

Aplikasi Rekomendasi Tempat Makan Menggunakan Algoritma Slope One pada Platform Android

Dharma Pratama¹, Seng Hansun*²

^{1,2}Program Studi Teknik Informatika, Universitas Multimedia Nusantara, Tangerang, Indonesia
e-mail: *¹dharmapratama24@gmail.com, ²hansun@umn.ac.id

Abstrak

Makanan adalah salah satu kebutuhan pokok dari manusia. Kebutuhan makanan akan selalu meningkat seiring jumlah penduduk yang selalu meningkat sehingga semakin banyak tempat makan yang bermunculan. Banyaknya tempat makan yang ada menyebabkan kebingungan saat memilih tempat makan. Oleh karena itu, dibutuhkan sebuah aplikasi untuk memberikan rekomendasi tempat makan. Pada aplikasi ini, rekomendasi yang diberikan didapatkan dari perhitungan menggunakan algoritma Slope One dan daftar tempat makan yang didapatkan berasal dari Google Places API. Algoritma Slope One membuat rekomendasi dengan menjumlahkan rating dari suatu tempat dengan rata-rata selisih suatu tempat dengan tempat lainnya. Aplikasi telah diuji coba pada pengguna menggunakan kuisioner yang dibuat berdasarkan kuisioner J.R. Lewis dengan kategori pertanyaan tingkat kegunaan aplikasi, kualitas informasi yang diberikan oleh aplikasi dan kualitas tampilan antar muka aplikasi. Hasil yang didapat dari uji coba melalui kuisioner adalah aplikasi berguna bagi user untuk menentukan pilihan tempat makan, kualitas informasi yang diberikan aplikasi baik dan tampilan antar muka dari aplikasi baik, yang berarti pengguna merasa puas dalam menggunakan aplikasi.

Kata kunci— Android, Google Places API, sistem rekomendasi, algoritma Slope One, restoran

Abstract

Food is one of the basic needs for human being. The needs of food will always increase unanimous with the number of people, so that many restaurants appear. Because of there are so many restaurants, it can arise a confusion when we want to choose a restaurant to eat. Therefore, an application which can give a restaurant recommendation will be built in this research. The recommendation given by the system is calculated using Slope One algorithm and the restaurants database is gathered from Google Places API. Slope One algorithm make the recommendation by summing the rating of a restaurants with the difference average to other restaurants. The application also had been tested to the user by using J.R.Lewis questionnaire with questions categories of application usefulness, information quality, and user interface quality. The results from the testing are user find the application useful to give the proper restaurant recommendation, the information quality is good, and the user interface quality is also good.

Keywords— Android, Google Places API, Recommendation System, Slope One algorithm, Restaurants

1. PENDAHULUAN

Makanan merupakan salah satu kebutuhan primer dari manusia [1]. Kebutuhan akan makanan akan selalu meningkat karena jumlah penduduk yang semakin banyak. Masalah yang sering muncul akibatnya banyak tempat makan yang tersedia adalah kebingungan untuk memilih tempat makan sehingga membutuhkan waktu yang lama untuk memutuskan pilihan tempat makan. Untuk menghemat waktu yang dibutuhkan untuk menentukan tempat makan dapat digunakan sebuah sistem rekomendasi tempat makan.

Saat ini banyak aplikasi yang bisa memberikan rekomendasi tempat makan yang ada di sekitar, seperti Foodspotting, Zomato dan Foursquare. Akan tetapi, kebanyakan dari aplikasi tersebut membutuhkan masukan dari pengguna lain untuk menampilkan daftar tempat makan yang ada di sekitarnya. Jika belum ada pengguna yang memberikan masukan untuk suatu tempat makan, maka tempat makan tersebut tidak akan ditampilkan pada aplikasi sebagai rekomendasi tempat makan.

Terdapat beberapa metode yang dapat digunakan untuk membuat sebuah sistem rekomendasi diantaranya *Collaborative filtering*, *Content-based filtering*, dan *Hybrid* [2]. Algoritma Slope One termasuk dalam *Item-based Collaborative filtering* yang merupakan pengembangan dari metode *Collaborative filtering* [3]. Perbedaan *Item-based Collaborative filtering* dengan *Collaborative filtering* adalah pada *Item-based*, hasil *rating* yang dibandingkan untuk memberikan rekomendasi [3].

Sebelum penelitian ini, penelitian mengenai aplikasi rekomendasi tempat makan sudah pernah dilakukan dengan judul Implementasi Algoritma Squeezer dan Term Frequency Ranking dalam Pembangunan Sistem Rekomendasi Tempat Makan [4]. Perbedaan penelitian ini dengan penelitian sebelumnya adalah pada penelitian sebelumnya digunakan algoritma Squeezer, sedangkan pada penelitian ini algoritma yang digunakan adalah algoritma Slope One, serta pada penelitian ini data yang digunakan didapatkan dari *Google Places API*, sedangkan pada penelitian sebelumnya data yang digunakan di-*input* secara manual oleh peneliti. Latar Belakang dipilihnya algoritma Slope One adalah algoritma Slope One hanya memerlukan *weight* dari setiap *item* yang dibandingkan, sehingga lebih cepat dalam memberikan rekomendasi dan memiliki hasil yang hampir sama akuratnya dibandingkan metode lainnya.

2. METODE PENELITIAN

2.1 Sistem Rekomendasi

Sistem rekomendasi adalah sebuah sistem informasi komputer yang membantu membuat keputusan untuk aktivitas bisnis dan organisasi. Sistem Rekomendasi dibuat dengan tujuan untuk memberikan rekomendasi kepada pengguna mengenai produk yang mungkin menarik untuk pengguna [5]. Beberapa metode yang digunakan pada sistem rekomendasi, antara lain:

2.1.1 Collaborative Filtering (CF)

Sistem *Collaborative Filtering* bekerja dengan cara mengumpulkan timbal balik dalam bentuk *rating-rating* dari *item* pada cakupan tertentu dan mencari kemiripan pada tingkah laku *rating* beberapa pengguna untuk menentukan rekomendasi terhadap suatu *item* [5].

2.1.2 Content-based Recommending

Content-based Recommending memberikan rekomendasi dengan cara membandingkan nilai representasi dari data *item* yang ada dengan ketertarikan dari pengguna [5].

2. 1.3 Hybrid

Pada metode *hybrid*, rekomendasi dibuat dengan cara *content-based* digunakan untuk mengubah matriks *rating user* menjadi matriks *rating* penuh, kemudian *collaborative filtering* digunakan untuk menyediakan rekomendasi [5].

2. 2 Algoritma Slope One

Algoritma Slope One adalah salah satu algoritma untuk membuat sistem rekomendasi. Slope one memberikan prediksi berdasarkan nilai hasil pencarian dari *item-item* yang dibandingkan. Keunggulan algoritma Slope One dibandingkan algoritma rekomendasi lainnya adalah algoritma Slope One mudah untuk diimplementasi, efisien saat melakukan *query*, tidak memerlukan banyak *requirement* dikarenakan rekomendasi berdasarkan *rating* dari setiap *item*, dan cukup akurat [6].

Algoritma Slope One melakukan perhitungan berdasarkan hubungan linear dari nilai preferensi atau *weight* dari setiap *item* yang dibandingkan. Estimasi umum dari dasar perhitungan algoritma Slope One adalah fungsi linear $y = mx + b$, dengan asumsi *gradient* $m = 1$, sehingga fungsi menjadi $b = y - x$. Cara kerja algoritma Slope One adalah dengan mencari selisih dari suatu *item* dengan *item-item* lain yang dibandingkan [7].

Perhitungan algoritma Slope One dapat diformulasikan dengan persamaan (1) untuk pencarian selisih [8]:

$$dev_{j,i} = \sum_{u \in S_{j,i}(x)} \frac{u_j - u_i}{card(S_{j,i}(x))} \quad (1)$$

dimana

$dev_{j,i}$ = rata-rata selisih *rating item j* dan *i*

u_j = *rating item j*

u_i = *rating item i*

$card(S_{j,i}(x))$ = banyaknya elemen yang dibandingkan

Apabila selisih sudah didapatkan, maka dapat dilakukan perhitungan rekomendasi untuk *item j* yang dapat dirumuskan dengan persamaan (2):

$$p^{SI}(u)_j = dev_{j,i} + u_j \quad (2)$$

dimana

$p^{SI}(u)_j$ = nilai rekomendasi untuk *item j*

Berdasarkan persamaan di atas, algoritma Slope One memberikan rekomendasi dengan melakukan perhitungan selisih dari setiap *item*. Selisih yang didapat akan dirata-rata per *item* yang kemudian akan dijumlahkan dengan *value* dari masing-masing *item*. *Value* yang sudah dijumlahkan dengan rata-rata selisihnya akan digunakan sebagai *point* untuk memberikan rekomendasi [8]. Gambar 1 memperlihatkan *pseudocode* algoritma Slope One yang diterapkan dalam penelitian ini.

```

BEGIN
  For every item i
    For every other item j
      For every user u expressing preference for
        both i and j
        Add the difference in u's preference for
          i and j to an average
      //Prediction Part
      For every item i the user u expresses no
        preference for
        For every item j that user u expresses a
          preference for
            Find the average preference difference
              between j and i
            Add this diff to u's preference value for
              j
            Add this to a running average
        Return the top items, ranked by these averages
END

```

Gambar 1 Pseudocode Algoritma Slope One

2.3 Survei

Survei adalah metode pengumpulan data dengan memberikan pertanyaan kepada beberapa orang yang berhubungan dengan penelitian. Ukuran sampel yang diterima sangat bergantung pada jenis penelitian, antara lain sebagai berikut [9]:

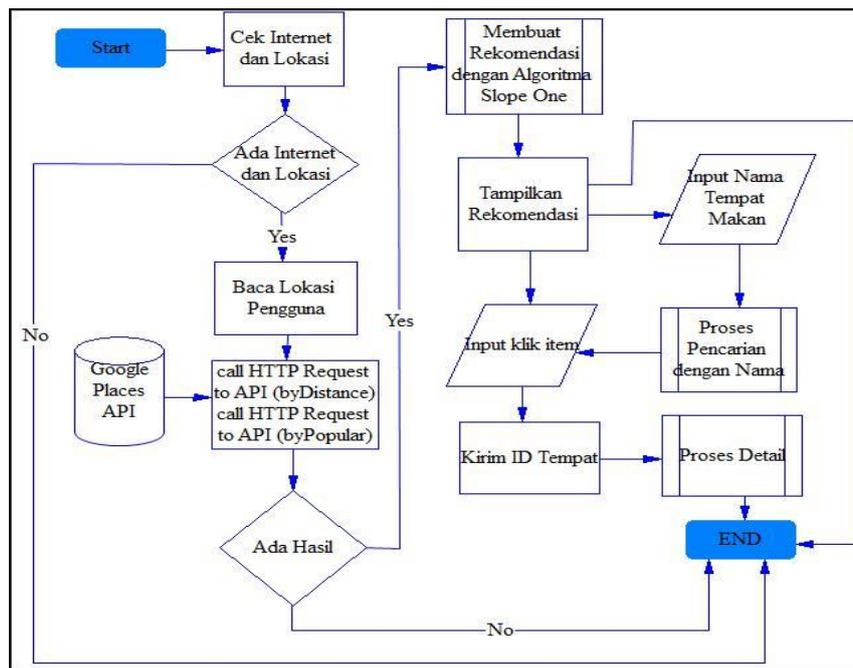
1. Penelitian deskriptif, sampel minimumnya adalah 10% dari populasi.
2. Penelitian korelasional, sampel minimumnya adalah 30 subjek.
3. Penelitian kausal perbandingan, sampel sebanyak 30 subjek per kelompok.
4. Penelitian eksperimental, sampel minimumnya adalah 15 subjek per kelompok.

Beberapa panduan lainnya untuk menentukan ukuran sampel yaitu [9]:

1. Ukuran sampel lebih dari 30 dan kurang dari 500 adalah ukuran yang digunakan oleh mayoritas peneliti.
2. Jika sampel dipecah ke dalam subsampel (pria/wanita, junior/senior, dan sebagainya), maka ukuran sampel minimum 30.
3. Penelitian multivariate (termasuk analisis regresi berganda), ukuran sampel 10 kali lebih besar dari jumlah variabel dalam penelitian.
4. Penelitian eksperimental sederhana dengan kontrol eksperimen yang ketat, ukuran sampel kecil antara 10 sampai dengan 20 orang.

3. HASIL DAN PEMBAHASAN

Pada penelitian ini, aplikasi sistem rekomendasi tempat makan dengan nama *Taste* diawali dengan menampilkan *progress dialog* dan menampilkan *list* rekomendasi tempat makan setelah berhasil mendapatkan daftar tempat makan di sekitar pengguna. Penjelasan lebih lanjut mengenai alur proses yang dilakukan dalam aplikasi sistem rekomendasi ini dijelaskan dengan *flowchart*, seperti yang diperlihatkan pada Gambar 2.



Gambar 2 Flowchart Sistem Rekomendasi Tempat Makan Taste

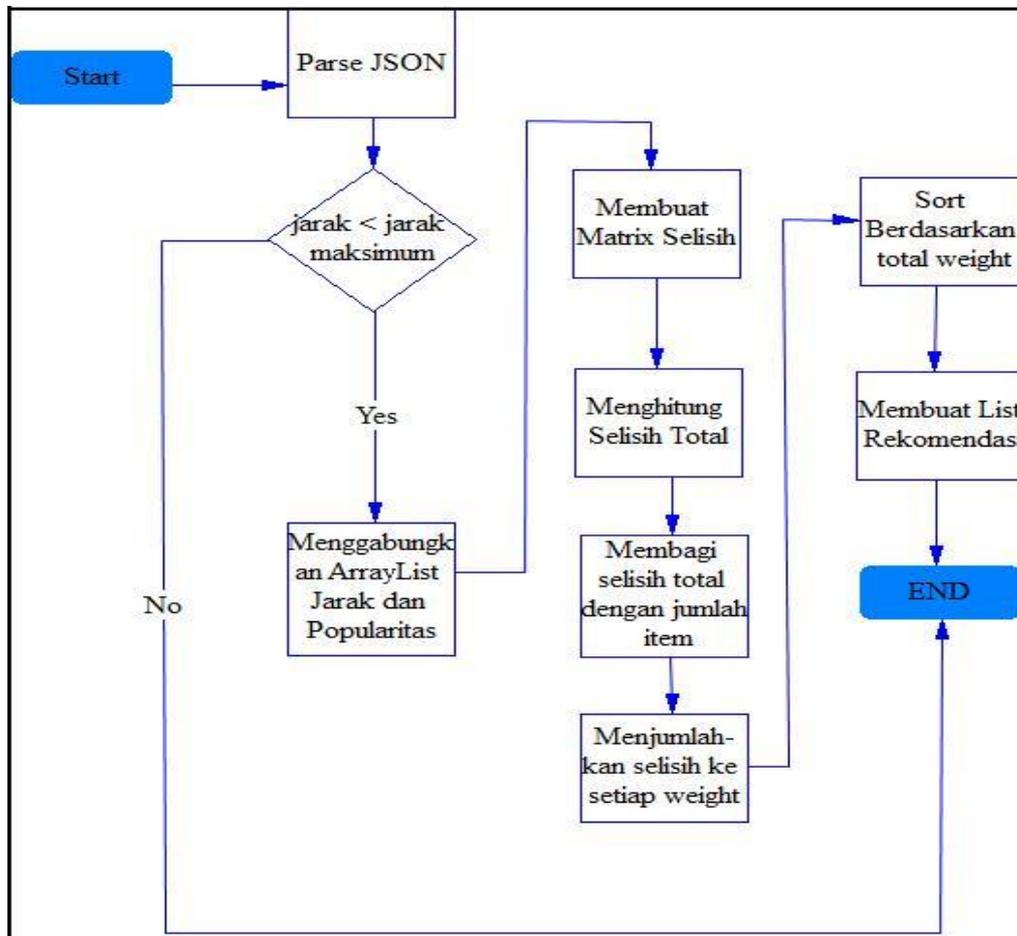
Pada Gambar 2 di atas, aplikasi ini akan digunakan oleh *user* dan pada saat *user* membuka aplikasi *Taste*, maka aplikasi akan langsung melakukan pengecekan apakah fitur internet dan lokasi pada *smartphone user* telah aktif. Jika fitur internet dan lokasi telah aktif, maka aplikasi akan langsung membaca lokasi *user* menggunakan *Google Play Services*. *Google Play Services* akan mengembalikan *longitude* dan *latitude* dari posisi *user*. Setelah lokasi didapatkan, aplikasi akan melakukan *request* ke *Google Places API* untuk mengambil data tempat makan. *Request* dilakukan sebanyak dua kali, yaitu *request* untuk data tempat makan berdasarkan tingkat popularitasnya dan berdasarkan jarak terdekat.

Setelah melakukan *request* ke *Google Places API*, aplikasi akan menerima hasil berupa *JSON (JavaScript Object Notation)*. *JSON* akan di-*parsing* dan dimasukkan ke dalam *arraylist*. Setelah data di-*parsing*, data akan diproses dengan algoritma *Slope One*, sehingga diperoleh *arraylist* rekomendasi. Rekomendasi dibagi menjadi tiga tempat makan per halaman. Setelah perhitungan rekomendasi selesai dilakukan, akan ditampilkan halaman utama dari aplikasi. Jika salah satu *item* diklik oleh *user*, maka detail dari tempat tersebut akan ditampilkan. Detail berupa alamat lengkap, *website*, foto-foto dan gambaran lokasi tempat makan. Selain itu, aplikasi juga bisa mencari tempat makan berdasarkan nama. Jika ada hasil, maka akan ditampilkan *list* dengan nama yang dicari dan jika di-klik maka akan dimunculkan detail dari tempat tersebut.

Gambar 3 menunjukkan alur algoritma *Slope One* yang diterapkan dalam aplikasi. Algoritma *Slope One* digunakan untuk memberikan rekomendasi tempat makan berdasarkan tingkat popularitas dan jarak terdekat. Proses perhitungan algoritma *Slope One* dimulai dengan mem-*parsing* *JSON* yang didapat setelah membuat *HTTP Request* tempat makan pada *Google Places API*. *JSON result* yang didapat akan di-*parsing* ke dalam *arraylist tempDistance* dan *tempPopular*. Setelah melakukan *parsing* *JSON*, maka akan dilakukan perbandingan antara jarak tempat makan ke *user* dengan jarak maksimum. Jika jarak tempat makan kurang dari jarak maksimum, maka tempat makan akan ditampilkan. Setelah *arrayList* berhasil di-*filter*, maka akan dilakukan penggabungan *arrayList tempDistance* dan *tempPopular* dalam sebuah *arraylist venuesList*.

Langkah berikutnya adalah membuat matriks selisih dari *weight* setiap tempat dengan *weight* tempat lainnya. *Weight* dari setiap tempat akan dikurangi dengan *weight* tempat lainnya

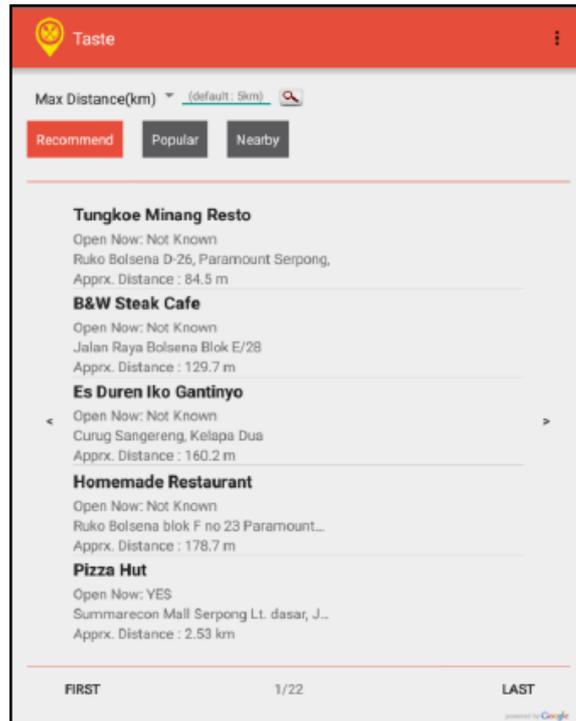
dan dimasukkan ke dalam *arrayList*. Setelah terbentuk matriks selisih *weight*, maka setiap tempat akan dihitung total selisih *weight*-nya. Total *weight* kemudian dijumlahkan dengan rata-rata *weight* setiap tempat untuk mendapatkan poin *Slope One*. *ArrayList venuesList* akan diurutkan sesuai dengan poin *Slope One* secara *descending*. Kemudian *arrayList venuesList* akan di-*set* ke *ListView adapter* sebanyak tiga *item* per halaman.



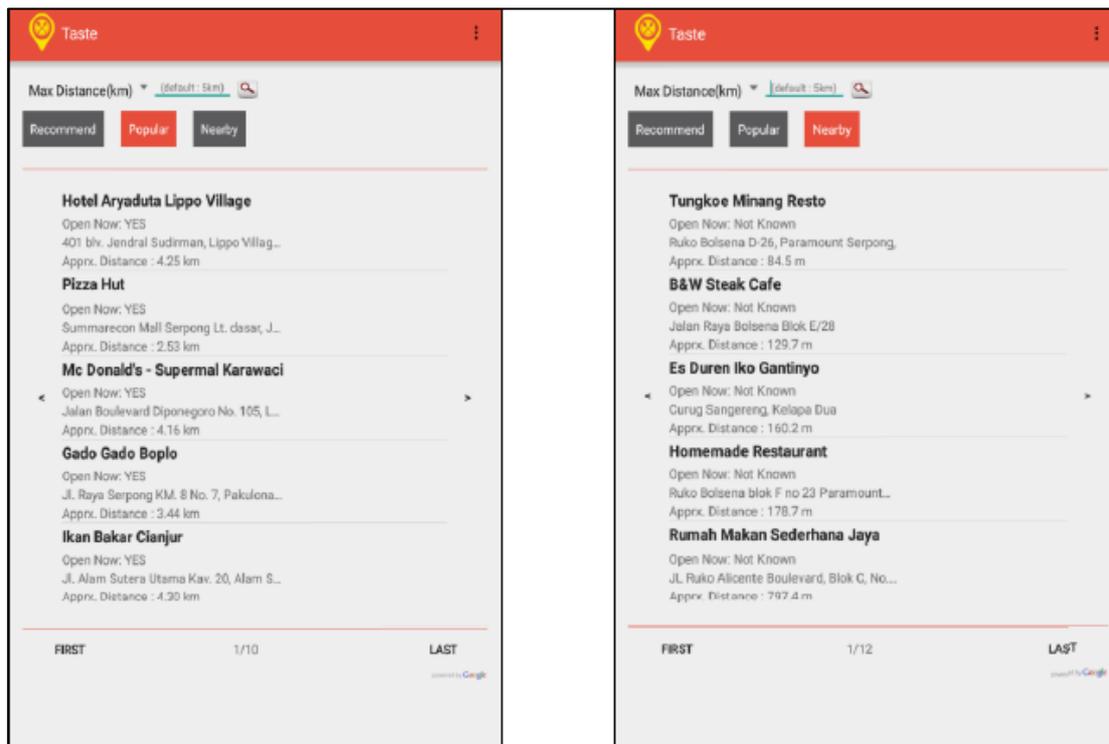
Gambar 3 *Flowchart* Algoritma Slope One

Penjelasan bab ini dilanjutkan dengan contoh hasil akhir aplikasi yang berhasil dibangun dengan menggunakan perancangan dan metode yang telah dijelaskan sebelumnya. Gambar 4 memperlihatkan halaman utama aplikasi *Taste* saat dijalankan. Pada halaman utama tersebut akan ditampilkan *list* rekomendasi tempat makan di sekitar *user*. Pada awal aplikasi dimulai, aplikasi akan *me-load list* tempat makan yang ada di sekitar *user* dan menghitung rekomendasi menggunakan algoritma Slope One. Hasil rekomendasi tempat makan di sekitar *user* ditampilkan dengan menggunakan *pagination*.

Selain *list* rekomendasi, aplikasi juga menampilkan *list* tempat makan berdasarkan tingkat popularitas dan jarak terdekat berdasarkan lokasi *user* dan jarak maksimum yang diberikan.



Gambar 4 Halaman Utama Aplikasi



Gambar 5 Daftar Tempat Makan Berdasarkan Tingkat Popularitas dan Jarak Terdekat

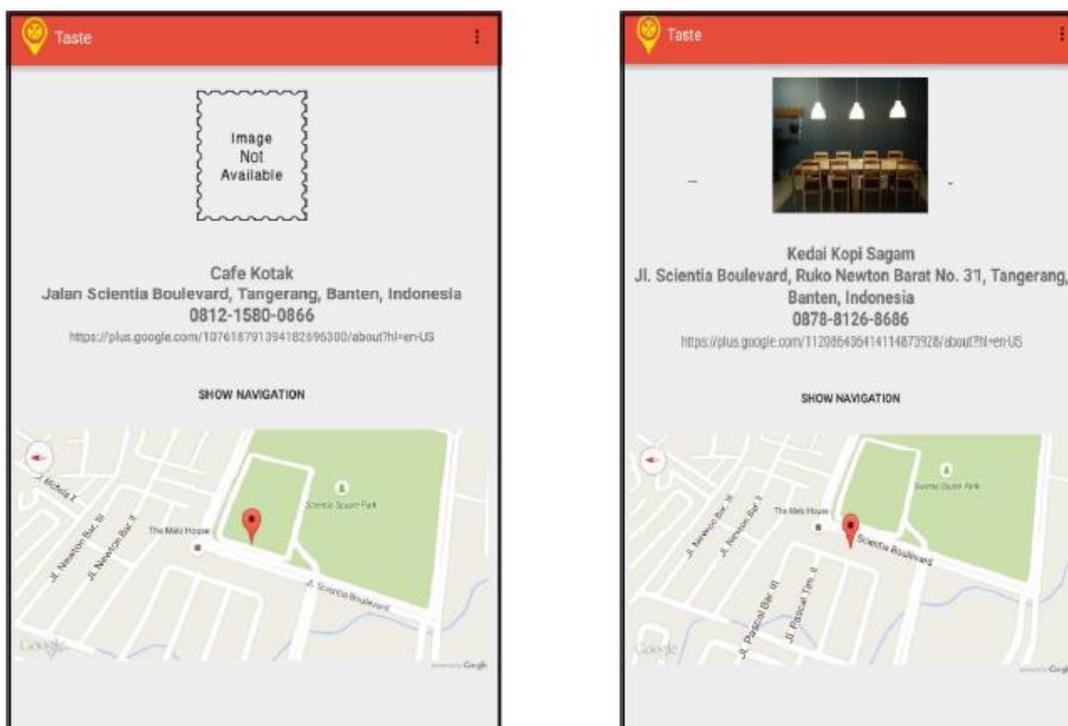
Pada Gambar 5 diperlihatkan tampilan aplikasi saat tombol *Popular* dan tombol *Nearby* ditekan. Pada saat tombol *Popular* ditekan, maka aplikasi akan menampilkan tempat makan berdasarkan popularitas dengan jarak maksimum 5 km, seperti yang ditunjukkan Gambar 5 bagian kiri. *List* yang ditampilkan adalah hasil *load list* saat aplikasi pertama dijalankan. Gambar 5 bagian kanan menunjukkan aplikasi saat tombol *Distance* ditekan dan aplikasi akan

menampilkan tempat makan berdasarkan jarak terdekat dengan jarak maksimum 5 km. Jarak maksimum untuk mencari tempat makan dapat diubah pada *search bar* yang terdapat pada bagian atas aplikasi.

Jika salah satu *item* pada *Listview*, baik pada halaman hasil pencarian ataupun halaman utama diklik oleh *user*, maka akan ditampilkan detail dari tempat makan tersebut. Aplikasi akan mengirimkan *id* dari tempat yang diklik ke halaman detail dan membuka halaman detail seperti yang diperlihatkan Gambar 6.

Saat halaman detail dibuka, aplikasi akan me-load detail dari tempat makan dengan cara mengirimkan *id* dari tempat makan tersebut ke *Google Places API*. Setelah didapatkan JSON detail dari *Google Places API*, aplikasi akan me-parsing JSON detail dan melakukan pengecekan apakah terdapat *photo_reference* dari tempat tersebut. *Photo_reference* digunakan untuk mendapatkan gambar dari tempat makan. Jika terdapat *photo_reference*, maka aplikasi akan mengambil gambar dari *Google Places API* dengan mengirimkan *photo_reference*. Setiap *photo_reference* memiliki satu gambar. Gambar akan diubah ke dalam *bitmap* dan ditampilkan pada *horizontalScrollView* yang terdiri dari banyak *imageView*. Gambar 6 bagian kiri merupakan tampilan tempat makan yang tidak memiliki *photo_reference*, sehingga ditampilkan keterangan gambar tidak tersedia. Gambar 6 bagian kanan merupakan tampilan detail tempat makan yang memiliki satu *photo_reference*.

Pada bagian bawah aplikasi terdapat peta yang menunjukkan lokasi dari tempat makan tersebut. Untuk melihat navigasi menuju tempat makan, *user* dapat meng-klik tombol *Show Navigation* dan aplikasi akan membuka *Google Maps* yang langsung menunjukkan navigasi menuju tempat tujuan.



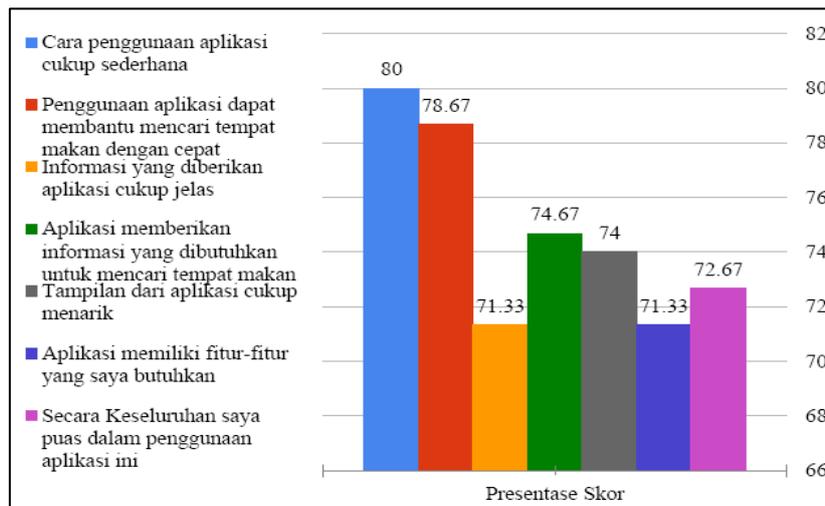
Gambar 6 Detail Tempat Makan

Selanjutnya pengujian dilakukan dengan cara membandingkan perhitungan manual dari algoritma dengan hasil dari aplikasi. Data yang digunakan pada uji coba diambil koordinat *latitude* -6.2626802 dan *longitude* 106.6203501. Pengujian dilakukan untuk mengetahui berhasil atau tidaknya algoritma Slope One diterapkan pada aplikasi. Berdasarkan hasil pengujian dengan membandingkan hasil perhitungan manual dengan hasil pada aplikasi, aplikasi sistem rekomendasi tempat makan telah berhasil dibuat berdasarkan algoritma Slope

One, karena hasil rekomendasi yang diberikan aplikasi sama dengan hasil pada uji coba perhitungan manual.

Uji coba selanjutnya dilakukan dengan melakukan survey terhadap 30 orang responden. Pertanyaan yang diajukan berjumlah tujuh dan diambil dari kuisioner J.R. Lewis mengenai *Computer System Usability*. Pertanyaan yang digunakan dibagi dalam tiga kategori, yaitu *system usefulness*, *information quality*, dan *interface quality* [10]. Skala yang digunakan pada kuisioner didapat dari skala Likert yang terdiri dari lima jenjang. Skala yang digunakan yaitu skala lima untuk menyatakan sangat setuju, skala empat untuk menyatakan setuju, skala tiga untuk menyatakan netral, skala dua untuk menyatakan tidak setuju, dan skala satu untuk menyatakan sangat tidak setuju [11].

Gambar 7 memperlihatkan rekapitulasi hasil survey dengan persentase tiap butir pertanyaan yang diajukan dalam kuisioner.



Gambar 7 Rekapitulasi Hasil Survey

Berdasarkan hasil perhitungan skor akhir yang didapatkan dari rekapitulasi kuisioner, dapat dihitung respons dari pengguna terhadap tiga kategori pertanyaan berdasarkan kuisioner J.R. Lewis, yaitu *system usefulness* yang terdapat pada pertanyaan nomor satu, dua, dan tujuh. Skor untuk *system usefulness* dari aplikasi adalah 77,11% yang berarti aplikasi berguna bagi pengguna. Kategori *information quality* dari aplikasi terdapat pada pertanyaan nomor tiga dan empat, memiliki skor 73% yang berarti informasi yang diberikan aplikasi baik. Kategori *interface quality* dari aplikasi terdapat pada pertanyaan nomor lima dan enam yang memiliki skor dari hasil perhitungan skor akhir sebesar 72,67% yang berarti kualitas tampilan dari aplikasi baik. Secara keseluruhan, tingkat kepuasan pengguna aplikasi dalam menggunakan aplikasi sistem rekomendasi tempat makan adalah 74,26% yang termasuk dalam kategori positif. Artinya pengguna puas dalam menggunakan aplikasi sistem rekomendasi tempat makan.

4. KESIMPULAN

Berdasarkan hasil dari perhitungan manual, rancang bangun aplikasi sistem rekomendasi tempat makan berbasis Android telah berhasil dibuat berdasarkan algoritma Slope One, dengan kriteria jarak dan tingkat popularitas dari tempat makan dalam memberikan rekomendasi. Pada uji coba terhadap pengguna saat menggunakan aplikasi yang diambil melalui kuisioner yang dibuat berdasarkan kuisioner J.R. Lewis dengan jumlah sampel 31 orang dan penilaian dengan skala satu sampai dengan lima. Uji coba dilakukan untuk mengetahui tingkat kepuasan pengguna dalam penggunaan aplikasi berdasarkan tiga kategori, yaitu tingkat

kegunaan aplikasi, kualitas informasi yang diberikan aplikasi, dan kualitas tampilan antar muka aplikasi. Berdasarkan hasil rekapitulasi kuisioner, dapat disimpulkan bahwa tingkat kegunaan aplikasi mencapai 77,11%, kualitas informasi yang diberikan mencapai 73%, dan kualitas tampilan antar muka aplikasi mencapai 72,67%. Secara keseluruhan, pengguna puas dalam menggunakan aplikasi sistem rekomendasi tempat makan *Taste* dengan nilai 74,26% yang termasuk dalam kategori positif.

5. SARAN

Dari hasil penelitian ini, terdapat beberapa saran untuk pengembangan lebih lanjut, yaitu:

1. Dapat dilakukan pengembangan menggunakan algoritma lain selain algoritma Slope One, contohnya *simple additive weighting*. Algoritma Slope One hanya menggunakan nilai skor dari masing-masing *item*, sedangkan pada *simple additive weighting* nilai dari preferensi *user* dapat digunakan sebagai faktor untuk membuat rekomendasi.
2. Dapat ditambahkan kategori tempat makan, sehingga *user* dapat memilih tempat makan sesuai dengan kategori yang diinginkan.

DAFTAR PUSTAKA

- [1] Shodiqin, A, 2015, Jenis-Jenis dan Contoh Kebutuhan Manusia Primer, Skunder dan Tersier, <http://www.ilmuekonomi.net/2015/12/jenis-jenis-dan-contoh-kebutuhan-manusia-primer-skunder-dan-tersier.html>, diakses tgl 8 Maret 2016.
- [2] Kim, B. M., Li, Q., Park, C. S., Kim, S. G., dan Kim, J.Y., 2006, A New Approach for Combining Content-Based and Collaborative Filters, *Journal of Intelligent Information Systems*, Vol.27, Issue 1, hal.79-91.
- [3] Sarwar, B., Karypis, G., Konstan, J., dan Riedl, J., 2001, Item-Based Collaborative Filtering Recommendation Algorithms, *Proceeding of WWW10*, Hong Kong, May 1-5.
- [4] Wirawan, V., Hansun, S., dan Nugroho, H. T., 2014, Implementasi Algoritma Squeezer dan Term Frequency Ranking dalam Pembangunan Sistem Rekomendasi Tempat Makan, *ULTIMA Computing*, Vol.VI, No.1, hal.25-32.
- [5] Melville, P. dan Sindhvani, V., 2010, *Recommender Systems*, Sammut, C. dan Webb, G. I. (ed.): *Encyclopedia of Machine Learning*, Springer, USA.
- [6] Jiang, T. dan Lu, W., 2013, Improved Slope One Algorithm Based on Time Weight, *Applied Mechanics and Materials*, Vols.347-350, hal.2365-2368.
- [7] Clevesy, L., 2010, Slope-one Recommender - Exclusive Article from Mahout in Action, <https://dzone.com/articles/slope-one-recommender>, diakses tgl 15 Maret 2015.
- [8] Masruri, F. dan Mahmudy, W. F., 2007, Personalisasi Web E-Commerce Menggunakan Recommender System dengan Metode Item-based Collaborative Filtering, *Kursor*, Vol.3, No.1, hal.1-12.
- [9] Hendryadi, 2012, Menentukan Ukuran Sampel Sederhana, <http://teorionline.net/menentukan-ukuran-sampel-menurut-para-ahli/>, diakses tgl 7 Juli 2015.
- [10] Lewis, J. R., 1995, IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use, *International Journal of Human-Computer Interaction*, Vol.7, Issue 1, hal.57-78.
- [11] Jainuri, M., 2015, Skala Pengukuran, http://www.academia.edu/5077784/Skala_Pengukuran, diakses tgl 13 Agustus 2015.