

## BAB II

### TINJAUAN PUSTAKA

#### A. Penelitian Terdahulu

Penelitian serupa pernah dilakukan oleh Steven Pangestu dkk (Pragestu et al., 2015) dengan judul Implementasi *Augmented reality* dengan Memanfaatkan *GPS Based Tracking* pada Sistem Pengenalan Gedung Universitas Tanjungpura, dimana dalam penelitian yang dilakukannya mereka memanfaatkan teknologi GPS pada *smartphone* untuk menentukan koordinat lokasi gedung dan menampilkan hasilnya melalui teknologi *Augmented reality* berupa nama gedung dan jarak gedung dengan *user* melalui kamera secara *realtime*. AR yang diimplementasikan pada aplikasi bekerja dengan cara memberikan label pada gedung yang ditampilkan oleh layar *smartphone*. Pada penelitian ini dilakukan pengujian akurasi aplikasi terhadap objek dan pengambilan kuesioner terhadap 100 responden.

Bagasworo Wijaya Wisnu dan kawan-kawan juga melakukan penelitian yaitu membangun suatu aplikasi pemandu wisata pada kota Jepara secara *mobile* berbasis *android* yang dapat menampilkan data lokasi berupa objek *augmented reality* (Wisnu, Handoko, & Wahyu, 2016).

Penelitian terdahulu berikutnya dilakukan oleh (Sagala, 2019) yaitu membangun sebuah aplikasi yang dapat membantu masyarakat khususnya yang berdomisili di kota Medan, dan wisatawan untuk mengetahui informasi tempat dan lokasi bangunan bersejarah yang ada di kota Medan

dengan memanfaatkan teknologi *augmented reality* pada judul “Penerapan *Augmented reality* Untuk Menampilkan Label Bangunan Bersejarah Di Kota Medan Menggunakan *Markerless GPS Based Tracking*”.

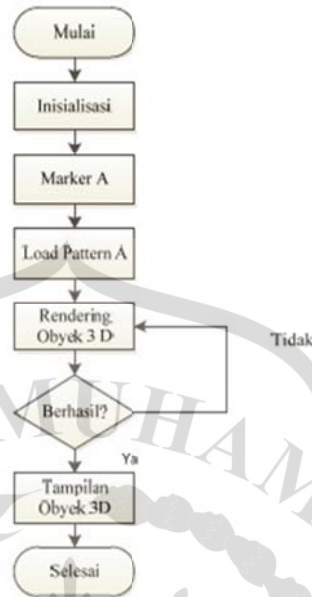
## **B. Landasan Teori**

Landasan teori dalam penelitian ini sebagai berikut:

### **1. *Augmented reality***

*Augmented reality* sebagai penggabungan benda-benda nyata dan maya di lingkungan nyata, berjalan secara interaktif dalam waktu nyata, dan terdapat integrasi antarbenda dalam tiga dimensi, yaitu benda maya terintegrasi dalam dunia nyata. Penggabungan benda nyata dan maya dimungkinkan dengan teknologi tampilan yang sesuai, interaktivitas dimungkinkan melalui perangkat-perangkat input tertentu, dan integrasi yang baik memerlukan penjejakan yang efektif (Suhandra, Purwandari, & Efendi, 2018).

Terdapat dua metode AR yang dikembangkan saat ini yaitu *Marker Based Tracking* dan *Markerless AR*. *Marker Based Tracking* adalah AR yang menggunakan marker atau penanda objek dua dimensi yang memiliki suatu pola yang akan dibaca komputer melalui media *webcam* atau kamera yang tersambung dengan komputer. *Marker* biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih (Haryani & Triyono, 2017).



**Gambar 2.1 Diagram Alir *Marker Based Tracking* (Haryani & Triyono, 2017)**

*Metode Markless AR* adalah metode AR dimana pengguna tidak perlu lagi mencetak sebuah marker untuk menampilkan elemen-elemen digital. Marker yang dikenali berbentuk posisi perangkat, arah, maupun lokasi. Deteksi ciri target objek dan pelacakan target objek berdasarkan pose kamera digunakan sebagai kriteria dalam proses pelacakan tanpa marker pada teknologi AR (Yudiantika, Sulisty, & Hantono, 2014).

## 2. *Location Based Service*

Layanan Berbasis Lokasi atau lebih dikenal dengan *Location Based Service* (LBS) istilah umum yang digunakan untuk menggambarkan teknologi yang digunakan untuk menemukan lokasi perangkat yang kita gunakan. LBS adalah layanan informasi yang

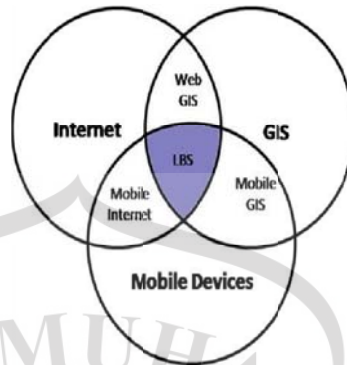
dapat diakses melalui *mobile device* dengan menggunakan *mobile network*, yang dilengkapi kemampuan untuk memanfaatkan lokasi dari *mobile device* tersebut (Safaat H, 2013). Terdapat dua unsur utama pada LBS yaitu:

a) *Location Manager (API Maps)*

Menyediakan *tools/source* untuk LBS, *Application Programming Interface (API Maps)* menyediakan fasilitas untuk menampilkan, memanipulasi *maps/peta* beserta *feature - feature* lainnya seperti tampilan satelit, *street (jalan)*, maupun gabungannya. Paket ini berada pada *com.google.android.maps*.

b) *Location Provider (API Location)*

Menyediakan teknologi pencarian lokasi yang digunakan oleh *device/perangkat*. *API Location* berhubungan dengan data *GPS (Global Positioning System)* dan data lokasi *real-time*. *API Location* berada pada paket *android* yaitu dalam paket *android.location*. Dengan *Location Manager*, kita dapat menentukan lokasi kita saat ini, *track gerakan/perpindahan*, serta kedekatan dengan lokasi tertentu dengan mendeteksi *perpindahan* (Safaat H, 2013).



**Gambar 2. 2 Teknologi *Location Based Services* (Budiman, 2016)**

*Location Based Service* dapat digambarkan sebagai suatu layanan yang berada pada pertemuan tiga teknologi yaitu: *Geographic Information System, Internet Service, dan Mobile Devices*. Teknologi *Location Based Services* berfokus bagaimana menentukan posisi dari peralatan yang Anda gunakan atau disebut dengan metode *positioning*. Jika berbicara tentang *positioning*, beberapa sistem yang digunakan ini dapat dibedakan dalam tiga jenis (Safaat H, 2013), yaitu:

1) Secara Manual

Metode manual adalah cara-cara konvensional yang selama ini sering dilakukan baik yang melalui *yellow pages*, telepon bantuan operator, dan sebagainya. Cara-cara ini tentunya sangat merepotkan dan sudah mulai ditinggalkan. Munculnya internet memberi perspektif baru bagi fasilitas

pencarian ini yang lebih luas. Ditambah dengan munculnya komunikasi selular, tentunya mobilitas pun semakin meningkat. Dengan menggabungkan kedua teknologi ini, sudah pasti metode manual akan ditinggalkan.

2) Melalui GPS (*Global Positioning System*)

GPS merupakan system navigasi radio di seluruh dunia yang memanfaatkan 24 satelit beserta stasiun bumi. Melalui system ini, bumi dibagi menjadi kotak-kotak dengan masing-masing memiliki alamat yang unik sehingga dapat mengidentifikasi dengan tepat setiap lokasi tersebut.

3) Melalui *Cellular Based Station* atau BTS (*Base Transceiver Station*)

Teknologi *Cellular Based Station*, berbasis pada jaringan telekomunikasi selular yang memungkinkan digunakan dalam ruangan atau indoor. Sebuah handphone dapat ditentukan posisinya saat ini. Berdasarkan posisi relative terhadap sebuah handphone dalam satu atau lebih *cell tower* terdekat dengan mempertimbangkan sinyal yang digunakan untuk melayani sebuah handphone. Dengan menggunakan prinsip triangulasi, posisi handphone dapat terdeteksi. *Cellular Based Station* memiliki akurasi yang sangat kurang baik jika dibandingkan dengan menggunakan GPS.

### 3. Aplikasi Web

Web aplikasi (*Web Application*) merupakan aplikasi yang dapat diakses melalui browser dengan menggunakan jaringan internet atau intranet. Web Aplikasi juga sebuah perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis web seperti HTML, CSS, JavaScript, Ruby, Python, Php, Java serta bahasa pemrograman lainnya (Markey, 2018).

Web aplikasi ini dapat digunakan untuk berbagai macam tujuan dan manfaat yang berbeda-beda. Banyak dari perusahaan-perusahaan berkembang yang menggunakan aplikasi berbasis web untuk mengelola data dan informasi dalam merencanakan sumber daya perusahaan.

Selain memiliki banyak kegunaan di berbagai perusahaan, keunggulan kompetitif dari aplikasi berbasis web adalah akselerasinya yang sangat ringan, sedikit sumber data, serta dapat diakses secara cepat melalui browser pada koneksi internet ke server.

Markey (2018) menyebutkan bahwa ada beberapa keunggulan dari aplikasi berbasis web antara lain:

- Mudah untuk diakses dan dikembangkan
- Tidak memerlukan spesifikasi yang tinggi untuk melakukan dan menggunakannya.
- Tidak juga memerlukan lisensi atau izin ketika menggunakan aplikasi berbasis web.

- Fleksibilitas yang mana dapat menyesuaikan pada berbagai devices dan sistem operasi.
- Setup server pada aplikasi berbasis web ini juga cukup mudah.

#### 4. AR.js

AR.js adalah pustaka JavaScript yang memungkinkan pengembang untuk memperkenalkan fitur dan fungsionalitas AR ke situs web apa pun menggunakan beberapa baris HTML (Egington, 2019).

Proyek ini *open-source* dan dengan hampir 13.000 bintang di repositori AR.js GitHub, ini digunakan oleh semua jenis pengembang untuk menciptakan pengalaman digital yang lebih inovatif.

Kerangka kerja AR.js mencakup kompatibilitas lintas-browser dan mendukung WebGL dan WebRTC, yang berarti berfungsi di seluruh perangkat Android dan iPhone di atas iOS 11.

Membungkus sejumlah kerangka kerja AR yang berbeda termasuk three.js, a-frame dan ARToolKit, AR.js membuatnya sangat sederhana dan mudah untuk membawa AR ke aplikasi web.

Beberapa fitur AR.js:

1. Kompatibilitas lintas-browser
2. Performa tinggi hingga 60fps bahkan pada perangkat lama
3. Berbasis web tanpa perlu instalasi
4. *Open Source* dan dapat diakses secara gratis

5. Bekerja di semua perangkat seluler dengan WebGL dan WebRTC
6. Tidak perlu perangkat keras tambahan atau tidak biasa
7. Dapat dilakukan dengan <10 baris HTML.

## 5. Node JS

Node.js merupakan salah satu peranti pengembang yang bisa digunakan untuk membuat aplikasi berbasis *Cloud*. Node.js dikembangkan dari engine JavaScript yang dibuat oleh Google untuk browser Chrome / Chromium (V8) ditambah dengan libUV serta beberapa pustaka internal lainnya. Dengan menggunakan Node.js, semua pengembangan akan dilakukan menggunakan JavaScript, baik pada sisi klien maupun server. Node.js dibuat pertama kali oleh Ryan Dahl ([twitter.com/ryah](https://twitter.com/ryah)) dan sampai saat ini dikembangkan oleh komunitas sebagai software bebas dengan pendanaan utama dari Joyent, perusahaan tempat Ryan Dahl bekerja (Purnomosidi, 2013).

## 6. MongoDB

MongoDB adalah salah satu *software* NoSQL yang termasuk dalam kategori Document Store / Document-Oriented Database, yaitu data disimpan dalam bentuk dokumen. Suatu dokumen bisa diibaratkan seperti suatu *record* dalam basis data relasional dan isi dari masing-masing dokumen tersebut bisa berbeda-beda dan ada pula yang sama. Hal ini berbeda dengan basis data relasional yang menetapkan keseragaman kolom serta tipe data dengan data yang NULL jika tidak

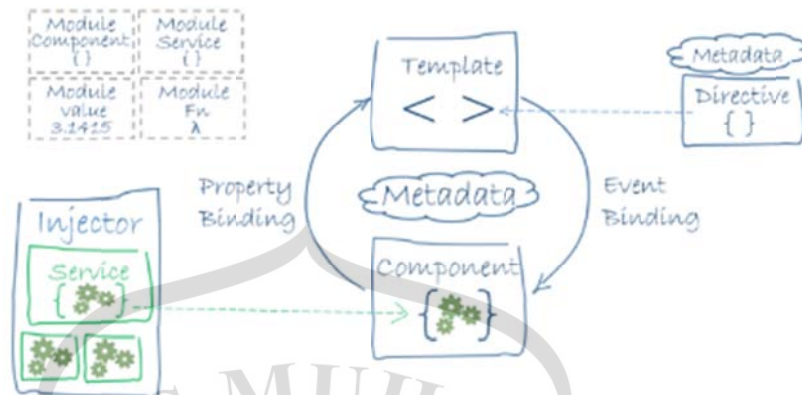
terdapat data. mongoDB menyimpan data dalam bentuk dokumen dengan menggunakan format JSON (Purnomosidi, 2013). Berikut adalah fitur dari mongoDB:

- menggunakan format JSON dalam penyimpanan data
- mendukung indeks
- mendukung replikasi
- auto-sharding untuk skalabilitas horizontal
- query yang lengkap
- pembaruan data yang cepat
- mendukung *Map/Reduce*
- mendukung GridFS

## 7. Angular

Menurut (Maulana, 2017), Angular merupakan salah satu framework yang digunakan untuk membangun modern web application yang digawangi oleh raksasa perusahaan teknologi Google.

Angular adalah versi modern dari AngularJS dimana masih membawa kekuatan AngularJS dan mencoba memenuhi kebutuhan para pengguna yang membutuhkan performa yang jauh lebih baik disbanding pendahulunya. Angular juga menggandeng Typescript dari Microsoft sebagai official language yang digunakan. Typescript ini sendiri merupakan bahasa pre-processor bagi Javascript dimana Typescript menambahkan kemampuan *Type Safe* atau pengenalan tipe data pada syntaxnya.

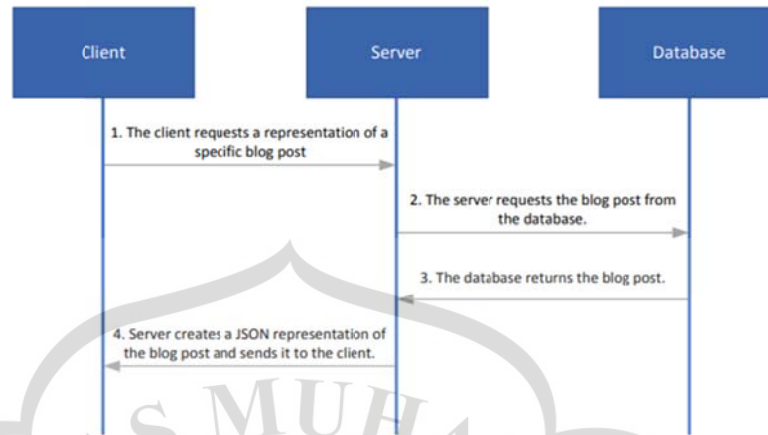


**Gambar 2. 3** Arsitektur Angular (Maulana, 2017)

Selain Typescript, Angular juga menggandeng RxJS dari ReactiveX sebagai *library* pendukungnya. RxJS memberikan arsitektur *Observer pattern* pada AngularJS dimana hal ini sangat dibutuhkan saat kita berurusan dengan *asynchronous/parallel programming*. Sederhananya Observer pada RxJS adalah merupakan penyempurnaan dari Promise yang sangat populer digunakan para Javascript developer saat ini untuk menangani *asynchronous function*.

## 8. REST API (REpresentational State Transfer API)

Menurut (Lange, 2016) dalam bukunya yang berjudul “*THE LITTLE BOOK ON REST SERVICES*”, REST API itu ketika server memiliki *resource* dan klien dapat meminta "*representation*" (yaitu deskripsi) dari "*state*" sumber daya (mis.data). Misalnya, jika sumbernya adalah posting blog yang disimpan di database, maka "*representation state*"-nya bisa menjadi daftar sederhana dari *value* dalam *database*.



**Gambar 2. 4 Cara Kerja REST API (Lange, 2016)**

Ini berarti bahwa *client* tidak peduli dengan *internal server* implementasi sumber daya. *Server* mungkin menyimpan blog posting di database Oracle, file datar, atau bahkan mungkin dihasilkan dengan panggilan prosedur; itu tidak masalah bagi klien. Semua *client* yang dipedulikan adalah representasi yang didapat dari *server*.

JSON (*JavaScript Object Notation*) sering digunakan untuk representasi sumber daya di REST. Ini adalah notasi name-value yang sederhana. Misalnya, representasi posting blog di Langkah 4, di atas, bisa terlihat seperti ini dalam format JSON:

```
{  
    "title": "7 Things You Didn't Know about Star Wars",  
    "content": "George Lucas accidentally revealed that..."  
}
```

Ketika *client* telah menerima representasi/data, itu dapat memperbaruinya (misalnya, ubah judul), dan kirim representasi/data yang diperbaharui kembali ke *server*. *Server* kemudian akan memperbaruinya dengan data dari representasi yang diperbarui, untuk contoh, perbarui catatan di tabel *database* yang mendasarinya dengan judul baru.

Jadi "REpresentational State Transfer" itu sebagai pengantar representasi/data antara client dan server.

#### 9. *Progressive Web App* (PWA)

*Progressive Web App* adalah sebuah aplikasi web yang menggunakan beberapa teknologi modern yang membuat *user experience* seperti menggunakan *Native App* (Android/iOS). (Putu Alfred Crosby, 2017).

Berbeda dengan aplikasi *hybrid*, aplikasi *hybrid* lebih menggabungkan antara aplikasi *native* dan web. PWA ini sepenuhnya web namun pengguna bisa merasakan *experience* seperti menggunakan aplikasi *native*. Seperti bisa dijalankan *offline* atau koneksi buruk sekalipun, bahkan menggunakan animasi yang *smooth* seperti aplikasi *native*. Aplikasinya pun tidak perlu diinstall, namun ikon-nya bisa di

pasang di *home screen* untuk memudahkan membuka aplikasi. PWA sepenuhnya mengandalkan browser. Keunggulan Progressive Web App:

a) *Instant Loading*

Teknologi yang mampu membuat PWA ini bisa digunakan offline bahkan pada koneksi internet yang buruk ialah karena *service worker*. Dengan *service worker* waktu muat web menjadi lebih cepat. Teknologi *service worker* memiliki mekanisme untuk mengontrol asset yang akan di cache serta menyediakan custom permintaan ke jaringan sehingga dapat memberikan pengalaman akses secara offline bagi pengguna. Tetapi untuk dapat menggunakannya pertama kali pengguna harus mengakses website tersebut. Setelah semua asset ter-cache oleh *service worker*.

b) *Add to Homescreen*

Yang membuat PWA ini seperti *native* karena kemampuan untuk menambahkan *icon* ke *homescreen*. Jadi pengguna lebih mudah untuk membuka kembali aplikasi web tanpa harus mengakses URL-nya kembali melalui browser. Untuk memaksimalkan penggunaan fitur ini, pengembang aplikasi wajib menyediakan web app *manifest* yang berisi pengaturan gambar *icon*, nama aplikasi, besar resolusi *icon shortcut*, dan pengaturan tampilan agar ketika dibuka aplikasi web terbuka *full screen* tanpa menampilkan *address bar* lagi.

c) *Fast* (Cepat)

Pengguna web saat ini berharap halaman web yang mereka kunjungi akan interaktif dan *smooth*. Maka *developer* perlu memfokuskan waktu dan usahanya disana. Halaman web tidak hanya cepat, tetapi juga harus berjalan dengan baik, *scrolling* yang cepat dan animasi serta interaksi yang juga harus tetap *smooth*. Mengakses di web di *desktop* berbeda dengan di *mobile*. Pada *mobile* diharapkan halaman web lebih ringan dengan tidak mendownload banyak *asset*.

d) *Secure* (Aman)

Untuk dapat menggunakan *service worker web* haruslah sudah menggunakan protokol HTTPS. HTTPS akan menyandikan atau mengenkripsi data sesi menggunakan protokol SSL (*Secure Socket layer*) atau protokol TLS (*Transport Layer Security*). Jadi data diantara *server web* dan pengguna akan lebih aman. Jika tidak terproteksi penyusup bisa saja mengambil data komunikasi antara *web* kamu dengan browser pengguna. HTTPS juga membantu mencegah penyusup dari gangguan komunikasi antara website dengan browser pengguna. Seperti ISP atau hotel yang meng-inject iklan ke dalam halaman. Penyusup dapat mengeksploitasi komunikasi yang tidak dilindungi untuk mengelabui pengguna agar memberikan informasi sensitif atau menginstal *malware*. Sebagai contoh, beberapa pihak ketiga menyuntikkan iklan ke dalam

website yang berpotensi memecah pengalaman pengguna dan menciptakan kerentanan keamanan. 16 Bahkan website dengan *protocol* https mendapatkan prioritas sinyal peringkat dalam mesin pencari.

e) *Responsive*

Pada era saat ini pengguna mulai bergeser dari desktop ke perangkat mobile. Bahkan banyak juga pengguna yang pertama kali mengakses internet dari perangkat mobile. Agar website bisa menyesuaikan di banyak perangkat seperti desktop, tablet dan smartphone maka penting sebuah website mendukung responsive.

(Anna Erdiawan, 2018)

