpenuhi.

### **2.11.1 Skala Likert Testing**

Menurut Sugiyono (2018), *Skala likert testing* adalah :

Skala *Likert* digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Dalam penelitian fenomena sosial ini telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel penelitian.

Dengan *skala likert* maka variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator variabel tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen yang dapat berupa pernyataan atau pertanyaan.

Jawaban setiap item instrumen yang menggunakan *skala Likert* mempunyai gradasi dari sangat positif sampai sangat negatif, yang dapat berupa kata-kata antara lain pada tabel 2.1:

Tabel 2.1 Instrumen *Skala Likert*

Keterangan	Skor
Sangat Setuju	5
Setuju	4
Ragu-ragu	3
Tidak setuju	2
Sangat tidak setuju	1