



# BAB 1

## PENDAHULUAN

---

### 1.1 Latar Belakang

Saat ini para nasabah bank sudah dimudahkan dengan fasilitas perbankan yang modern. Salah satu fasilitas tersebut adalah *Automated Teller Machine* (ATM). Dengan menggunakan ATM, nasabah dapat melakukan berbagai aktivitas perbankan seperti penarikan tunai, transfer, pembayaran tagihan, dll. ATM dari berbagai bank khususnya yang ada di kota Bandung jumlahnya sangat banyak, hal tersebut terkadang menyulitkan nasabah dalam mencari lokasi ATM terdekat yang sesuai dengan bank yang dituju. Padahal ATM merupakan salah satu objek publik yang vital. Untuk itu diperlukan suatu solusi dalam bentuk aplikasi yang dapat mencari lokasi ATM terdekat sesuai dengan bank yang ingin dituju.

Perkembangan teknologi internet sekarang ini sudah sangat pesat. Buktinya adalah berkembangnya teknologi *Web Service*. Dengan menggunakan teknologi *Web Service*, maka *service* tersebut bisa diimplementasikan kembali oleh aplikasi mobile yang mana aplikasi tersebut memudahkan nasabah dalam mencari info lokasi ATM terdekat. *Web Service* tersebut menyediakan *service* dengan arsitektur *REST* dimana arsitektur tersebut lebih mudah di akses oleh aplikasi mobile dibandingkan dengan arsitektur *Web Service* yang lain. Aplikasi mobile tadi akan mengakses *service* tersebut melalui *API* sebagai interface dalam menggunakan *service*. Lalu timbul berbagai permasalahan, menentukan lokasi ATM serta menampilkannya ke mobile dan menentukan jarak ATM terdekat serta rute dari sebuah lokasi yang dituju. Salah satu perusahaan yang sangat terdepan dalam memanfaatkan *service* geografis adalah Google Maps.



Google Maps dengan fitur-fiturnya yang menyajikan data berupa map serta lokasi-lokasi suatu dan tidak hanya itu saja tetapi google juga menyediakan rute dari info lokasi yang telah dipilih. Fitur-fitur itu sendiri sangat digemari masyarakat pada umumnya terutama pada masyarakat pengguna telepon genggam yang berkeinginan simple dalam mengakses Google Maps. Melihat peluang ini para perusahaan-perusahaan platform telepon genggam besar seperti *Intenetwork Operating System (IOS)*, *Research In Motion (RIM)* bahkan Android seperti berlomba menggunakan aplikasi google maps tersebut sebagai salah satu *service* unggulan mereka.

Melihat serta mempertimbangkan kedua aspek diatas, maka dari itu dibutuhkan sebuah aplikasi baru sebagai kemudahan masyarakat dalam menemukan lokasi ATM, melihat jarak ATM yang terdekat serta menampilkan rute perjalanan ke ATM tersebut dengan memanfaatkan *API* yang tentunya hasil pemanfaatan layanan google dan *service*.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang diatas maka dapat disimpulkan rumusan masalah pada proyek akhir ini adalah :

1. Bagaimana menangkap & menampilkan lokasi data *API ATM* yang telah diterima dari sisi server di platform Android serta memperlihatkannya di Android dengan bantuan Google Maps?
2. Bagaimana menentukan fungsi jarak *ATM* yang telah dipilih *user*?
3. Bagaimana menampilkan rute dari *ATM* yang dipilih dengan bantuan Google?



### 1.3 Tujuan

Tujuan dari proyek akhir ini adalah :

1. Membuat sebuah aplikasi yang menampilkan info lokasi *ATM* di Android.
2. Menampilkan jarak terdekat antara user dengan tujuan lokasi *ATM*.
3. Menampilkan rute perjalanan antara user dengan tujuan lokasi *ATM*.
4. Mempermudah masyarakat dalam mencari lokasi *ATM* terdekat.

### 1.4 Batasan Masalah

Adapun batasan masalah pada proyek akhir ini adalah :

1. Aplikasi yang dikerjakan menerima dan mengambil suatu *service* dari server dengan pemanfaatan *API*.
2. Data yang diterima berupa *API* yang berisi lokasi-lokasi *ATM*.
3. *ATM* yang dipakai yaitu 3 *ATM* terpopuler di Indonesia, *ATM BNI*, *ATM BRI* dan *ATM Mandiri*.
4. Data yang diterima berupa *XML type*
5. Fungsi yang digunakan berupa pencarian lokasi, pencarian jarak terdekat serta rute.
6. Jenis Perangkat OS Android yang digunakan yaitu Android versi 2.2 (*Froyo* : *Frozen Yoghurt*).
7. *Froyo (Frozen Yoghurt)* dimana di versi ini dijalankan di versi di atasnya seperti contoh versi 2.3 (*Gingerbread*), 3.0 (*Honeycomb*), dll dapat dijalankan. Tetapi di versi sebelumnya seperti 2.1 (*Éclair*), 1.6 (*Donut*), dll tidak dapat dijalankan.



## 1.5 Definisi Operasional

Dalam pembuatan aplikasi proyek akhir ini, untuk mendapatkan data-data yang diterima dari sisi server, harus terhubung dengan server tersebut entah melalui internet atau *wireless* dengan IP tertentu. Lalu sisi server akan mengembalikan dalam bentuk longitude dan longitude tertentu yang telah disimpan di database. Dan komunikasi antara server dan client itu yang disebut dengan "API".

Serta untuk menemukan lokasi user dimana server dapat membaca koordinatnya yaitu dengan *A-GPS (Assisted Global Positioning System)* yang mana *A-GPS* adalah tipe *GPS* yang digunakan spesifik di telephone genggam yang sesuai dengan bawaan dari Android SDK nya yaitu *Froyo*. Sedangkan di Emulator hanya menggunakan *DDMS* sebagai pengiriman posisi dari user dengan melakukan pengiriman latitude dan longitude nya.

Lalu di tahap terakhir output yang akan ditampilkan berupa pemetaan tempat-tempat yang akan dituju user dengan menggunakan Google Maps setelah data yang user minta telah diproses oleh server sehingga lokasi-lokasi *ATM* tersebut muncul.

## 1.6 Metode Pengerjaan

Metode yang dipakai untuk proyek akhir ini terdiri dari langkah-langkah sebagai berikut :

### 1.6.1 Studi Pustaka

Melakukan studi kepustakaan terhadap referensi yang berkaitan dengan proyek akhir ini. Topik yang akan dikaji meliputi :

- a) Tata cara serta aturan pengambilan data dengan metode *REST*
- b) Tata cara pembuatan aplikasi berbasis Android
- c) Tata cara penulisan bahasa pemrograman Java sebagai bahasa mesin yang digunakan oleh Android itu sendiri.
- d) Software-software pendukung yang digunakan.



### 1.6.2 Perancangan Sistem

Setelah berbagai referensi dan kajian sudah didapatkan selanjutnya masuk kepada tahap perancangan. Perancangan system ini hampir keseluruhan melakukan desain serta coding.

Proses desain akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat coding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi interface dan detail (algoritma) procedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*. Dokumen inilah yang akan digunakan *programmer* untuk melakukan aktivitas pembuatan sistemnya.

*Coding* merupakan penerjemahan desain dalam bahasa yang bisa dikenali computer oleh computer. Dilakukan oleh *programmer* yang akan menterjemahkan apa yang diminta oleh *user*. Tahapan ini-lah yang merupakan tahapan secara nyata dalam mengerjakan sesuatu system. Dalam artian penggunaan computer akan dimaksimalkan dalam tahapan ini.

### 1.6.3 Testing & Debug

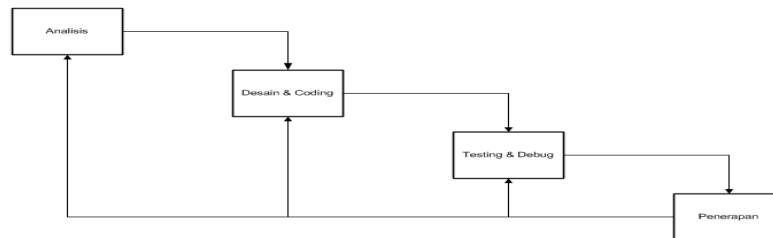
Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki. Berikut saya paparkan testing yang akan dibuat.

- 1) Testing akan dilakukan di dalam emulator Android.
- 2) Lalu melakukan sambungan ke server dengan koneksi internet atau dengan koneksi *wired/wireless* dengan menyamakan IP.
- 3) Lalu yang terakhir testing semua fungsi serta fitur agar tidak terjadi *error* atau kesalahan sistem.



### 1.6.4 Penyusunan Laporan Serta Penerapan

Setelah dilakukan testing dan sudah tidak ada lagi kesalahan atau *error* pada system baru-lah memasuki tahapan ini yaitu penyusunan laporan. Segala sesuatu yang telah dikerjakan diawal akan dituliskan kedalam sebuah laporan serta setelah melakukan beberapa analisa, design dan pengkodean maka system yang sudah jadi akan diterapkan oleh *user*. Berikut keterangan berupa gambar metode *Waterfall*.



Gambar 1.1 Model Waterfall

### 1.7 Jadwal Pengerjaan

Dalam pengerjaan proyek ini akan dibuat jadwal pengerjaan sebagai tahap penyelesaian proyek. Jadwal pengerjaan didasari dari metodologi yang telah dijabarkan diatas. Adapun yang menjadi jadwal pengerjaan dalam perancangan proyek ini adalah sebagai berikut:

Tabel 1.1 Jadwal Pengerjaan

Kegiatan	Mei 2012				Juni 2012				Juli 2012				Agustus 2012			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Analisis Kebutuhan	■	■	■	■												
Seminar Proyek Akhir					■											
Perancangan Aplikasi					■	■										
Pembuatan Kode Aplikasi						■	■	■	■	■						
Pengujian											■					
Sidang Proyek Akhir													■	■		