

BAB II TINJAUAN PUSTAKA

2.1 Data Mining

Data Mining merupakan proses eksplorasi dan analisis, oleh cara otomatis atau semi otomatis, dalam jumlah besar data untuk menemukan pola dan aturan yang bermakna. (Nikita & Srivastava, 2013). Secara umum fungsi *data mining* digunakan untuk menentukan jenis pola yang dapat ditemukan dalam tugas-tugas *data mining*. Tugas *data mining* dapat diklasifikasikan dalam dua kategori-deskriptif dan prediktif. *Data mining* deskriptif menggambarkan sifat umum dari data dalam *database*. Tugas *data mining* prediktif melakukan inferensi pada data saat ini untuk membuat prediksi (Nikita & Srivastava, 2013).

2.1.1. Pengelompokan *Data Mining*

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan (Baradwaj & Pal, 2012), yaitu :

1. Deskripsi: digunakan untuk menggambarkan pola dan kecenderungan yang terdapat dalam data.
2. Estimasi: model menggunakan *record* lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi.
3. Prediksi: prediksi nilai dari hasil akan ada di masa mendatang.
4. Klasifikasi: contoh kasusnya adalah penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.
5. Pengklusteran: merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan.
6. Asosiasi: Tugas asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam satu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang belanja.

2.1.2. Elemen Pada *Data Mining*

Data mining terdiri dari lima elemen utama (Nikita & Srivastava, 2013)., diantaranya :

1. Mengekstrak, mengubah, dan memuat data transaksi kesistem data warehouse.
2. Menyimpan dan mengelola data dalam multidimensisistem basis data.
3. Menyediakan akses data ke analis bisnis dan profesional teknologi informasi.
4. Menganalisa data dengan perangkat lunak aplikasi.
5. Sajikan data dalam format yang bermanfaat, seperti grafik atau meja.

2.2 Sistem Rekomendasi

Sistem rekomendasi adalah alat dan teknik perangkat lunak yang menyediakan saran untuk item yang kemungkinan besar menarik bagi pengguna tertentu (Ricci, et al, 2011). Saran-saran tersebut berhubungan dengan proses pengambilan keputusan, seperti *item* mana yang akan dipilih. Sistem rekomendasi mengandalkan dua tipe *input* yang berbeda, yaitu *explicit input* dan *implicit input* (Hu, et al, 2008). *Explicit input* didapat dari hasil penilaian yang diberikan langsung oleh pengguna, misalnya pemberian *rating* atau *thumbs-up/down* untuk *item* tertentu. Sedangkan *implicit input* didapat dengan cara mengamati kebiasaan pengguna. Misalnya catatan penelusuran, pola pencarian, atau bahkan *click-stream*.

Awalnya Rekomendasi sistem didasarkan pada pemfilteran berdasarkan konten dan demografis dan penyaringan kolaboratif. Sistem memasukkan informasi sosial untuk meningkatkan kualitas proses rekomendasi. Untuk perbaikan rekomendasi proses di masa depan, sistem rekomendasi akan menggunakan data pribadi, informasi lokal dan implisit dari *internet* (Thorat, et al, 2015).

Sistem rekomendasi memerlukan model yang tepat agar yang direkomendasikan sesuai dengan keinginan pengguna, serta mempermudah pengguna mengambil keputusan yang tepat dalam menentukan pilihannya (McGinty & Smyth, 2003). Sistem rekomendasi yang ada pada saat ini terdiri dari empat tipe yaitu *content based filtering*, *collaborative filtering*, *knowledge based recommender system* dan *hybrid recommender system* (Jannach, et al, 2018).

2.2.1. Content-Based Filtering

Content-based filtering memberikan rekomendasi dengan menganalisis deskripsi item-item itu telah diberi peringkat oleh pengguna dan deskripsi item yang akan direkomendasikan. Semakin banyak algoritma yang dimiliki telah diusulkan untuk menganalisis konten dokumen teks dan menemukan kesamaan dalam konten ini yang dapat ditayangkan sebagai dasar untuk membuat rekomendasi.

Dalam sistem pencarian informasi, langkah pertama adalah mengidentifikasi kata kunci untuk mewakili dokumen, Itumenghindari pengindeksan kata-kata yang tidak berguna, sistem pencarian teks sering mengaitkan daftar berhenti dengan seperangkat dokumen. Itu kata-kata yang tidak relevan disebut *stop list* (*the, of, for, with, dll*) (Manjula & Chilambuchelvan, 2016).

Matriks frekuensi istilah (berat) TF (d,t) mengukur keterkaitan suatu istilah dengan sehubungan dengan dokumen yang diberikan d. Ini didefinisikan sebagai 0 jika dokumen tidak mengandung istilah dan bukan nol sebaliknya. Itu frekuensi istilah relatif diukur menggunakan frekuensi istilah versus jumlah total kemunculan semua ketentuan dalam dokumen. Frekuensi istilah dihitung.

$$TF(d,t) = \begin{cases} 0 & \text{if } freq(d,t) = 0 \\ 1 + \log(1 + \log(freq(d,t))) & \text{otherwise} \end{cases} \quad (1)$$

Keterangan :

TF :banyaknya kata yang dicari pada sebuah dokumen

d : dokumen ke- d

t : kata ke- t dari kata kunci

Ada ukuran penting lainnya, yang disebut frekuensi dokumen terbalik (IDF) dalam Persamaan.1 yang mewakili faktor penskalaan, atau pentingnya istilah t dan itu akan berkurang jika istilah t muncul di banyak dokumen. Untuk contoh istilah informasi mungkin kurang penting dalam banyak makalah penelitian. Formula untuk IDF (t) diberikan dalam Persamaan.2.

$$IDF(t) = \frac{\log 1 + |d|}{|dt|} \quad (2)$$

Keterangan :

IDF : *Inversed Document Frequency*

d : dokumen ke-*d*

t : kata ke-*t* dari kata kunci

Di mana, *d* adalah koleksi dokumen, dan *dt* adalah kumpulan dokumen yang berisi istilah *t*. Dalam ruang vektor lengkap

model, TF dan IDF digabungkan bersama, yang membentuk ukuran TF-IDF diberikan dalam Persamaan.3:

$$TF\text{-}IDF(d,t) = TF(d,t) * IDF(t)$$

(3)

Keterangan :

TF : banyaknya kata yang dicari pada sebuah dokumen

IDF : *Inversed Document Frequency*

d : dokumen ke-*d*

t : kata ke-*t* dari kata kunci

Untuk menentukan kesamaan antara dua dokumen, cosine similarity digunakan. Salah satu yang paling jelas. Kelebihan dari algoritma *content-based filtering* adalah algoritma ini tidak perlu domain pengetahuan. Inicukup untuk mengumpulkan umpan balik dari pelanggan tentang prioritas mereka. Keuntungan selanjutnya dari *content-based filtering* yang dapat kita pertimbangkan adalah, algoritma ini lebih baik daripada *Collaborative Filtering (CF)*.

a. *Measure Similarity*

Metode *measure similarity* memeriksa kelengkapan atau sampel data *item* pengguna untuk membuat sebuah prediksi. Setiap pengguna adalah bagian dari sekelompok orang dengan minat yang sama. Dengan mengidentifikasi dugaan tetangga dari suatu pengguna aktif prediksi selera pada item baru baginya akan dihasilkan. Aturan *Neighborhood-Based Collaborative Filtering*, menggunakan langkah-langkah sebagai berikut:

- i. Hitung persamaan atau berat, w_{ij} , yang mencerminkan jarak, korelasi, atau berat, antara keduanya pengguna atau 2 item, i dan j .
- ii. Hasilkan prediksi untuk pengguna aktif dengan mengambil rata-rata dari semua peringkat pengguna atau item atau pengguna tertentu. Ketika tugasnya membangun rekomendasi $top-N$, dilakukan pencarian pengguna atau item yang paling mirip (terdekat dengan tetangga) menghitung kesamaan, jadi agregasikan tetangga untuk mendesak item paling sering $top-N$ sebagai rekomendasi.

b. *Correlation-Based Similarity*

Dalam hal ini, kesamaan antara dua pengguna u dan v , atau antara dua item dan, dihitung dengan menghitung *correlation-based similarity* yang berbeda. *correlation-based similarity* mengukur sejauh dua variabel secara linear berhubungan satu sama lain. Untuk aturan algoritma, korelasi Pearson antara pengguna u dan v diberikan dalam Persamaan 4.

$$W_{uv} = \frac{\sum_{i \in I} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{vi} - \bar{r}_v)^2}} \quad (4)$$

Keterangan :

Di mana $i \in I$ meringkas item yang dinilai pengguna dan pengguna dan merupakan peringkat rata-rata item berkorelasi dari pengguna. Dalam algoritma berbasis item, set pengguna yang dilambangkan oleh $u \in U$ yang memberi peringkat kedua item i dan j , maka *correlation person* diberikan dalam Persamaan.5.

$$W_{uv} = \frac{\sum_{i \in I} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{vi} - \bar{r}_v)^2}} \quad (5)$$

Keterangan :

Di mana r_{ui} adalah peringkat pengguna u pada *item* i , adalah peringkat rata-rata item ke- i oleh pengguna tersebut.

c. *Vector Cosine-Based Similarity*

Vector cosine similarity antara *item* i dan j diberikan dimana " \bullet " menunjukkan titik-produk dari dua vektor. Sebuah matriks kesamaan $n \times n$ dihitung untuk mendapatkan perhitungan kesamaan yang diinginkan, untuk n *item*. Jika :

Persamaan vector kosinus antara A dan B yang diberikan dalam persamaan 6.

$$W_{A,B} = \cos(\vec{A} \cdot \vec{B}) = \frac{x_1x_2 + y_1y_2}{\sqrt{x_1^2 + y_1^2} \sqrt{x_2^2 + y_2^2}}$$

(6)

Keterangan :

Vektor $A = \{x_1, y_1\}$, vektor $B = \{x_2, y_2\}$.

Jadi, kesimpulannya, pilihan harus dibuat di antara semua ukuran kesamaan. Poin yang perlu diingat, diantaranya :

- i. Jika data tunduk pada tingkat inflasi yaitu (pengguna yang berbeda mungkin menggunakan skala yang berbeda) maka gunakan *pearson correlation coefficient*.
- ii. Jika data padat yaitu (jika hampir semua atribut memiliki nilai bukan nol) dan besarnya nilai atribut adalah yang penting, gunakan pengukuran jarak seperti Euclidean atau *Manhattan*.
- iii. Jika datanya jarang, pertimbangkan untuk menggunakan *cosine-similarity*.

2.2.2. Collaborative Filtering

Collaborative filtering merupakan metode sistem rekomendasi yang pada dasarnya memprediksi *item* yang sama berdasarkan ketertarikan antar pengguna (Lee, 2015). Pada *collaborative filtering*, pengguna aktif akan memberikan rekomendasi kepada pengguna lain yang memiliki preferensi yang sama berdasarkan *similarity* dari tiap tiap pengguna. Proses *collaborative filtering* dilakukan dengan melihat persamaan *rating* pada pekerjaan dengan metode *user to user*.

Similarity antara pengguna dapat dihitung berdasarkan dari *rating* yang diberikan pada oleh pengguna. Metode *collaborative filtering* juga sering disebut sebagai “*people-to-people correlation*”. Proses pada *collaborative filtering* dimulai dengan menghitung *similarity* antar pengguna yang memiliki ketertarikan yang sama. Kemudian dihitung prediksi interaksi menggunakan persamaan 7 (Rahmawati, et al, 2018).

$$S_{uj} = \frac{\sum_{x \in I_{uj}} (U_x - U)(J_x - J)}{\sqrt{\sum_{x \in I_{uj}} (U_x - U)^2 \sum_{x \in I_{uj}} (J_x - J)^2}} \dots \quad (7)$$

Keterangan:

- S adalah *similarity* antara pengguna *u* dan pengguna *j*
- U adalah interaksi pengguna *u* pada pekerjaan *x*
- U adalah rata rata interaksi dari pengguna *u*
- J adalah interaksi dari pengguna *j* pada pekerjaan *x*
- j adalah rata rata interaksi dari pengguna *j*

$$U_x = U + \frac{\sum_{j \in \text{Raters of } x} (J_x - J) S_{uj}}{\sum_{j \in \text{Raters of } x} |S_{uj}|} \dots \quad (8)$$

Keterangan :

- U adalah prediksi interaksi pengguna *u* pada pekerjaan *x*
- u adalah rata rata interaksi dari pengguna *u*
- J adalah interaksi dari pengguna *j* pada pekerjaan *x*
- S adalah *similarity* antara pengguna *u* dan pengguna *j*
- j adalah rata rata interaksi dari pengguna *j*

2.2.3. Knowledge Based Recommender System

Knowledge based system merekomendasikan pekerjaan berdasarkan *domain knowledge*, tentang pengaruh fitur dari pekerjaan terhadap kebutuhan dan preferensi pribadi pengguna. Pada kasus ini *similarity score* pada *knowledge based system* merepresentasikan kesamaan fitur antara pekerjaan yang berbeda. Nilai *similarity* antar individu bisa didapatkan dengan persamaan 6. *Similarity* direpresentasikan dengan *reference matrix* (Rahmawati, et al, 2018).

$$S_{ab} = \sum_{i=1}^{\#p} \left(\frac{\text{common}(a, b, P[i])}{\max(\text{deg}(a, P[i]), \text{deg}(b, P[i]))} \right) * \text{Weight}(P[i]) \quad (9)$$

Keterangan:

- P (*properties*) adalah vektor yang mengandung sejumlah atribut dan hubungan antar atribut pada *reference class* dalam formula tersebut digunakan untuk menghitung *similarity*
- $\text{deg}(a, P)$ merepresentasikan jumlah *instance* yang terhubung dengan individual a melalui relasi P
- $\text{common}(a, b, P)$ merepresentasikan jumlah *instance* sama yang terhubung dengan a dan b melalui relasi dari *property* P
- $\text{Weight}(P)$ menandakan pentingnya suatu *property* pada relasi P dalam formula.

2.2.4. Hybrid Recommender System

Hybrid recommender system adalah suatu teknik yang menggabungkan beberapa teknik rekomendasi (Rahmawati, et al, 2018). *Hybrid recommender system* memiliki teknik *hybrid* seperti yang dijelaskan pada Tabel 2.1.

Tabel 2.1 Teknik *Hybrid*

Teknik <i>Hybrid</i>	Deskripsi
<i>Weighted</i>	Nilai Pada rekomendasi antara beberapa teknik di gabungkan berdasarkan parameter tertentu
<i>Switching</i>	Sistem menggabungkan 2 buah teknik secara bergantian
<i>Mixed</i>	Rekomendasi dari beberapa recommender disatukan
<i>Cascade</i>	Suatu rekomendasi memperbaiki hasil dari rekomendasi lain
<i>Feature Augmentation</i>	Output dari suatu teknik digunakan untuk input dari teknik lainnya
<i>Meta-level</i>	Model yang dipelajari dari suatu recommender digunakan untuk input pada recommender lain

Jenis *input* pada sistem rekomendasi tergantung pada jenis algoritma *filtering* yang digunakan. Berikut penjelasan *rating* pendapat pengguna berdasarkan *item* yang dipilih (Mueller, 2014), diantaranya :

- a. *Explicit Ratings*, pengguna diharuskan untuk secara eksplisit menentukan preferensi untuk barang tertentu, biasanya dengan menunjukkan tingkat penghargaan pada *likert scales* 5 poin atau 7 poin. kemudian dipetakan ke nilai numerik, misalnya pada rentang kontinu :

$$r_i(b_k) \in [-1, +1] \quad (10)$$

Keterangan :

Nilai negatif menunjukkan ketidaksukaan, sementara nilai positif mengekspresikan keinginan pengguna.

- b. *Implicit Ratings*, pada *implicit ratings* memaksakan upaya tambahan pada pengguna. Akibatnya, pengguna sering cenderung menghindari beban secara eksplisit.

Output dari sistem rekomendasi adalah prediksi atau rekomendasi, diantaranya :

- a. Prediksi (penilaian individu) dinyatakan sebagai nilai numerik, $p_{a,j}$, yang mewakili pendapat yang diantisipasi dari adanya penggunaan aktif u_a untuk *item* i_j .
- b. Rekomendasi dinyatakan sebagai daftar *item*, dimana $x < n$. Daftar ini hanya mencakup *item* yang belum dinilai oleh pengguna.

2.3 Clustering

Clustering adalah proses pengelompokan satu *set* objek sedemikian rupa sehingga objek dalam grup yang sama lebih mirip dalam beberapa cara tertentu satu sama lain daripada yang ada di kelompok lain (Ghuman S.S, 2016). Algoritma *clustering* biasanya digunakan untuk menemukan suatu kelompok dari pengguna atau *item* yang berhubungan. Algoritma ini secara khusus dikembangkan untuk data di mana *Euclidean*, atau pengukuran numerik lainnya, pengukuran jarak tidak dapat diterapkan. Terdapat dua kategori algoritma *clustering*, yaitu *hierarchical* dan *partitional*. Algoritma *hierarchical clustering* membuat dekomposisi (pemecahan) secara hirariki dari objek-objek. Dekomposisi bisa dilakukan secara *agglomerative (bottom-up)* atau *divisive (top-down)*. Sedangkan algoritma *partitional clustering* membuat sejumlah k partisi dari data yang dimana setiap kelompok mengoptimasikan acuan pengelompokkan. Teknik *partitional* yang biasanya digunakan adalah *k-means*, PAM (*Partitioning Around Medoids*), CLARA (*Clustering LARge Applications*), dan CLARANS (*Clustreing LARge Applications*) (Andritsos, 2002).

Berikut tabel perbandingan pendekatan yang dekat antara metode parsial dan hierarkis :

Tabel 2.3 Perbandingan pada teknik *clustering*

<i>Clustering Technique</i>	<i>Shape of Cluster</i>	<i>Clustering Algorithm</i>	<i>Outlier Handling</i>
<i>Hierarchical</i>	<i>Arbitrary</i>	<i>BIRCH, CURE</i>	<i>Yes</i>
<i>Partition</i>	<i>Spherical</i>	<i>K-means, K-mode</i>	<i>No</i>
<i>Density</i>	<i>Arbitrary</i>	<i>DBSCAN</i>	<i>Yes</i>
<i>Grid</i>	<i>Arbitrary</i>	<i>CLIQUE, Wave Cluster</i>	<i>Yes</i>

1. *K-Medoids Clustering*

Algoritma *k-Means* sensitif terhadap *outlier* karena objek dengan nilai yang sangat besar mungkin secara substansial mendistorsi distribusi data. Alih-alih mengambil nilai rata-rata objek dalam a kluster sebagai titik referensi, medoid dapat digunakan, yang merupakan objek paling terpusat terletak di sebuah *cluster*. Dengan demikian, metode partisi masih bisa dilakukan berdasarkan prinsip meminimalkan jumlah perbedaan antara setiap objek dan objeknya titik referensi yang sesuai (Velmurugan T., 2012). Ini membentuk dasar dari metode *k-medoids*. Strategi dasar algoritma pengelompokan *k-medoids* adalah menemukan k *cluster* di n keberatan dengan terlebih dahulu menemukan perwakilan secara sewenang-wenang objek (*medoid*) untuk setiap *cluster*. Masing-masing tersisa objek dikelompokkan dengan *medoid* yang itu paling mirip. Metode *k-medoid* menggunakan objek representatif sebagai titik referensi alih-alih mengambil nilai rata-rata objek di setiap cluster adalah titik kunci dari metode ini. Algoritma mengambil input parameter k , jumlah *cluster* menjadi dipartisi di antara seperangkat objek n . Algoritma *k-medoids* untuk partisi berdasarkan objek *medoid* atau pusat adalah sebagai berikut:

Input: k : Jumlah *cluster*

D: Kumpulan data yang berisi n objek

Output: Satu set k *cluster* yang meminimalkan jumlah ketidaksamaan semua objek dengan medoid terdekat.

Method: Pilih objek k dalam D sebagai objek representatif awal;

Ulangi Penetapan setiap objek yang tersisa ke cluster dengan medoid terdekat;

pilih secara acak objek non medoid O_{random} ;

Hitung total poin S dari objek *swap* O_j dengan O_{random} ;

jika $S < 0$ maka tukar O_j dengan O_{random} untuk membentuk set baru k medoid;

Sampai tidak ada perubahan;

Tentukan partisi k untuk n objek. Setelah pemilihan acak awal k *medoid*, the algoritma berulang kali mencoba membuat pilihan yang lebih baik *medoid*. Karena itu, algoritmanya sering disebut sebagai algoritma berbasis objek representatif.

2.4 Event

Event merupakan suatu kegiatan yang diselenggarakan untuk memperingati hal-hal penting sepanjang hidup manusia baik secara individu atau kelompok yang terkait secara adat, budaya, tradisi dan agama yang diselenggarakan untuk tujuan tertentu serta melibatkan lingkungan masyarakat yang diselenggarakan pada waktu tertentu (Any Noor, 2009).

Namun pada perkembangannya, *event* tumbuh dan berkembang menjadi sebuah kegiatan bisnis. Hal tersebut dilandasi dengan munculnya istilah *event organizer (EO)* sebagai jasa penyelenggara *event*. Pengertian yang hingga kini terus terbentuk dalam dunia *EO* bahwa ruang lingkup kegiatan yang digarap terbatas pada kegiatan berskala atau ukuran tertentu di luar segmen yang digarap oleh perjalanan insentif, konferensi dan pameran, dan berkaitan dengan kegiatan sosial budaya, perayaan-perayaan, hobi, hiburan, *sport* khusus dan lainnya, meskipun sebenarnya dunia *EO* bisa lebih luas lagi.

2.4.1 Kategori Event

Event terbagi ke dalam 4 kategori yaitu :

1. *Leisure Event*

Pada saat ini *leisure event* yang berkembang banyak berdasarkan pada kegiatan keolahragaan. Kegiatan yang sama memiliki unsur pertandingan di dalamnya dan mendatangkan banyak pengunjung pada *event* tersebut.

2. *Personal Event*

Personal event adalah segala bentuk kegiatan yang di dalamnya terlibat anggota keluarga atau teman.

3. *Cultural Event*

Budaya selalu identik dengan upacara adat, dan tradisi memiliki nilai sosial yang tinggi dalam tatanan masyarakat, sehingga penyelenggaraannya saat ini menjadi sangat penting.

4. *Organizational Event*

Merupakan kegiatan besar pada setiap organisasi. Pada kategori ini, bentuk *event* yang diselenggarakan tentunya disesuaikan dengan tujuan organisasi (Noor, 2009).

2.4.2 Event sebagai Media Promosi

Kutipan dari buku *Integrated Marketing Communication* (Sulaksana, 2007) menyebutkan “Meskipun *event* kontes-kontesan termasuk kegiatan promosi *below the line*, penyelenggaraannya kini tidak hanya dilakukan di lini bawah. Perusahaan semakin cerdas memanfaatkan *event* ini untuk membangun *awareness* dan *brand image*-nya di benak khalayak.” Dari kutipan tersebut dapat diketahui bahwa *event* dapat digunakan sebagai media untuk melakukan kegiatan promosi.

Event dikelompokkan ke dalam saluran komunikasi non-personal. Saluran *non-personal* meliputi media, atmosfer dan *event*. *Event* merupakan peristiwa-peristiwa yang dirancang untuk mengkomunikasikan pesan tertentu pada *audience*.

2.4.3 Pendekatan Event

Terdapat tiga pendekatan yang harus diperhatikan dalam menyelenggarakan sebuah *event*, yang di kenal dengan “3 E” (Leonard, 2002), yaitu:

1. Entertainment

Kunci keberhasilan pemasaran *event* adalah mampu menyediakan hiburan yang menarik audiens untuk keluar dari rumahnya dan mencoba sesuatu yang tidak bisa mereka peroleh dirumah.

2. Excitement

Hal ini adalah kunci agar *event* yang di selenggarakan selalu dikenang. Mengesankan dapat diciptakan dengan memberi penghormatan (*award*) sebagai perusahaan terbaik, pengenalan logo baru, atau perayaan ulang tahun perusahaan. Setiap *event* harus dirancang untuk memberi kesan mendalam, dan harus menjadi bagian dalam pemasaran. Jadi *event* apapun yang dipasarkan, buatlah agar “mengesankan”.

3. Enterprise

Enterprise diartikan sebagai kesiapan untuk menanggung resiko atau mencoba sesuatu yang belum pernah dicoba. *Event* yang menuntut pengembangan akal, berlayar tanpa diberi petunjuk arah, mengubah cara-cara yang biasa menjadi cara yang lebih imajinatif. Inilah kreativitas dan inovasi yang harus selalu ada dalam *event*.

2.4.4 Karakteristik *Event*

Event memiliki beberapa karakteristik karena setiap penyelenggaraan *event* harus memiliki ciri tersendiri. Bagaimanapun karakteristik *event* hampir sama dengan pelayanan yang diberikan oleh industri pelayanan lainnya (Noor, 2009).

Berikut adalah karakteristik yang mendasari dari evaluasi *event*:

1. Keunikan

Kunci utama suksesnya sebuah *event* adalah pengembangan ide, maka *event* yang diselenggarakan akan memiliki keunikannya tersendiri.

2. *Perishability*

Setiap *event* yang diselenggarakan tidak pernah sama, apabila *event* yang sama diselenggarakan lagi, akan tetapi *event* yang dihasilkan tidak akan sama persis seperti *event* sebelumnya.

3. *Intangibility*

Setelah menghadiri *event*, yang tertinggal di benak pengunjung adalah pengalaman yang telah di dapatkan dari penyelenggaraan *event*. Hal tersebut merupakan proses perubahan dari *tangible* menjadi *intangible*.

4. Suasana dan Pelayanan

Suasana dan pelayanan merupakan karakteristik yang penting pada saat keberlangsungan *event*. *Event* yang di laksanakan dengan suasana dan pelayanan yang tepat akan menciptakan *event* yang sukses.

5. Interaksi *personal*

Interaksi *personal* dari pengunjung merupakan kunci sukses penyelenggaraan *event*, karena pengunjung akan merasa menjadi bagian dalam *event* tersebut.

2.5 Hobi

Menurut kamus besar bahasa Indonesia, hobi adalah suatu kesenangan istimewa yang dilakukan pada waktu senggang dan ditekuni oleh seseorang. Ruang hobi terbagi menjadi berbagai macam, yaitu hobi diluar (*outdoor*) dan didalam (*indoor*) dan hobi yang berhubungan dengan

aktivitas fisik atau tanpa aktivitas fisik. Aktivitas hobi memerlukan keutamaan atau preferensi dalam fungsionalitas ruang. Diperlukan pendapat menurut masing-masing individu dengan hobi yang paling sering mereka lakukan. Pendapat dari masing-masing individu mengungkapkan tingkat kepentingan dalam pemenuhan fungsionalitas ruang.

Empat variabel laten dimunculkan untuk mengetahui preferensi tersebut. Profil responden dengan hobinya dimunculkan melalui analisis distribusi. Sedangkan variabel laten preferensi ruang dan keterkaitannya dengan jenis hobi, diketahui melalui analisis faktor. Kecenderungan jenis hobi yang bersifat relaksasi memiliki preferensi privasi dan akustik yang baik. Sedangkan kecenderungan jenis hobi yang bersifat *refreshing* memiliki preferensi ruang dengan akses sosial dan kualitas *visual* yang baik.

Pendapat ruang hobi ideal berbeda pada masing-masing individu. Jenis hobi yang dijalankan juga mempengaruhi pendapat ideal. Selain itu faktor ruang dimana melakukan hobi juga ikut menentukan pendapat ini. Oleh karena itu diperlukan peninjauan pendapat tentang bagaimana ruang hobi yang ideal. Penelitian ini ditujukan sebagai preferensi perancangan ruang arsitektur terhadap aktivitas hobi (Nurhariyadi, 2016).

1.6 Metode Waterfall

Model Waterfall mendefinisikan beberapa berturut-turut fase yang harus diselesaikan satu demi satu dan pindah ke fase berikutnya hanya ketika fase sebelumnya adalah sepenuhnya selesai. Untuk alasan ini, model Waterfall adalah rekursif dalam setiap fase yang dapat diulang tanpa henti sampai itu disempurnakan (Bassil, Y., 2012).

Secara umum, proses simulasi yang diusulkan terdiri dari langkah-langkah berikut:

1. Jalankan simulasi, periksa data yang dihasilkan oleh simulasi.
2. Temukan perubahan yang akan dilakukan pada model berdasarkan pada analisis data yang dihasilkan oleh simulasi.
3. Ulangi sebanyak yang diperlukan untuk mencapai yang optimal hasil.

Secara teknis, proses simulasi metode *waterfall* terdiri dari langkah-langkah berikut:

1. Bagi model Air Terjun ke dalam fase independen,
2. Memahami konsep dan persyaratan yang ada di balik setiap fase.
3. Tentukan sumber daya, tugas, entitas, dan pekerjaan aliran setiap fase.

4. Simulasikan setiap fase terpisah dan catat hasil,
5. Mengintegrasikan seluruh fase bersama, mensimulasikan sistem, dan catat hasil.

1.7 Pengujian Pada Perangkat Lunak dan Pengujian Keakuratan Hasil Rekomendasi

1. Pengujian sistem pada perangkat lunak dengan *blackbox testing* berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. *Black box testing* cenderung untuk menemukan hal-hal seperti, fungsi yang tidak benar atau tidak ada, kesalahan antarmuka (*interface errors*), kesalahan pada struktur data dan akses basis data, kesalahan performa (*performance errors*), kesalahan inisialisasi dan terminasi (Mustaqbal et al, 2015).

Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut:

- a. Bagaimana fungsi-fungsi diuji agar dapat dinyatakan valid?
- b. Input seperti apa yang dapat menjadi bahan kasus uji yang baik?
- c. Apakah sistem sensitif pada input-input tertentu?
- d. Bagaimana sekumpulan data dapat diisolasi?
- e. Berapa banyak rata-rata data dan jumlah data yang dapat ditangani sistem?
- f. Efek apa yang dapat membuat kombinasi data ditangani spesifik pada operasi sistem?

Saat ini terdapat banyak metoda atau teknik untuk melaksanakan *Black Box Testing*, diantaranya *Equivalence Partitioning*, *Boundary Value Analysis/Limit Testing*, *Comparison Testing*, *Sample Testing*, *Robustness Testing*, *Behavior Testing*, *Requirement Testing*, *Performance Testing*, Uji Ketahanan (*Endurance Testing*), Uji Sebab-Akibat (*Cause-Effect Relationship Testing*).

2. Pengujian Keakuratan Hasil Rekomendasi

Pengujian digunakan untuk mengetahui seberapa akurat hasil rekomendasi yang diberikan sistem kepada *user*. Beberapa cara yang dapat dilakukan untuk mengevaluasi sistem rekomendasi yaitu dengan menghitung tingkat error pada hasil rekomendasi dan juga menghitung nilai presisi dari hasil *User Acceptance Test (UAT)* yang diberikan *user*.

Dalam sistem rekomendasi, hasil rekomendasi yang diberikan perlu dilakukan pengukuran untuk mengukur tingkat akurasi hasil rekomendasi dapat dilakukan dengan melihat nilai *error* pada hasil rekomendasi yang diberikan. Terdapat banyak metode perhitungan *error* misalnya saja seperti salah satu perhitungan sebagai contoh yaitu, *Mean Absolute Error* (MAE) (Kim et al., 2016).

Berikut perhitungan Mean Absolute Error (MAE) seperti yang disajikan di bawah ini. (Archana, K., et al., 2017).

$$\text{MAE} = \frac{\sum_t^n |Y_t - \hat{Y}_t|}{n} \quad (11)$$

Dimana,

Y_t = Nilai Aktual

\hat{Y}_t = Nilai Prediksi

n = Jumlah Konten

Semakin rendah nilai MAE yang didapat, maka sistem semakin akurat dalam memprediksi.

UNIVERSITAS MIKROSKIL