

BAB II

TINJAUAN PUSTAKA

2.1 Sistem

Sistem dapat berupa abstrak atau fisik, system yang abstrak adalah susunan yang teratur dari gagasan-gagasan tentang tuhan, manusia dan sebagainya sedangkan System yang bersifat fisik adalah serangkaian unsur yang bekerja sama untuk mencapai tujuan (Jogiyanto, 2014).

Sistem merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan (*input*) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan (*output*) yang diinginkan (Jogiyanto, 2014).

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Leitch & Davis, 2005)

2.2 Metode *Hill Climbing*

Metode *Hill Climbing* adalah salah satu metode yang di gunakan dalam menyelesaikan permasalahan pencarian jarak terdekat. Cara kerjanya adalah menentukan langkah berikutnya dengan menempatkan node yang akan muncul sedekat mungkin dengan sasarannya. Proses Pengujian dilakukan dengan menggunakan fungsi heuristik. Pembangkitan keadaan berikutnya sangat tergantung pada *feedback* dari prosedur pengetesan. Tes yang berupa fungsi heuristik ini akan menunjukkan seberapa baiknya nilai terkaan yang diambil terhadap keadaan keadaan lainnya yang mungkin (Kusumadewi, 2003).

Hill climbing adalah teknik optimasi untuk menemukan pemecahan masalah dari data input dengan menggunakan algoritma. Tahapan terbaik yang di gunakan dalam pemecahan permasalahan adalah properti yang penjelasan state itu sendiri berisi semua informasi yang diperlukan untuk solusi (Russell dan Norvig, 2003). Algoritma berisi memori yang efisien

karena tidak mempertahankan pohon pencaharian namun hanya dapat terlihat pada kondisi saat ini, dan state yg akan datang.

2.3 Location Based Service (LBS)

Location Based Service (LBS) mengacu pada sekumpulan aplikasi yang mengeksplorasi pengetahuan / informasi dari lokasi geografis perangkat *mobile* untuk mendapatkan layanan berdasarkan informasi tersebut” (*Internet and Mobile Association of India, 2008*).

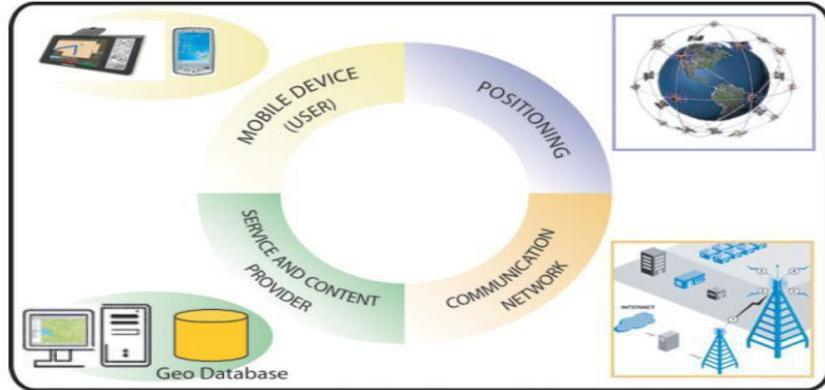
Pemanfaatan LBS memudahkan pengguna perangkat *mobile* mengatur dan memilih layanan sesuai kebutuhan dan dapat dimanfaatkan untuk memberikan berbagai layanan seperti informasi kondisi lingkungan (kemacetan lalu lintas, cuaca, lokasi fasilitas umum terdekat), maupun promosi produk dan jasa.

Pemanfaatan LBS pada sistem operasi perangkat mobile android dimungkinkan dengan adanya dukungan dua unsur utama berikut:

1. *Location Manager (API Maps)* Menyediakan *tool* atau *source* untuk LBS, *Application Program Interface (API)*. *Maps* menyediakan fasilitas untuk menampilkan, memanipulasi maps atau peta beserta *feature-feature* lainnya seperti tampilan satelit, *street* (jalan), maupun gabungannya. Paket ini berada com.google.android.map.
2. *Location Provider (API Location)* Menyediakan teknologi pencarian lokasi yang digunakan oleh *device* atau 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*. *Location Manager* dapat ditentukan lokasi posisi saat ini, *track* gerakan atau perpindahan, serta kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan.

2.3.1 Komponen Location Based Service (LBS)

Dalam Layanan Berbasis Lokasi terdapat lima komponen penting yaitu meliputi:



Gambar 2.1 Komponen LBS

Sumber : *Internet and Mobile Association of India* (2008).

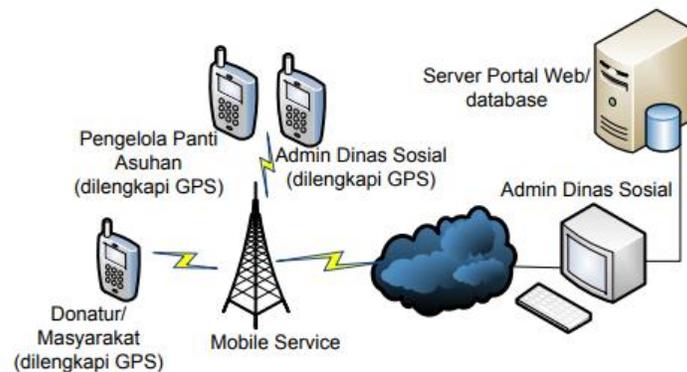
1. *Mobile Devices*: Suatu alat yang digunakan oleh pengguna untuk meminta informasi yang dibutuhkan. Informasi dapat diberikan dalam bentuk suara, gambar, dan teks.
2. *Communication Network*: Komponen kedua adalah jaringan komunikasi yang mengirim data pengguna dan informasi yang diminta dari *mobile* terminal ke *Service Provider* kemudian mengirimkan kembali informasi yang diminta ke pengguna. *Communication Network* dapat berupa jaringan seluler (*GSM*, *CDMA*), *Wireless Local Area Network (WLAN)*, atau *Wireless Wide Area Network (WWAN)*
3. *Positioning Component*: Untuk memproses suatu layanan maka posisi pengguna harus diketahui
4. *Service and Application Provider*: Penyedia layanan menawarkan bermacam-macam layanan kepada pengguna dan bertanggung jawab untuk memproses informasi yang diminta oleh pengguna.
5. *Data and Content Provider*: Penyedia layanan tidak selalu menyimpan semua data yang dibutuhkan yang bisa diakses oleh pengguna. Untuk itu, data dapat diminta dari data dan *content provider*. (Imaniar, 2011)

2.3.2. Cara Kerja Location Based Service (LBS)

Cara kerja LBS pada aplikasi pencarian objek terdekat berdasarkan posisi *user*.

- a. *Smartphone* akan membuka aplikasi yang memanfaatkan layanan LBS yang sudah *ter-install*.

- b. Aplikasi akan melakukan sambungan dengan jaringan *provider* yang dipakai oleh *user*.
- c. Selanjutnya informasi posisi *user* pada perangkat *mobile* yang diperoleh dari *Location Sensor*. Hal ini dapat dilakukan oleh perangkat dengan menggunakan GPS sendiri atau layanan posisi jaringan yang berasal dari *provider*.
- d. Perangkat *mobile* pengguna akan mengirimkan permintaan informasi ke satelit untuk menentukan *longitude* dan *latitude* dari pengguna aplikasi tersebut.
- e. *Provider* menghubungkan aplikasi pada *smartphone* dengan *server* LBS dan meminta data yang diinginkan *user* beserta informasi tentang jarak dan cara yang diperlukan dalam menjangkau lokasi tujuan.
- f. Terakhir, *user* mendapatkan data dan ditampilkan di *smartphone* melalui aplikasi.



Gambar 2.2 Cara Kerja *Location Based Service*.

Sumber : Justino , Syahputri, & Nurfiana (2020).

2.4 Android

Pemakaian android di Indonesia semakin pesat, terlihat dari jumlah penggunaan android yang terus meningkat. Menurut data Badan Pusat Statistik (BPS) pemakaian android di Indonesia kini telah mencapai angka 171 juta pada tahun 2019. Jika disesuaikan dengan jumlah populasi penduduk Indonesia mencapai 264 juta jiwa, maka pengguna internet di Indonesia mengalami peningkatan sebanyak 54,86% dari tahun 2017 yang lalu.

Android adalah sistem operasi yang berfungsi sebagai penghubung (*device*) antara pengguna dan perangkat keras pada *smarthphone* atau alat elektronik tertentu. Sehingga, hal tersebut memungkinkan pengguna dapat berinteraksi dengan *device* dan menjalankan berbagai macam aplikasi *mobile* (Firly, 2018).

Berikut ini adalah kelebihan dalam menggunakan Android, yaitu :

1. *User interface* yang mudah digunakan
2. Memiliki sifat *open source*
3. Memiliki tampilan yang mudah diubah.

2.5 Perangkat Lunak yang Digunakan

Untuk membangun perangkat lunak sistem pencari tempat kost, dibutuhkan beberapa perangkat lunak, yaitu :

2.5.1 Android SDK

Android SDK adalah merupakan sebuah kit yang berfungsi untuk mengembangkan berbagai aplikasi berbasis Android oleh para developer. Didalam SDK telah terdapat berbagai tools yang bertujuan untuk proses pengembangan aplikasi seperti proses debugger, emulator, *software libraries*, dan dokumentasi (Firly, 2018)

2.5.2 Android Studio

Android Studio adalah sebuah untuk *Android Development* yang diperkenalkan *google* pada acara *Google I/O 2013*. Android Studio merupakan pengembangan dari *Eclipse IDE*, dan dibuat berdasarkan *IDE Java* populer, yaitu *IntelliJ IDEA*. Android Studio merupakan *IDE* resmi untuk pengembangan aplikasi Android (Firly, 2018).

2.5.3 Google Maps

Google Maps merupakan layanan dari *google* yang mempermudah penggunaannya untuk melakukan kemampuan pemetaan untuk aplikasi yang dibuat. Sedangkan *Google Maps API* memungkinkan pengembangan untuk mengintegrasikan *Google Maps* ke dalam

situs web. Dengan menggunakan *Google Maps API* memungkinkan untuk menanamkan situs *Google Maps* ke dalam situs eksternal, di mana situs data tertentu dapat dilakukan *overlay*. Meskipun pada awalnya hanya *JavaScript API*, *API Maps* sejak diperluas untuk menyertakan sebuah *API* untuk *Adobe Flash* aplikasi, layanan untuk mengambil gambar peta statis, dan layanan web untuk melakukan *geocoding*, menghasilkan petunjuk arah mengemudi, dan mendapatkan profil elevasi. Kelas kunci dalam perpustakaan *Maps* adalah *MapView*, sebuah *subclass* dari *ViewGroup* dalam standar perpustakaan Android. Sebuah *MapView* menampilkan peta dengan data yang diperoleh dari layanan *Google Map* (Firly, 2018).

2.5.4 MySQL

MySQL merupakan *software* yang tergolong sebagai *DBMS* (*Database Management System*) yang bersifat *open source*. *Open source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat *MySQL*), selain itu tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan *men-download* (mengunduh) di internet secara gratis (A.S Rosa & Shalahuddin, 2018).

2.5.5 XAMPP

XAMPP adalah aplikasi yang berfungsi sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri beberapa program antara lain: Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP sendiri merupakan singkatan dari X empat sistem operasi, yang meliputi Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU, merupakan web server yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis (Sadeli, 2014).

2.6 UML (*Unified Modelling Language*)

UML merupakan sistem arsitektur yang bekerja dalam OOAD (*Object-Oriented Analysis/Design*) dengan satu bahasa yang konsisten untuk menentukan, visualisasi, mengkonstruksi, dan mendokumentasikan artifact (sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa software, dapat berupa model, deskripsi, atau software) yang terdapat dalam sistem software. UML merupakan bahasa pemodelan yang

paling sukses dari tiga metode OO yang telah ada sebelumnya, yaitu Booch, OMT (*Object Modeling Technique*), dan OOSE (*Object-Oriented Software Engineering*). UML merupakan kesatuan dari ketiga pemodelan tersebut dan ditambah kemampuan lebih karena mengandung metode tambahan untuk mengatasi masalah pemodelan yang tidak dapat ditangani ketiga metode tersebut. UML dikeluarkan oleh OMG (*Object Management Group, Inc*) yaitu organisasi internasional yang dibentuk pada 1989, terdiri dari perusahaan sistem informasi, *software, developer,* dan para user sistem komputer.

Dengan adanya UML, diharapkan dapat mengurangi kekacauan dalam bahasa pemodelan yang selama ini terjadi dalam lingkungan industri. UML diharapkan juga dapat menjawab masalah penotasian dan mekanisme tukar menukar model yang terjadi selama ini.

Tujuan UML diantaranya adalah :

- a) Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
- b) Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
- c) Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.

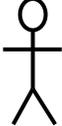
Bahasa Pemodelan Pengembangan Sistem (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa and Shalahudin, 2018). Beberapa jenis diagram *UML* antara lain sebagai berikut:

2.6.1 Use Case Diagram

Use case diagram atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Rosa and Shalahudin, 2018), simbol-simbol yang ada pada diagram *use case* dapat dilihat pada gambar 2.1 di bawah ini:

Tabel 2.1 Simbol Diagram Use Case
Sumber : (Rosa and Shalahudin, 2018)

Simbol	Deskripsi
--------	-----------

Use Case 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
Aktor/actor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor

Tabel 2.1 Simbol Diagram Use Case (Lanjutan)

Sumber : (Rosa and Shalahudin, 2018)

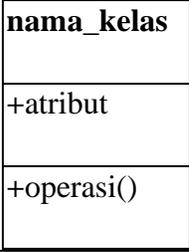
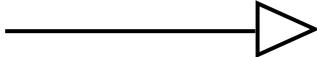
Simbol	Deskripsi
Ekstensi/extend <i><<extend>></i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan.
Asosiasi/association 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan <i>actor</i>
Generalisasi/generalization 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
Menggunakan/Include/uses <i><<include>></i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya

2.6.2 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa and Shalahudin, 2018), simbol-simbol yang ada pada diagram kelas pada tabel *class diagram* 2.2 di bawah ini:

Tabel 2.2 Simbol Class Diagram

Sumber : (Rosa and Shalahudin, 2018)

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka/<i>Interface</i></p>  <p>nama_interface</p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
<p>Asosiasi/<i>asociation</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Asosiasi berarah/<i>directed association</i></p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
<p>Kebergantungan/<i>dependecy</i></p> 	Relasi antar kelas dengan makna kebergantungan antar kelas
<p>Agregasi/<i>agregation</i></p> 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)

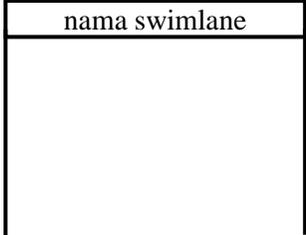
2.6.3 Activity Diagram

Activity diagram atau Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi

aktivitas yang dapat dilakukan oleh sistem (Rosa and Shalahudin, 2018). simbol-simbol yang ada pada *activity diagram* dapat dilihat pada tabel 2.4 di bawah ini :

Tabel 2.3 Simbol *Activity Diagram*

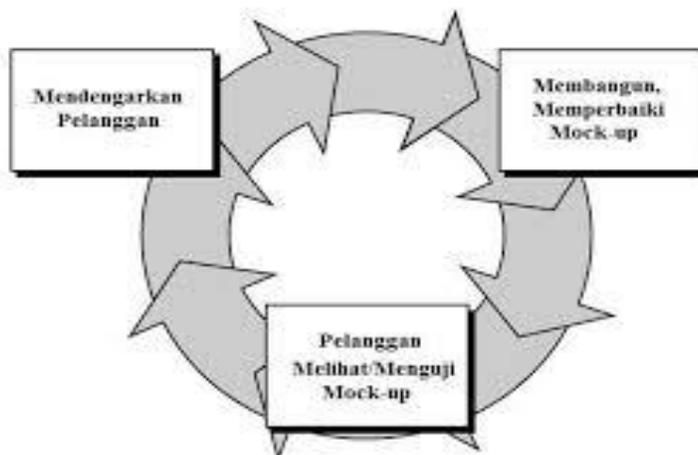
Sumber : (Rosa and Shalahudin, 2018)

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Dokumen 	Menunjukkan dokumen sumber atau laporan
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.7 Metode Pengembangan Perangkat Lunak

Model prototipe dapat digunakan untuk menyambung ketidakpahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (A.S Rosa, & Shalahuddin, M., 2018)

Gambar 2.3 Ilustrasi model *prototipe*.



Sumber : (Rosa and Shalahudin, 2018)

Menurut Rosa and Shalahudin (2018) terdapat tahapan dalam proses *prototipe* yaitu:

1. Mendengarkan Pelanggan

Pada tahap ini, dilakukan pengumpulan kebutuhan dari sistem dengan cara mendengar kebutuhan pelanggan sebagai pengguna sistem perangkat lunak untuk menganalisis serta mengembangkan kebutuhan pengguna.

2. Merancang dan Membuat *Prototipe*

Pada tahap ini, dilakukan perancangan dan pembuatan prototipe sistem yang disesuaikan dengan kebutuhan pengguna.

3. Uji Coba

Pada tahap ini, dilakukan pengujian *prototipe* sistem oleh pengguna kemudian dilakukan

evaluasi sesuai dengan kekurangan-kekurangan dari kebutuhan pelanggan. Jika sistem sudah sesuai dengan prototipe, maka sistem akan diselesaikan sepenuhnya. Namun, jika masih belum sesuai kembali ke tahap pertama.

2.8 Pengujian Sistem *Black – Box*

Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perancang perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program (Dodenti, 2013). Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

- a. Fungsi – fungsi yang tidak benar atau hilang,
- b. Kesalahan interface
- c. Kesalahan dalam struktur data atau akses eksternal
- d. Kesalahan kinerja

2.9 Penelitian Terkait

Berikut ini adalah penelitian yang terkait dengan peneliti, yaitu :

1. Sundari, & Arumaryawan (2018), meneliti tentang Sistem Informasi Geografis dengan Google Map Untuk Pencarian Rumah Kost. Metode *research and development* menentukan permasalahan mencari tempat kos yang sesuai dengan kebutuhan tidaklah mudah, masalah yang dihadapi terkadang susah mencari lokasi yang dekat dengan tempat kerja ataupun tempat kuliah dan harga yang terjangkau. Dan mengembangkan aplikasi baru dengan pemanfaatan teknologi melalui Sistem Informasi Geografis (SIG).
Pengelolaan rumah kos yang ada di sekitar di Jakarta dalam bentuk informasi spasial. Penggunaan SIG yang disajikan dalam bentuk web, memberikan kemudahan pemilik kos dalam menginformasikan lokasinya, dan memudahkan masyarakat dalam pencarian lokasi rumah kos.
2. Djusar, Fajrizal, Rhoma, & Rofiantoro (2017), meneliti tentang Implementasi Kos Seeker Di Wilayah Pekanbaru Dengan Metode Content Based Recommender System Berbasis Web. Dengan aplikasi Kos Seeker pencari kos lebih mudah mencari informasi kosan dan memilih kosan sesuai kriteria seperti fasilitas kamar,

fasilitas umum, harga dan lokasi kampus yang direkomendasikan dari Aplikasi Kos Seeker. Aplikasi ini juga dapat memudahkan juga bagi pemilik kos dalam memasarkan tempat kos-kosannya, tanpa harus menyebarkan brosur yang menggunakan biaya. Hasil uji konsultasi terhadap website pencari kos ini menunjukkan bahwa website nantinya mampu memudahkan dalam mencari rumah kos yang sesuai dengan kriteria yang di pilih baik segi fasilitas, kampus terdekat dan harga yang sesuai.

3. Usman, & Masdi (2018), meneliti tentang Aplikasi Pencarian Lokasi Kos Di Kota Tembilahan Berbasis Web Mobile. Masalah yang mereka hadapi adalah sulitnya mendapatkan informasi mengenai lokasi kos-kosan. Mereka terkendala dengan terbatasnya pengetahuan tentang seluk beluk kota dan nama-nama jalan. Aplikasi ini akan manajemen data-data kos dan memberikan informasi kos secara detail dan lokasi yang akurat dengan memanfaatkan peta Google. Selain itu aplikasi ini juga memberikan keuntungan bagi pengelola kos untuk memasarkan kos mereka secara online kepada masyarakat luas.
4. Pratikto, Suraya, & Sutanta (2014), meneliti tentang Sistem Pencarian Dan Pemesanan Rumah Kos Menggunakan Sistem Informasi Geografi (SIG). Para pendatang yang ingin menetap sementara di kota Yogyakarta biasanya menghadapi kendala seperti tidak mengetahui wilayah, atau tidak memiliki teman yang dapat menunjukkan rumah kost yang sesuai. Penelitian ini mengembangkan sebuah aplikasi sistem informasi geografis (SIG) berbasis web untuk membantu para pendatang di kota Yogyakarta dalam proses pencarian dan pemesanan rumah kos yang sesuai kebutuhannya, dan membantu pemilik rumah kos dalam memasarkan rumah kosnya secara online. Aplikasi dikembangkan dengan memanfaatkan framework CodeIgniter dan Google maps.
5. Sagita & Simpony (2018), meneliti tentang Web Sistem Informasi Pencarian Info Kostan Menggunakan Google Maps API 3. Untuk mendapatkan informasi kostan biasanya dilakukan dengan datang ke tempat tujuan secara langsung untuk mencari tempat kostan. Dengan semakin berkembangnya fungsi dari sebuah website, website bisa diintegrasikan dengan Google Maps API untuk mendapatkan lokasi info kostan yang sesuai sehingga dapat memberikan kemudahan terhadap

konsumen yang akan mencari kostan. Selain bisa mengakses website, pemilik kost juga dapat mempromosikan tempat kost dengan mudah dan efektif tanpa harus mencetak brosur atau flyer serta dapat mempermudah penyebaran informasi. Metode yang digunakan dalam pembuatan rancangan sistem ini adalah metode waterfall. Metode waterfall adalah model SDLC yang paling sederhana, model ini hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah. Hasil dari pembuatan website ini adalah untuk memberikan kemudahan terhadap konsumen dan pemilik kost dalam setiap aktivitas.

6. Setianni, & Syahputri (2019), meneliti tentang Sistem Informasi Pemesanan Salon *Online* Berbasis *Location Based Service*. Pada penelitian ini dibangun sistem pemesanan salon online (PESONA) berbasis android untuk memesan layanan salon kerumah atau tempat lain yang diinginkan (*off-site service*) dan perkiraan lamanya waktu yang dibutuhkan untuk perawatan dan biaya jasa yang akan ditagihkan. Berdasarkan uji coba menggunakan *black box*, aplikasi ini dapat menghitung waktu perawatan dan jasa dengan cepat. Sedangkan penentuan lokasi yang menggunakan *Location Based Service* telah menampilkan hasil yang akurat.