

BAB 2

LANDASAN TEORI

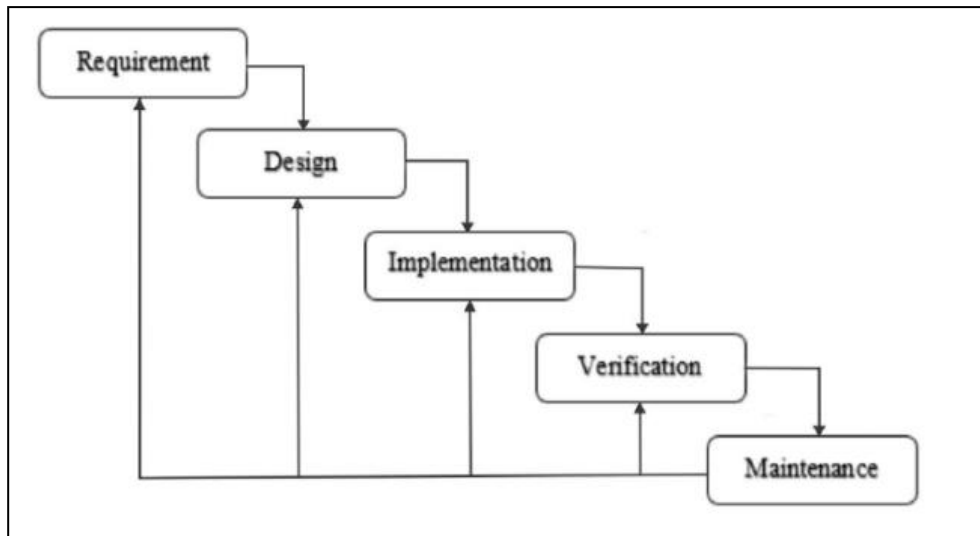
2.1 Emergency Alert

Emergency Alert merupakan suatu aplikasi berbasis android yang digunakan untuk mengirim informasi darurat kepada kerabat terdekat yang berada dikontak aplikasi dan petugas layanan darurat secara cepat dan tepat. Selain itu sistem ini juga melakukan penerapan pada teknologi *location base service* dan *threat button* dimana *button* ini berfungsi sebagai tanda pemberitahuan atau alarm yang akan mengirimkan pesan untuk memberitahu adanya musibah atau penyakit yang sedang terjadi saat itu kepada orang terdekat yang berada dikontak aplikasi atau melakukan panggilan kepada petugas layanan darurat. Orang yang berada pada kontak aplikasi akan menerima sebuah pesan darurat yang berisikan link *user current location* dan jika link tersebut ditekan maka akan diarahkan ke aplikasi google maps untuk mengetahui *user current location* yang sedang membutuhkan bantuan. Selain itu, kelebihan dari sistem ini untuk petugas layanan darurat juga dapat digunakan sebagai sarana peningkatan kinerja untuk segera melakukan tindakan pro-aktif (Anike & Melo, 2019).

2.2 Metode Waterfall

Metode Waterfall merupakan salah satu model System Development Life Cycle (SDLC) yang sering digunakan dalam pengembangan perangkat lunak dan prosesnya dilakukan secara sistematis serta berurutan seperti air terjun. Metode Waterfall diperkenalkan pertama kali oleh Winston Royce pada Tahun 1970. Metode pengembangan ini bersifat linear yang tahapannya dimulai dari tahapan

perencanaan hingga tahapan pemeliharaan. Tahapan metode waterfall dapat dilihat pada Gambar 2.1 (Wahid, Oktober 2020).



Gambar 2.1 Tahapan Metode Waterfall (Wahid, Oktober 2020).

1. Requirement

Tahap ini merupakan tahap analisis kebutuhan sistem perangkat lunak dalam menentukan seluruh kebutuhan yang ada pada sistem secara lengkap. Analisis kebutuhan sistem juga digunakan untuk mendukung dan membantu jalannya proses pembuatan sistem. Analisis kebutuhan sistem dibagi menjadi dua yaitu analisis kebutuhan secara fungsional dan secara non fungsional (Sudarsono & Krisnawati, 2014). Analisis kebutuhan fungsional merupakan analisa mengenai sebuah kebutuhan dari fitur-fitur yang akan diterapkan ke dalam aplikasi sedangkan analisis kebutuhan non fungsional merupakan sebuah kebutuhan yang dilakukan untuk mengetahui spesifikasi kebutuhan secara software/ hardware yang dinilai dari kelayakan sebuah system untuk mendukung pengerjaan pada program yang akan dirancang dan dibangun.

2. Design

Pada tahap ini merupakan translasi dari kebutuhan sistem perangkat lunak ke representasi *design* dengan membuat sebuah arsitektur sistem secara keseluruhan.

3. Implementation

Pada tahap ini, perancangan desain pada perangkat lunak direalisasikan kedalam *code* untuk membentuk suatu program yang dibutuhkan berdasarkan tahap sebelumnya.

4. Verification

Pada tahap ini merupakan proses verifikasi dan pengujian pada program yang sudah dibuat. Pengujian ini dimaksudkan untuk mengecek dan melihat kelancaran dari aplikasi yang sudah dibuat.

5. Maintenance

Tahap maintenance ini merupakan tahap akhir dari metode waterfall yang digunakan dalam pemeliharaan aplikasi yang sudah diuji dan sudah berjalan dengan baik sesuai dari fungsinya.

Kelebihan metode waterfall dapat dilihat pada penjelasan dibawah ini (Wahid, Oktober 2020).

1. Dalam pelaksanaannya dilakukan secara bertahap dengan menghasilkan kualitas sebuah sistem yang baik.
2. Pada proses pengembangannya dilakukan secara berurutan sehingga meminimalisir kesalahan yang mungkin akan terjadi.
3. Dokumentasi yang dihasilkan akan tersusun dengan rapih dan lengkap sebelum melanjutkan ke proses berikutnya.

Kekurangan metode waterfall dapat dilihat pada penjelasan dibawah ini (Wahid, Oktober 2020).

1. Dalam pelaksana pengembangannya membutuhkan waktu yang cukup lama.
2. Jika terjadi kesalahan pada tahap awal dalam proses pengembangan, maka akan sulit untuk melakukan pengulangan dan dapat berakibat di tahap berikutnya.

2.3 SMS Alert

Short Message Service (SMS) merupakan salah satu media untuk mengirimkan atau mendapatkan informasi pesan singkat berupa teks tanpa harus terikat pada tempat dan waktu melalui mobile device seperti handphone (Pratama, Anton, & Firmansyah, 2015). Layanan produk SMS ini dapat dikelompokkan menjadi 3 macam yaitu SMS premium, SMS *broadcast* dan SMS *gateway*. Pada SMS *alert* (pengingat) masih masuk ke dalam lingkup layanan SMS *gateway* (Siregar & Taufik, 2017). SMS *gateway* merupakan layanan pengiriman atau penerimaan pesan ke jaringan selular dengan menyebarkan pesan ke ratusan nomor secara otomatis (Bulan & Wibowo, 2017) . Namun pada SMS *alert* ini hanya berfokus pada *blast* dalam pengiriman pesan darurat. Istilah *alert* juga sering digunakan dalam keadaan darurat dengan mengirimkan sinyal atau tanda darurat. Oleh karena itu, dengan adanya SMS alert ini informasi akan dapat terkirim lebih cepat dan tepat serta pengiriman pesan ini dapat menjadi lebih efektif dan efisien dalam pemberitahuan informasi darurat kepada orang yang sudah terdaftar di dalam kontak aplikasi *user*. Tujuan SMS alert ini agar bisa sebagai penyelamat untuk kondisi tertentu sebelum melanjutkan proses berikutnya (Bulan & Wibowo, 2017)

2.4 Black Box Testing

Black box testing merupakan metode pengujian dimana pengujian perangkat lunak didasarkan oleh spesifikasi kebutuhan functional. Pengujian dalam *black box testing* tidak dilihat dari desain dan *source code* program, tetapi berfokus kepada input dan output yang dihasilkan oleh sistem yang diuji. *Black box testing* juga merupakan solusi praktis untuk peningkatan akurasi dalam memperbaiki *error* yang ada (Cholifah, Yulianingsih, & Sagita, 2018).

Black box testing dilakukan berdasarkan kebutuhan dari pengguna sehingga kebutuhan yang kurang atau kebutuhan tidak terduga dapat diidentifikasi dengan mudah. *Black box testing* membutuhkan *test case* yang dirancang berdasarkan kebutuhan dari sistem yang diuji. *Test case* tersebut pada umumnya berisi kondisi pengujian, skenario pengujian, data pengujian yang digunakan sebagai input, hasil uji yang diharapkan, dan hasil uji sesungguhnya.

Keunggulan dari *black box testing* adalah penguji tidak memerlukan pengetahuan *programming* dan implementasinya. Keunggulan lainnya adalah *black box testing* dapat membantu dalam mengidentifikasi kebutuhan yang ambigu atau inkonsisten (Nidhra & Dondeti, 2012).

2.5 Technology Acceptance Model

Technology Acceptance Model (TAM) merupakan teori sistem informasi yang berlandaskan pada kepercayaan (belief), sikap (attitude), niat (intention) dan hubungan perilaku pengguna (user behavior relationship). Dimana proses ini memodelkan bagaimana pengguna dapat melihat, memakai dan menerima dari manfaat dan kemudahan penggunaan teknologi informasi ini (Irawati, Rimawati, & Pramesti, 2019).

Menurut Davis, TAM menggunakan *variabel Perceived ease of use* dan *variabel Perceived usefulness* dalam menentukan seberapa cepat dan efisien pengguna dapat beradaptasi dengan teknologi. Serta makna *Perceived ease of use* didefinisikan sebagai ukuran dimana menggunakan sistem tertentu akan menyebabkan pengguna bebas dari usaha. Sedangkan *Perceived usefulness* didefinisikan sebagai ukuran dimana menggunakan sistem tertentu akan meningkatkan performa pekerjaan dari pengguna sistem tersebut (Davis, 1989).

2.6 Skala Likert

Skala Likert merupakan skala pengukuran yang dikembangkan oleh Likert (1932). Jamieson (2004) menyatakan bahwa kategori respon pada skala likert mempunyai tingkatan tetapi jarak diantara kategori tidak dapat dianggap sama, sehingga skala likert adalah kelas skala ordinal. Pada kelas ini, statistik yang dapat digunakan adalah *median* atau *modus* untuk mendapatkan hasil pemutusannya.

Pengukuran dapat dilakukan dengan memberikan pertanyaan kepada responden dan kemudian diminta untuk memilih dari lima pilihan jawaban, dimana nilai jawaban memiliki nilai jawaban yang berbeda (Janti, 2014).

Lima pilihan jawaban yang dapat dipilih adalah sangat setuju (SS), setuju (S), netral (N), tidak setuju (TS), dan sangat tidak setuju (STS) pada pernyataan yang positif maupun pernyataan negatif (Sappaile, 2007).

Tabel 2.1 Kriteria Skala Likert (Sugiyono, 2011)

<i>Kategori</i>	<i>Kriteria</i>	<i>Syarat</i>
<i>SS</i>	Sangat Setuju	$P \geq 80\%$
<i>S</i>	Setuju	$60\% \leq P < 80\%$
<i>N</i>	Netral	$40\% \leq P < 60\%$
<i>TS</i>	Tidak Setuju	$20\% \leq P < 40\%$
<i>STS</i>	Sangat Tidak Setuju	$0\% \leq P < 20\%$

Bedasarkan Tabel Kriteria Skala Likert, Rumus yang digunakan untuk menghitung presentase skor adalah sebagai berikut (Sugiyono, 2011)

$$P = \frac{(SS*5)+(S*4)+(N*3)+(TS*2)+(STS*1)}{5+n} * 100\%$$

Dengan P = presentase skor skala Likert

SS = jumlah jawaban sangat setuju

S = jumlah jawaban setuju

N = jumlah jawaban netral

TS = jumlah jawaban tidak setuju

STS = jumlah jawaban sangat tidak setuju

n = jumlah responden