**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

1. **IDENTITAS PENGUSUL**

**NAMA : Ivan Ahmed Giovanni**

**NRP : 5109 100 042**

**DOSEN WALI : Waskitho Wibisono, S.Kom., M.Eng., Ph.D.**

1. **JUDUL TUGAS AKHIR**

**Pengingat Agenda Berdasarkan Konteks Lokasi Menggunakan GPS dengan Algoritma k-Nearest Neighbor pada Perangkat Bergerak**

***Task Reminder Based on Location Context Using GPS with k-Nearest Neighbor Algorithm on Mobile Device***

1. **ABSTRAKSI**

Pada masa sekarang ini aplikasi pengingat tugas pribadi menjadi kebutuhan penting bagi banyak orang, terutama bagi pengguna yang mempunyai banyak kesibukan. Penting untuk mengingatkan tugas apa yang seharusnya dilakukan ketika berada di suatu lokasi tertentu karena banyaknya tugas yang harus dilakukan. Pengingat tugas menggunakan kalender di telepon genggam merupakan teknologi yang populer pada saat ini, pengingat tersebut menggunakan waktu sebagai pemicunya. Hal ini tentunya akan lebih baik jika pengingat tersebut menggunakan pemicu berdasarkan lokasi orang tersebut berada dan juga bisa mengingatkan apa saja yang harus dibawa atau dilakukan untuk menyelesaikan tugas tersebut.

Untuk menyelesaikan permasalahan tersebut maka digunakanlah gabungan teknologi *Location-based Service* atau *LBS* dengan *GPS* untuk melakukan reminder terhadap tugas. Penggunaan *LBS* akan lebih menguntungkan lagi jika bisa memberikan sugesti kepada si pengguna apabila ada yang perlu dilakukan dan diperhatikan ketika melakukan suatu tugas. Untuk mengatasi hal tersebut maka diperlukan yang namanya teknologi c*ontext-aware* yang digunakan untuk memberikan sugesti tentang apa yang diperlukan untuk memenuhi suatu tugas dengan memunculkan lokasi sekitar untuk mendapatkan kebutuhan tugas tersebut dan data-data apa saja yang digunakan untuk memenuhi tugas tersebut. Untuk menerapkan *context-aware* tersebut maka diperlukan algoritma *k-NN* *atau k-Nearest Neighbor* untuk melakukan sugesti terhadap tugas-tugas yang diperlukan.

Tujuan dari pembuatan aplikasi ini adalah membuat suatu aplikasi *mobile* dengan *platform* Android yang dapat mengingatkan pengguna tentang tugas yang akan dilakukan berdasarkan pemicu lokasi. Ditambah pula dengan konsep *context-aware* dengan algoritma *k-NN* atau *k-Nearest Neighbor* untuk memberikan sugesti lokasi terdekat untuk memenuhi kebutuhan yang harus dibawa dan memberikan sugesti tentang data-data atau *file-file* pekerjaan yang berhubungan dengan konteks lokasi tersebut.

1. **PENDAHULUAN**
   1. **LATAR BELAKANG**

Dewasa ini manajemen pengingat tugas pribadi merupakan suatu hal yang penting pada masa sekarang ini. Hal ini penting untuk mengingatkan tugas apa yang seharusnya dilakukan ketika berada di suatu lokasi tertentu. Pengingat tugas menggunakan kalender di telepon genggam merupakan teknologi yang populer pada saat ini, namun kebanyakan pengingat ini menggunakan waktu sebagai pemicunya. Hal ini tentunya akan lebih baik jika pengingat tersebut menggunakan pemicu berdasarkan lokasi orang tersebut berada dan juga bisa mengingatkan apa saja yang harus dibawa untuk menyelesaikan tugas tersebut.

*Platform* Android sebenarnya bisa menyelesaikan permasalahan tersebut. Dengan teknologi sensor *Global Positioning System(GPS)*[1] yang terdapat di platform Android maka permasalahan diatas bisa diatasi, tentu saja dengan gabungan dari teknologi lain.

Ada beberapa teknologi untuk melengkapi teknologi tersebut. Salah satunya dengan *Location Based Service(LBS)*[1] yang sudah umum digunakan pada *smart phone*. Sistem kerja dari *LBS* ini adalah menggunakan sensor *GPS* untuk mengetahui lokasi dari si pengguna dan memberikan peringatan pada si pengguna tentang lokasi tersebut dan apa yang harus dilakukan di tempat tersebut.

Penggunaan *LBS* akan lebih menguntungkan lagi jika bisa memberikan sugesti kepada si pengguna apabila ada yang perlu dibawa ketika melakukan suatu tugas. Untuk mengatasi hal tersebut maka diperlukan yang namanya teknologi *context-aware*[2] yang digunakan untuk memberikan sugesti tentang apa yang diperlukan untuk memenuhi suatu tugas dengan ditambah lokasi sekitar untuk mendapatkan kebutuhan tugas tersebut dan data-data apa saja yang digunakan untuk memenuhi tugas tersebut. Untuk menerapkan c*ontext-aware* tersebut maka diperlukan algoritma *k-NN* *atau k-Nearest Neighbor* [3] untuk melakukan sugesti terhadap tugas-tugas yang diperlukan.

* 1. **TUJUAN DAN MANFAAT**

Tujuan dari tugas akhir ini adalah membuat suatu aplikasi *mobile* dengan *platform* Android yang dapat mengingatkan pengguna tentang tugas yang akan dijalankan berdasarkan lokasi. Ditambah pula dengan konsep *context-aware* dengan algoritma *k-NN* untuk memberikan sugesti lokasi terdekat kebutuhan yang harus dibawa untuk memenuhi suatu tugas dan data-data yang berhubungan dengan konteks lokasi tersebut.

* 1. **RUMUSAN MASALAH**

Adapun yang menjadi permasalahan utama dalam tugas akhir ini.

1. Bagaimana cara menerapkan pengingat tugas berdasarkan koodinat lokasi pada telepon genggam Android?
2. Bagaimana cara menerapkan metode *context-aware* dengan algoritma *k-NN* untuk memberikan sugesti lokasi terdekat dan keperluan yang harus dibawa untuk memenuhi suatu tugas?
   1. **BATASAN MASALAH**

Dari permasalahan yang telah diuraikan di atas, terdapat beberapa batasan masalah terhadap tugas akhir ini, yaitu:

1. Menggunakan Android kernel versi 2.3.3
2. Algoritma yang digunakan untuk metode *context-aware* adalah *k-Nearest Neighbor*.
3. Menggunakan GPS yang sudah menjadi satu paket dengan Android.
4. Radius ketika pengguna mendekati lokasi yang dituju sebesar 10 meter.
5. Menggunakan peta yang didapat dari Google Maps.
6. **TINJAUAN PUSTAKA**
   1. **GPS**

GPS (Global Positioning System) merupakan sistem navigasi satelit yang dikembangkan oleh Departemen Pertahanan Amerika Serikat (US DoD = United States Department of Defense). GPS memungkinkan kita mengetahui posisi geografis kita (lintang, bujur, dan ketinggian di atas permukaan laut). Jadi dimanapun kita berada di muka bumi ini, kita dapat mengetahui posisi kita dengan tepat.

Perangkat GPS menerima sinyal yang ditransmisikan oleh satelit GPS. Dalam menentukan posisi, kita membutuhkan paling sedikit 3 satelit untuk penentuan posisi 2 dimensi (lintang dan bujur) dan 4 satelit untuk penentuan posisi 3 dimensi (lintang, bujur, dan ketinggian). Semakin banyak satelit yang diperoleh maka akurasi posisi kita akan semakin tinggi. Untuk mendapatkan sinyal tersebut, perangkat GPS harus berada di ruang terbuka. Apabila perangkat GPS kita berada dalam ruangan atau kanopi yang lebat dan daerah kita dikelilingi oleh gedung tinggi maka sinyal yang diperoleh akan semakin berkurang sehingga akan sukar untuk menentukan posisi dengan tepat atau bahkan tidak dapat menentukan posisi[6].

* 1. **Google Maps API**

Google Maps *API*(*Aplication Programming Service*) merupakan *API* yang di gunakan jika ingin menambahkan fitur Google Maps di aplikasi yang sedang dikembangkan. Google Maps merupakan salah satu layanan gratis yang dimiliki Google untuk peta digital yang menawarkan tampilan peta dengan tampilan gambar satelit ataupun mode gambar jalan raya. *API* yang telah disediakan oleh Google Maps ini juga memungkinkan untuk melakukan *overlay* dengan data tertentu pada peta termasuk *overlay* untuk menggambarkan jalan, posisi, dan beberapa objek lain.

* 1. **Web Service**

*Web service* adalah segala pelayanan yang tersedia di internet, menggunakan format data standar XML dan tidak bergantung pada sistem operasi atau bahasa pemrograman [7]. Penggunaan dari *web service* adalah sebagai penyedia layanan kepada sistem lain, sehingga bisa saling berinteraksi dengan sistem penyedia *web service.* Informasi yang tersedia memiliki format XML, sehingga data dapat diakses oleh system yang lain meski terdapat perbedaan platform, sistem operasi maupun bahasa pemrograman. *web service* berjalan di port 80 yang merupakan port standar HTTP sehingga tidak diperlukan pengaturan khusus di sisi *firewall.*

* 1. **Location-based Service**

Merupakan suatu jenis penggunaan teknologi yang pada umumnya menggunakan gabungan sinyal *GPS* dan peta dari Google Maps maupun peta yang dibuat sendiri. Jenis teknologi ini menyediakan informasi yang berhubungan dengan lokasi yang diminta secara *real time*[1]. Sinyal *GPS* diterima dan diolah berdasarkan kebutuhan pengguna seperti *nearest location* atau mencari rute.

* 1. **Context-Aware**

Merupakan jenis teknologi yang dapat mendeteksi konteks situasi dan atau kegiatan pengguna dan dapat berinteraksi secara cerdas dan bekerja dengan tepat untuk memberikan reaksi atau respon yang tepat terhadap pengguna[2].

* 1. **k-Nearest Neighbor**

k-Nearest Neighbor merupakan jenis algoritma yang menggunakan metode non-parametrik[3]. Pendekatan terhadap input *neighbor* yaitu membandingkan *training* *data* dengan data untuk *testing*. Algoritma ini menghasilkan klasifikasi data yang paling dekat dengan data *testing* sesuai dengan jumlah k yang telah ditentukan.

1. **METODOLOGI**

Dalam tugas akhir ini dibuat sebuah aplikasi berbasis *mobile* yang dijalankan pada platform mobile Android yang digunakan untuk mengingatkan pengguna akan tugas yang akan dilakukan berdasarkan lokasi beserta tugas apa saja yang harus dilakukan. Jadi pada dasarnya aplikasi ini adalah sebuah aplikasi manajemen tugas pengguna berdasarkan lokasi dan ditambahkan metode *context-aware*.

Terdapat dua metode atau cara yang akan digunakan dalam aplikasi ini. Yang pertama yaitu untuk melakukan *reminder* terhadap tugas yang dilakukan dengan parameter koordinat lokasi yang akan dituju dengan cara pengguna memasukkan data koordinat tempat yang akan dituju kedalam *database*. Ketika pengguna berada dekat di radius lokasi yang ada di *database* maka aplikasi akan melakukan reminder terhadap tugas yang akan dilakukan.

Metode yang kedua berhubungan dengan cara untuk melakukan context-aware yaitu dengan algoritma *k-Nearest Neighbor*[3]atau *k-NN*. Algoritma ini akan melakukan klasifikasi terhadap koordinat lokasi *point of interest* dalam *database* yang konteksnya berhubungan dengan konteks pada tugas yang akan dilakukan oleh pengguna. Data *point of interest* merupakan data lokasi yang telah dimasukkan oleh *administrator* dari *server*.

Berikut adalah cara menentukan sugesti lokasi yang berhubungan dengan konteks tugas yang akan dilakukan. Apabila ditentukan jumlah data yang diklasifikasi sebanyak *k* telah ditentukan maka akan dilakukan perhitungan jarak suatu *suggestion location* terhadap lokasi yang ditentukan sebagai tempat tugas. Perhitungan tersebut menggunakan rumus *Euclidean*[4] untuk menghitung jarak dari titik lokasi pengguna berada ke titik lokasi POI dengan parameter hitung koordinat titik lokasi POI yang terdapat pada data *training*. Rumus Euclidean ditunjukkan pada Persamaan 1[3] dibawah ini.

(1)

Pada rumus diatas merupakan rumus asli dari persamaan Euclidean untuk menghitung jarak antara titik A dan titik B. Dimana adalah jarak skalar dari dua buah vektor a dan b dari matriks berukuran *d* dimensi. Simbol *n* berarti data koordinat dari suatu lokasi yang sejumlah banyak *n*.

Dikarenakan hanya dipakai untuk 2 koordinat saja, yaitu x dan y maka rumus diubah menjadi Persamaan 2[3]. Persamaan hanya menampilkan 2 koordinat tempat untuk perhitungan.

(2)

Koordinat data *training* didapat dari input pengguna yang disimpan dalam *database*. Dalam *database* data *training* selain koodinat dan nama tempat, disimpan juga jenis konteks dari tempat tersebut agar bisa dihubungkan dengan lokasi yang dituju. Pada Algoritma 1[5] dijelaskan algoritma untuk melakukan klasifikasi koordinat dengan jarak terdekat yang didapat pada peta. Algoritma berikut ini hanya digunakan untuk klasifikasi jarak saja, sedangkan algoritma untuk aplikasi terdapat tambahan parameter berupa konteks yang berhubungan dengan pekerjaan yang dilakukan.

1. Let k be number of nearest neighbors and D be the set of training samples(yj).

2. for each test sample xi do compute d( xi , yj ) using Euclidean distance for every sample yj of D

3. Select the k closeset training samples yj (neighbours) to test sample xi

4. Classify the sample xi based on majority class among its nearest neighbors.

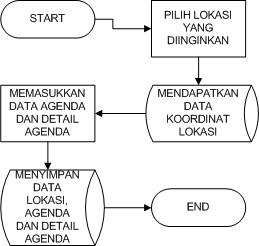
5. end for

**Algoritma 1. Klasifikasi *k-Nearest Neighbor***

Setelah dilakukan penghitungan maka data training tersebut diambil sejumlah *k* dan diurutkan berdasarkan jarak terdekat dari lokasi pengguna serta telah dipilah berdasarkan konteks yang cocok. Hal tersebut ditambahkan dalam algoritma untuk memudahkan klasifikasi data. Data *training* sejumlah k tersebut dibandingkan dengan data *testing* mana yang paling dekat dan cocok untuk pengguna. Sehingga algoritma *k-Nearest Neighbor* yang digunakan memakai 2 parameter yaitu lokasi koordinat dan konteks yang berhubungan dengan pekerjaan yang dilakukan.

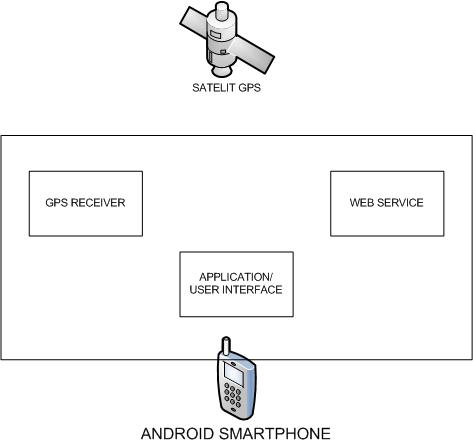
Setelah ditemukan rekomendasi tempat-tempat yang cocok untuk memenuhi tugas yang akan dijalankan selanjutnya akan diberikan rekomendasi terhadap *file-file* atau data-data yang berkenaan dengan tugas tersebut. Hal tersebut bisa dilakukan dengan melakukan *query* terhadap data-data tentang *file* yang telah sebelumnya dimasukkan oleh pengguna. Aplikasi mencari persamaan konteks yang ada di database dengan konteks tugas yang menjadi parameter. Jika terdapat yang cocok maka akan dimunculkan data yang cocok untuk pengguna.

Gambar 2[1] berikut ini menjelaskan alur untuk mempersiapkan atau membuat data pengingat agenda berdasarkan lokasi yang nantinya akan menjadi data *location-based reminder*.



Gambar . Diagram Alir Membuat Data Pengingat

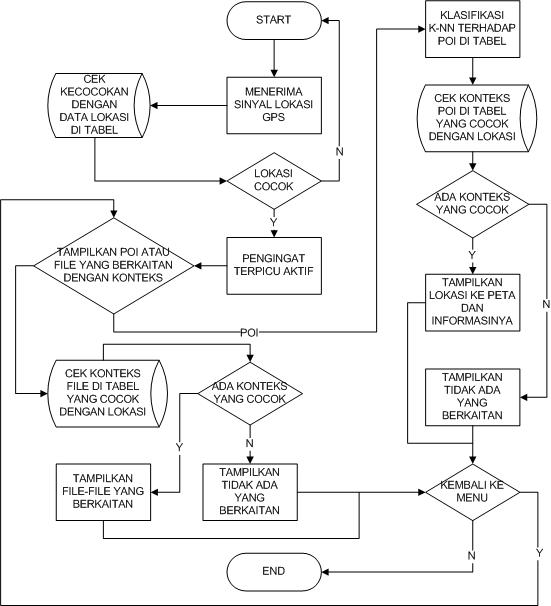
Alur kerja dimulai ketika pengguna memilih lokasi yang diinginkan dan sudah menentuka koordinat lokasi. Ketika sudah ditentukan, pengguna menambahkan agenda dan detail dari agenda seperti konteks dari pekerjaan dan *file-file* yang berhubungan dengan konteks pekerjaan baru setelah itu disimpan ke *database*. Berikut pada Gambar 3[1] ditampilkan arsitektur dari sistem pengingat berdasarkan lokasi ini. Menggunakan *smart phone* Android sebagai aplikasi di sisi klien dan menerima data dari satelit GPS dan *database* dari server.



Gambar . Arsitektur Aplikasi

Pada Gambar 3[1] diatas dijelaskan tentang arsitektur dari sistem pengingat tugas berdasarkan lokasi. Lebih jelas diutarakan di Gambar 4[1] GPS *signal receiver* menerima sinyal balik berupa data koordinat yang diminta oleh aplikasi. Aplikasi lalu melakukan cek terhadap kecocokan antara koordinat dari GPS dan data yang ada di *database*. Hal tersebut dilakukan berkelanjutan sampai akhirnya menemukan kecocokan antara koordinat yang diberikan dan data yang ada di *database*.

Ketika pengguna berada di lokasi yang cocok dengan lokasi yang berada di *database*, maka pengingat akan terpicu dan memberikan peringatan. Setelah itu pengguna dapat memilih untuk menampilkan lokasi terdekat sejumlah *k* yang merupakan lokasi-lokasi yang cocok konteksnya dengan konteks pekerjaan yang akan dilakukan oleh pengguna di suatu lokasi. Selain itu pengguna juga dapat menampilkan *file-file* atau data-data yang penting dan dibutuhkan untuk menyelesaikan tugas si pengguna. *File-file* yang ditampilkan adalah *file-file* yang cocok konteksnya dengan pekerjaan atau tugas si pengguna di suatu lokasi.



Gambar . Diagram Alir Pengingat Tugas

Data-data lokasi point of interest atau lokasi-lokasi penting yang sering dikunjungi untuk menyelesaikan suatu pekerjaan dimasukkan di *server* dan diakses oleh pengguna. Untuk *testing*, area *point of interest* dibatasi daerah sekitar kampus ITS.

Pada fungsi untuk menampilkan *file-file* atau data-data yang berhubungan untuk menyelesaikan suatu tugas digunakan *query* pada *database*. *Query* dilakukan dengan menyamakan konteks yang ada pada database dengan konteks pada pekerjaan yang dilakukan.

1. **JADWAL KEGIATAN**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | Bulan | | | | | | | | | | | |
| April | | | | Mei | | | | Juni | | | |
| 1 | Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Implementasi dan Pembuatan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Uji Coba dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |

1. **DAFTAR PUSTAKA**

[1] Chi-Yi Lin, Ming-Tze Hung and Wei-Hsun Huang, A Location-based Personal Task Management Application for Indoor and Outdoor Environments, in: 15th International Conference on Network-Based Information Systems (2012).

[2] Karol Waga, Andrei Tabarcea, Pasi Fränti, Context Aware Recommendation of Location-based Data, in: 15th International Conference on System Theory, Control, and Computing (ICSTCC) (2011).

[3] Prof. Thomas B. Fomby, K-Nearest Neighbors Algorithm: Prediction and Classification, Department of Economics Southern Methodist University Dallas (2008).

[4] Anonim, “Euclidean Distance”, 27 Februari 2013. [Online]. Tersedia: <http://en.wikipedia.org/wiki/Euclidean_distance>. [Diakses pada tanggal 21 Februari 2013]

[5] Asha Gowda Karegowda, M.A. Jayaram, A.S. Manjunath, Cascading K-means Clustering and K-Nearest Neighbor Classifier for Categorization of Diabetic Patients, International Journal of Engineering and Advanced Technology (IJEAT) (2012).

[6] Anonim, “GPS”. [Online]. Tersedia: http://www.navigasi.net/gofaq.php. [Diakses pada tanggal 21 Februari 2013]

[7] Ethan Cerami, Web Services Essensials, Amerika Serikat: O’Reilly (2002) 6-7.