

# Sistem Rekomendasi Konsentrasi Berdasarkan Aggregate Function Multi Criteria Pada Prodi Informatika UK Petra

Willy Santoso, Rolly Intan, Yulia

Program Studi Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) - 8417658

E-mail: willy\_santoso21@hotmail.com, rintan@petra.ac.id, yulia@petra.ac.id

## ABSTRAK

Sistem rekomendasi merupakan sistem yang berfungsi untuk membantu *user* dalam mengambil keputusan dengan cara memberikan rekomendasi *item* kepada *user*. Dalam perkuliahan, mahasiswa dihadapkan pada pilihan untuk mengambil konsentrasi studi tertentu. Namun, seringkali mahasiswa kebingungan untuk memilih konsentrasi studi yang harus diambil. Oleh karena itu, diperlukan suatu sistem yang dapat memberikan rekomendasi konsentrasi sehingga dapat membantu mahasiswa dalam pengambilan keputusan akan konsentrasi studi yang diambil.

Sebelumnya telah ada skripsi untuk menyelesaikan masalah serupa yaitu dengan menggunakan metode *User Based Collaborative Filtering*. Pada penelitian ini menggunakan metode *Aggregate Function* untuk menggabungkan hasil dari ketiga kriteria. Implementasi *Aggregate Function* dilakukan dengan cara menghitung nilai rata-rata setiap konsentrasi dari hasil setiap kriteria. Hasil kriteria pertama didapat dengan menggunakan metode *User Based Collaborative Filtering* berbasis algoritma *Adjusted Cosine Similarity*. Hasil kriteria kedua didapat dengan menghitung nilai rata-rata mata kuliah wajib tertentu yang telah diambil oleh *user*. Hasil kriteria ketiga didapat dengan menghitung nilai rata-rata kesukaan *user* pada suatu bidang profesi.

Pengujian sistem rekomendasi dilakukan dengan menggunakan *confusion matrix* dengan mengukur nilai akurasi sistem rekomendasi. Hasil dari penelitian ini adalah urutan rekomendasi konsentrasi yang sesuai untuk *user* berdasarkan penggunaan *Aggregate Function*. Hasil dari pengujian yang dilakukan terhadap 30 *user* tingkat akurasi yang didapatkan mencapai 73.3%.

**Kata Kunci:** Sistem Rekomendasi, *User Based Collaborative Filtering*, *Adjusted Cosine Similarity*, *Aggregate Function*.

## ABSTRACT

*The Recommendation system is a system that uses to help users in making decisions by providing item recommendations to the user. In lectures, students are faced with the choice to take a certain concentration of study. However, students are often confused about which concentration of study should they take. Therefore, a system that can provide concentration recommendations is needed so that it can help students in making decisions on choosing the best concentration.*

*Previously there had been a thesis to solve a similar problem using User Based Collaborative Filtering method. In this research the method that will be used is Aggregate Function. Aggregate Function is used to combine the results of the three criteria. The implementation of the Aggregate Function is done by calculating the average value of each concentration from the results of each*

*criterion. The results of the first criteria are obtained using the User Based Collaborative Filtering method based on the Adjusted Cosine Similarity Algorithm. The results of the second criteria are obtained by calculating the average value of the certain mandatory courses that have been taken by the user. The results of the third criterion are obtained by calculating the average value of user preferences in a professional field.*

*Testing of the recommendation system is done by using confusion matrix by measuring the value of accuracy of the recommendation system. The result of this research is the order of concentration recommendations that are suitable for the user based on the use of the Aggregate Function. The results of the tests that used on 30 users obtained 73.3% accuracy rate.*

**Keywords:** *Recommender System, User Based Collaborative Filtering, Adjusted Cosine Similarity, Aggregate Function.*

## 1. PENDAHULUAN

Informatika merupakan salah satu program studi (prodi) yang ada di Universitas Kristen Petra. Prodi ini akan membekali mahasiswa dengan berbagai ilmu yang diperlukan dalam bidang IT (*Information Technology*). Selama proses pendidikan di prodi Informatika, setiap mahasiswa diwajibkan untuk menyelesaikan berbagai macam mata kuliah wajib sesuai dengan kurikulum yang digunakan. Mahasiswa juga akan diminta untuk memilih suatu konsentrasi dari beberapa macam konsentrasi yang telah disediakan oleh prodi Informatika untuk menjadi fokus pembelajaran yang diminati.

Pada saat telah memasuki semester pengambilan konsentrasi, mahasiswa seringkali merasa kesulitan dan kebingungan untuk memilih konsentrasi apa yang cocok dengan dirinya [1]. Permasalahan di atas menjadi dasar diperlukannya sebuah sistem rekomendasi. Sistem rekomendasi adalah sistem yang dapat membantu pengguna dalam mengambil keputusan berdasarkan preferensinya. Sistem rekomendasi dapat membantu mahasiswa dalam mengambil keputusan sehingga mahasiswa dapat memilih konsentrasi yang tepat.

Pada penelitian ini, sistem rekomendasi akan dibuat menggunakan metode *Aggregate Function Multi Criteria*. Terdapat 3 kriteria yang akan digunakan dalam pembuatan sistem rekomendasi. Kriteria pertama yaitu mendapatkan hasil rekomendasi berdasarkan penggunaan metode *User Based Collaborative Filtering* berbasis algoritma *Adjusted Cosine Similarity*. Kriteria kedua yaitu mendapatkan hasil rekomendasi berdasarkan nilai mata kuliah wajib tertentu. Untuk menentukan nilai mata kuliah wajib yang digunakan didapatkan dengan cara melakukan wawancara dengan kepala bidang (kabid) setiap konsentrasi. Kriteria ketiga yaitu

mendapatkan hasil rekomendasi berdasarkan nilai kesukaan terhadap bidang profesi tertentu. Untuk mendapatkan nilai kesukaan *user* terhadap bidang profesi tertentu dilakukan dengan cara meminta *user* untuk mengisi sebuah survei tentang penilaian kesukaan *user* terhadap bidang profesi yang ada.

## 2. LANDASAN TEORI

### 2.1 Sistem Rekomendasi

Sistem rekomendasi adalah sebuah perangkat lunak yang dirancang dengan tujuan untuk membantu pengguna dalam mencari sesuatu yang mungkin disukai dengan cara memberikan rekomendasi kepada penggunaannya [5].

### 2.2 Collaborative Filtering

*Collaborative filtering* merupakan salah satu metode yang sering digunakan dalam pembuatan sistem rekomendasi. Ide utama penggunaan *collaborative filtering* dalam sistem rekomendasi adalah memanfaatkan riwayat pengguna aktif lain untuk memprediksi konsentrasi yang mungkin akan disukai/diminati oleh seorang pengguna [5].

#### 2.2.1 Proses collaborative filtering

Ide dasar dari menghitung *similarity* antara mahasiswa  $U_i$  dengan  $U_m$  adalah dengan membandingkan data semua nilai mata kuliah dari mahasiswa  $U_i$  dengan  $U_m$  untuk menghasilkan nilai prediksi pada mata kuliah konsentrasi yang belum pernah diambil oleh mahasiswa  $U_m$ . Cara penerapan metode ini yaitu dengan mencari semua *user* aktif yang memiliki nilai *Top-K similarity* paling tinggi dengan mahasiswa  $U_m$ , setelah itu maka dilakukan perhitungan prediksi pada mata kuliah konsentrasi yang belum pernah diambil oleh *user*  $U_m$ . Berdasarkan pada hasil nilai prediksi ini, maka akan ditentukan konsentrasi terbaik yang akan direkomendasikan kepada mahasiswa [1].

#### 2.2.2 User based collaborative filtering

*User Based Collaborative Filtering* merupakan suatu metode yang menghasilkan rekomendasi *item* dengan membandingkan semua *item* pada semua *user* aktif dengan *user* tertentu untuk mencari *Top-K similarity user* aktif dengan *user* tertentu [1].

Kelebihan dari *User Based Collaborative Filtering* [7]:

- Algoritma dapat digunakan di banyak macam kondisi.
- Sangat mudah dalam melakukan *update* pada *database* karena algoritma ini prediksi selalu dilakukan berdasarkan pada semua *user* pada *database*.
- Lebih cocok dipergunakan pada karakteristik data dimana perbedaan jumlah *item* yang dimiliki oleh *user* yang satu dengan lainnya kecil.

Kelemahan dari *User Based Collaborative Filtering* [7]:

- Membutuhkan waktu yang lama, hal ini dikarenakan setiap terjadi penambahan data *user* baru, akan dilakukan *update* pada nilai prediksi.

#### 2.2.3 Algoritma adjusted cosine similarity

*Adjusted Cosine Similarity* merupakan suatu algoritma pengembangan dari *Cosine Similarity* dimana akan terdapat pertimbangan terhadap data / nilai yang 0, sehingga perlu dilakukan normalisasi pada data untuk menjadikan data nilai kosong yang diibaratkan 0 menjadi sebuah nilai tengah. Dalam studi kasus ini tentunya terdapat mata kuliah yang belum diambil oleh mahasiswa.

Algoritma ini dipilih karena dengan menggunakan algoritma ini maka nilai 0 yang mewakili nilai nilai dari mata kuliah yang belum diambil oleh mahasiswa tidak lagi menjadi nilai terendah, namun menjadi nilai tengah dari perhitungan *similarity* sehingga akan menghasilkan hasil yang lebih akurat [1]. Rumus dari *Adjusted Cosine Similarity* adalah sebagai berikut [2].

$$sim(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_j)^2}} \quad (1)$$

Dimana:

- $i$  merupakan mahasiswa yang nilainya akan diprediksi.
- $j$  merupakan mahasiswa yang datanya digunakan untuk memprediksi.
- $sim(i, j)$  merupakan nilai *similarity* antara  $i$  dan  $j$ .
- $R_{u,i}$  merupakan nilai mata kuliah  $u$  dari mahasiswa  $i$ .
- $R_{u,j}$  merupakan nilai mata kuliah  $u$  dari mahasiswa  $j$ .
- $\bar{R}_j$  merupakan nilai rata - rata dari semua mata kuliah dari mahasiswa  $j$ .
- $\bar{R}_i$  merupakan nilai rata - rata dari semua mata kuliah dari mahasiswa  $i$ .

#### 2.2.4 Nilai prediksi

Setelah dilakukan perhitungan *similarity* untuk mencari *Top-K* mahasiswa yang memiliki *similarity* tertinggi dengan mahasiswa  $U_m$  maka langkah yang dilakukan adalah menghitung nilai prediksi pada mata kuliah konsentrasi yang belum diambil oleh mahasiswa  $U_m$  berdasarkan pada data nilai mata kuliah dari semua *Top-K* mahasiswa [9].

Rumus dari nilai prediksi dapat dinyatakan sebagai berikut [3]:

$$P_{u,i} = \min(4, \bar{R}_a + \frac{\sum_{j \in U} (R_{u,j} - \bar{R}_j) * S_{i,j}}{\sum_{j \in U} |S_{i,j}|}) \quad (2)$$

Dimana:

- $i$  merupakan mahasiswa yang nilainya akan diprediksi.
- $j$  merupakan mahasiswa yang datanya digunakan untuk memprediksi.
- $P_{u,i}$  adalah nilai prediksi mata kuliah yang dari mahasiswa  $i$  pada mata kuliah  $u$ .
- $j \in U$  merupakan *Top-K* mahasiswa yang memiliki *similarity* paling tinggi dengan mahasiswa.
- $\bar{R}_a$  merupakan rata-rata dari semua mata kuliah mahasiswa yang diprediksi.
- $R_{u,j}$  merupakan nilai mata kuliah mahasiswa  $j$  terhadap mata kuliah  $u$ .
- $\bar{R}_j$  merupakan nilai rata-rata dari semua mata kuliah dari mahasiswa  $j$ .
- $S_{i,j}$  adalah nilai *similarity* antara mahasiswa  $i$  dan  $j$ .

## 2.3 Aggregate Function

Fungsi agregasi merupakan suatu perhitungan matematis yang melibatkan seperangkat nilai dan bukan nilai tunggal. Fungsi agregasi sering digunakan di *database* dan *spreadsheet*, dan mencakup rata-rata atau jumlah dari serangkaian angka. Perhitungan yang dilakukan oleh fungsi ini mengembalikan nilai tunggal dari beberapa nilai. Terdapat berbagai fungsi yang termasuk dalam fungsi agregasi, misalnya *average (mean)*, *sum*, dan banyak lainnya [8].

### 2.3.1 Average

*Average*, atau yang lebih dikenal dengan *mean*, merupakan suatu kumpulan nilai *metrics* tunggal yang menghitung rata-rata nilai numerik yang diambil dari penggabungan data. Nilai ini dapat dihasilkan dari kolom numerik tertentu dalam suatu dokumen, atau dihasilkan melalui teks yang disediakan [8].

Rumus yang digunakan untuk menghitung nilai *average* ialah:

$$\bar{X} = \frac{\sum X_i}{n} \quad (3)$$

Dimana:

- $\bar{X}$  merupakan nilai rata-rata (*average*)
- $\sum X_i$  merupakan jumlah dari seluruh data
- $n$  merupakan banyaknya data

## 2.4 Confusion Matrix

*Confusion Matrix* merupakan tabel yang menggambarkan performa dari sebuah model atau algoritma secara spesifik. Setiap baris dari matrix tersebut, merepresentasikan kelas aktual dari data, dan setiap kolom merepresentasikan kelas prediksi dari data (atau sebaliknya) [6].

*Confusion Matrix* sering digunakan untuk mengevaluasi jalannya program. Salah satu bentuk evaluasi yang dapat dilakukan yaitu mengukur akurasi dari program. Akurasi merupakan rasio prediksi benar (positif dan negatif) dengan keseluruhan data.

Rumus yang digunakan untuk menghitung nilai akurasi ialah [4]:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (4)$$

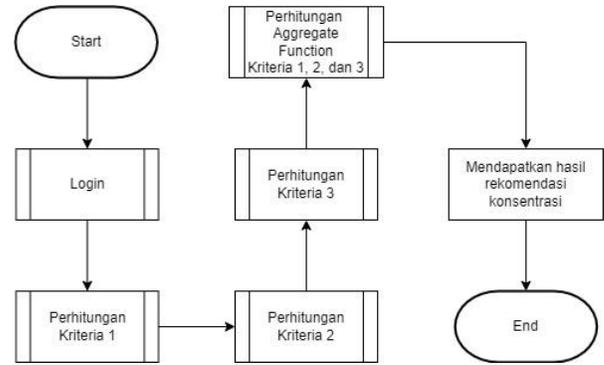
Keterangan:

- *True Positif*: Berarti seberapa banyak data yang aktual kelasnya positif, dan model juga memprediksi positif.
- *True Negatif*: Berarti seberapa banyak data yang aktual kelasnya negatif, dan model memprediksi negatif.
- *False Positif*: Berarti seberapa banyak data yang aktual kelasnya negatif, namun model memprediksi positif.
- *False Negatif*: Berarti seberapa banyak data yang aktual kelasnya positif, namun model memprediksi negatif.

## 3. DESAIN SISTEM

### 3.1 Desain Flowchart

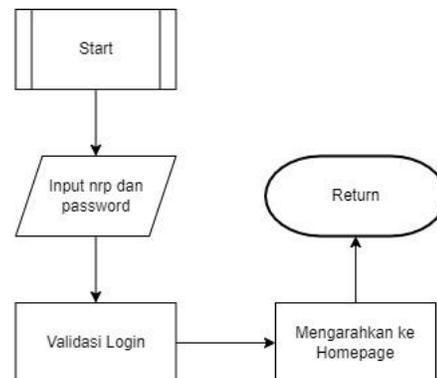
*Flowchart* digunakan untuk menggambarkan proses yang terjadi pada suatu sistem yang dibuat. Dalam *flowchart* akan disajikan gambaran secara terperinci mengenai *input*, proses, dan *output* yang dihasilkan dari sistem. Proses jalannya sistem rekomendasi dimulai dari *user* melakukan *login* lalu dilanjutkan perhitungan kriteria 1, 2, dan 3. Selanjutnya melakukan perhitungan *aggregate function*, pada proses perhitungan *aggregate function* yang akan digunakan yaitu fungsi *average* yang berfungsi untuk menghitung nilai rata-rata dari ketiga kriteria. Setelah melalui proses perhitungan nilai rata-rata, proses akan dilanjutkan dengan mengambil hasil urutan rekomendasi yang sesuai berdasarkan nilai perhitungan rata-rata dari yang tertinggi ke yang terendah. Tampilan *flowchart* sistem secara keseluruhan dapat dilihat pada Gambar 1.



Gambar 1. Flowchart alur sistem rekomendasi konsentrasi

### 3.1.1 Flowchart login

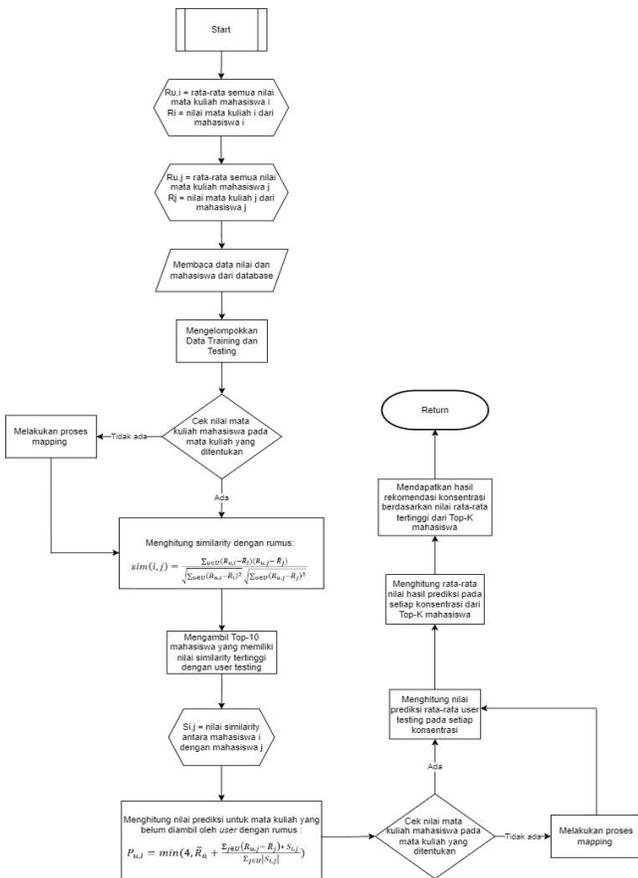
Flowchart *login* menyajikan gambaran secara detail mengenai *input*, proses, dan *output* untuk *login*. Proses *login* dimulai dari *user* memasukkan *nrp* dan juga *password*. Selanjutnya sistem akan melakukan validasi data *user* dengan data yang ada di *database*. Selanjutnya sistem akan mengarahkan *user* ke *homepage user* mahasiswa jika yang *login* adalah mahasiswa, dan mengarahkan ke *homepage admin* apabila *user* yang *login* adalah *admin*. Tampilan flowchart *login* dapat dilihat pada Gambar 2.



Gambar 2. Flowchart login

### 3.1.2 Flowchart kriteria 1

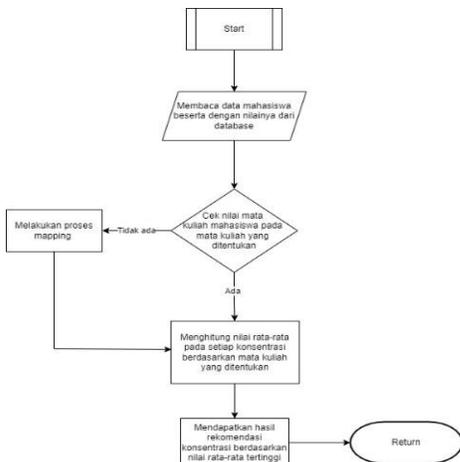
*Flowchart* kriteria 1 menyajikan gambaran secara detail mengenai *input*, proses, dan *output* dari kriteria 1. Proses pada kriteria 1 dimulai dengan membaca data mahasiswa beserta dengan nilainya dari *database*. Setelah itu sistem akan mengelompokkan data *training* dan data *testing*. Selanjutnya akan dilakukan proses *mapping*. Proses selanjutnya yaitu *perhitungan similarity* antara *user* dengan *user* lain. Setelah didapatkan *list* mahasiswa lain yang *similar* dengan *user* maka diambil 10 *user* lain dengan nilai *similarity* tertinggi, setelah itu dilakukan perhitungan nilai prediksi pada mata kuliah yang belum diambil oleh *user*. Selanjutnya sistem akan melakukan perhitungan nilai rata-rata setiap bidang konsentrasi untuk menentukan konsentrasi apa yang paling sesuai dengan *user*. Tampilan *flowchart* kriteria 1 dapat dilihat pada Gambar 3.



Gambar 3. Flowchart kriteria 1

### 3.1.3 Flowchart kriteria 2

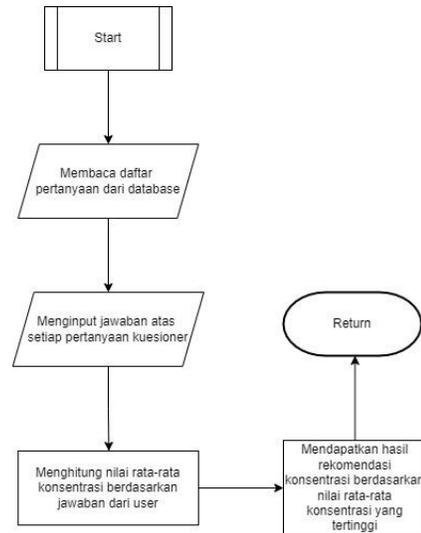
Flowchart kriteria 2 menyajikan gambaran secara detail mengenai input, proses, dan output dari kriteria 2. Proses pada kriteria 2 dimulai dengan membaca data mahasiswa beserta dengan nilainya dari database. Selanjutnya sistem mengecek apakah nilai mata kuliah pada mata kuliah yang ditentukan ada atau tidak. Jika tidak maka sistem akan melakukan proses mapping mata kuliah. Lalu sistem akan menghitung nilai rata-rata mata kuliah pada setiap bidang konsentrasi. Terakhir sistem akan merekomendasikan konsentrasi berdasarkan nilai rata-rata konsentrasi yang tertinggi. Tampilan flowchart kriteria 2 dapat dilihat pada Gambar 4.



Gambar 4. Flowchart kriteria 2

### 3.1.4 Flowchart kriteria 3

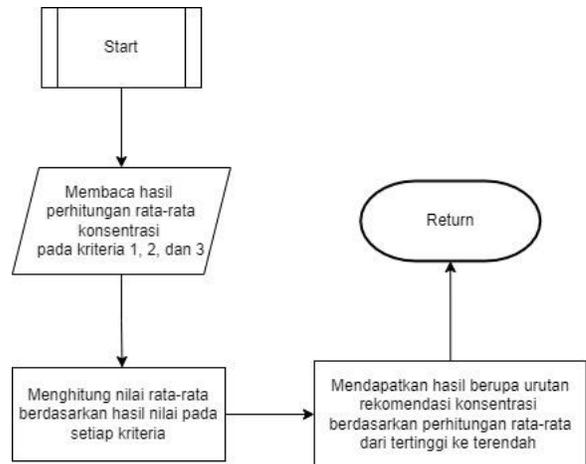
Flowchart kriteria 3 menyajikan gambaran secara detail mengenai input, proses, dan output dari kriteria 3. Proses yang dilakukan di kriteria 3 dimulai dari sistem membaca daftar pertanyaan dari database, daftar pertanyaan yang akan digunakan dapat ditambah atau dihapus oleh admin. Selanjutnya user diminta untuk mengisi jawaban atas kuesioner yang telah diberikan, setelah itu sistem akan menghitung nilai rata-rata setiap bidang konsentrasi berdasarkan jawaban dari user. Selanjutnya sistem akan menghasilkan rekomendasi konsentrasi berdasarkan nilai rata-rata konsentrasi tertinggi. Tampilan flowchart kriteria 3 dapat dilihat pada Gambar 5.



Gambar 5. Flowchart kriteria 3

### 3.1.5 Flowchart aggregate function

Flowchart aggregate function menyajikan gambaran secara detail mengenai input, proses, dan output dari aggregate function. Proses aggregate function dimulai dari membaca seluruh data hasil perhitungan dari kriteria 1, 2, dan 3. Selanjutnya sistem akan melakukan perhitungan rata-rata setiap bidang konsentrasi berdasarkan hasil pada setiap kriteria. Selanjutnya sistem akan menghasilkan urutan rekomendasi konsentrasi dari nilai perhitungan rata-rata tertinggi ke terendah. Tampilan flowchart aggregate function dapat dilihat pada Gambar 6.



Gambar 6. Flowchart aggregate function

## 4. PENGUJIAN SISTEM

### 4.1 Pengujian Aplikasi

Pengujian aplikasi akan menggunakan data dari *database* UK Petra. Data yang digunakan untuk *training* merupakan data mahasiswa angkatan 2012-2016, sedangkan data yang digunakan untuk testing merupakan data mahasiswa angkatan 2017.

### 4.2 Hasil Pengujian

Setelah pembuatan aplikasi selesai, selanjutnya dilakukan pengujian pada skripsi ini melalui survei yang diambil dari 30 orang yang telah mengambil konsentrasi. Hasil dari pengujian ini akan berupa urutan rekomendasi konsentrasi dengan menerapkan metode *Aggregate Function Multi Criteria* dan juga tingkat akurasi dari sistem yang dibuat.

### 4.3 Cara Pengujian

Pengujian ini dilakukan menggunakan data survei dari *user*. Survei yang diisi berisikan konsentrasi *real* yang diambil oleh user dan juga jawaban atas pertanyaan dari kuesioner pada kriteria 3. Dari hasil survei ini akan digunakan untuk mendapatkan hasil rekomendasi konsentrasi dari sistem. Proses pengujian sistem dimulai dari kriteria 1, kriteria 2, kriteria 3 lalu *aggregate function* 3 kriteria. Setelah itu akan dilakukan pengujian dengan cara mengukur tingkat akurasi menggunakan *confusion matrix*.

#### 4.3.1 Pengujian kriteria 1

Pengujian aplikasi diawali dengan perhitungan kriteria 1. Perhitungan kriteria 1 dilakukan dengan menggunakan metode *User Based Collaborative Filtering* berbasis Algoritma *Adjusted Cosine Similarity*. Data yang digunakan pada kriteria 1 yaitu data mata kuliah beserta nilainya dari mahasiswa nrp 325281161 dan mahasiswa *training*. Langkah pertama sistem akan mencari *top-k similar user* menggunakan *Adjusted Cosine Similarity*, *top-k similar user* berjumlah 10 *user*. Selanjutnya Sistem akan melakukan prediksi terhadap nilai mata kuliah konsentrasi yang telah diambil oleh *top-k similar user* dan belum pernah diambil oleh *user*. Selanjutnya sistem akan menghitung rata-rata nilai pada setiap bagian konsentrasi tiap *user*. Proses selanjutnya adalah sistem akan menghitung rata-rata setiap konsentrasi dari hasil prediksi *top-k similar user*.

Hasil perhitungan nilai rata-rata pada kriteria 1 untuk bidang konsentrasi *Artificial Intelligence* sebesar 3,26, *Game Development* sebesar 3,41, *Cyber Security* sebesar 3,47, *Mobile Application Development* sebesar 3,38 dan *Data Analytics* sebesar 3,47. Nilai ini didapat dari rata-rata nilai prediksi pada setiap konsentrasi berdasarkan *top-k similar user*. Setelah didapatkan nilai rata-rata pada setiap konsentrasi dapat ditentukan rekomendasi konsentrasi yang paling sesuai dengan *user* berdasarkan perhitungan kriteria 1 yaitu konsentrasi *Data Analytics*, karena nilai rata-rata konsentrasi *Data Analytics* merupakan nilai rata-rata tertinggi.

#### 4.3.2 Pengujian kriteria 2

Setelah melalui proses pengujian kriteria 1, selanjutnya akan dilakukan pengujian pada kriteria 2. Data yang digunakan untuk pengujian kriteria 2 yaitu data nilai mata kuliah wajib tertentu dari *user*, penentuan mata kuliah wajib yang digunakan didapatkan dari kepala bidang (kabid) setiap konsentrasi. Pengujian kriteria 2 dilakukan dengan cara menghitung nilai rata-rata pada setiap bidang konsentrasi.

Hasil perhitungan nilai rata-rata pada kriteria 2 untuk bidang konsentrasi *Artificial Intelligence* sebesar 3,50, *Game Development* sebesar 3,42, *Cyber Security* sebesar 3,75, *Mobile Application Development* sebesar 3,42 dan *Data Analytics* sebesar 3,50. Untuk mendapatkan nilai rata-rata setiap bidang konsentrasi dilakukan dengan cara menghitung *average* dari nilai mata kuliah yang termasuk pada tiap bidang konsentrasi. Setelah didapatkan nilai rata-rata pada setiap konsentrasi dapat ditentukan rekomendasi konsentrasi yang cocok untuk *user* berdasarkan perhitungan kriteria 2 yaitu konsentrasi *Cyber Security*, karena nilai rata-rata konsentrasi *Cyber Security* merupakan nilai rata-rata tertinggi.

#### 4.3.3 Pengujian kriteria 3

Setelah melalui proses pengujian kriteria 2, selanjutnya akan dilakukan pengujian pada kriteria 3. Data yang digunakan untuk pengujian pada kriteria 3 yaitu data hasil jawaban survei yang telah diisi oleh *user*. Pengujian kriteria 3 dilakukan dengan cara menghitung nilai kesukaan rata-rata pada setiap bidang konsentrasi. Setiap bidang konsentrasi berisi beberapa bidang profesi yang dapat dijalani setelah lulus dari jenjang perkuliahan. *User* diminta untuk memberikan nilai kesukaan pada setiap bidang konsentrasi yang ada. Setelah *user* memberikan nilai kesukaan dapat dilakukan perhitungan nilai kesukaan rata-rata pada setiap bidang konsentrasi. Aturan nilai kesukaan adalah sebagai berikut:

- Tidak suka mendapat nilai 0
- Sedikit suka mendapat nilai 1
- Cukup Suka mendapat nilai 2
- Suka mendapat nilai 3
- Sangat Suka mendapat nilai 4

Hasil perhitungan nilai rata-rata pada kriteria 3 untuk bidang konsentrasi *Artificial Intelligence* sebesar 2,17, *Game Development* sebesar 0,75, *Cyber Security* sebesar 0,40, *Mobile Application Development* sebesar 2,50 dan *Data Analytics* sebesar 1,80. Untuk mendapatkan nilai kesukaan rata-rata setiap bidang konsentrasi dilakukan dengan cara menghitung *average* dari nilai kesukaan pada bidang profesi yang termasuk dalam bidang konsentrasi. Setelah didapatkan nilai kesukaan rata-rata pada setiap konsentrasi dapat ditentukan rekomendasi konsentrasi yang paling sesuai untuk *user* berdasarkan perhitungan kriteria 3 adalah konsentrasi *Mobile Application Development*, karena *Mobile Application Development* memiliki nilai rata-rata tertinggi.

#### 4.3.4 Pengujian aggregate function multi criteria

Setelah didapatkan hasil perhitungan dari setiap kriteria, Langkah selanjutnya adalah mengimplementasikan *Aggregate Function* untuk mendapatkan urutan rekomendasi konsentrasi yang cocok untuk *user*. *Aggregate Function* yang digunakan yaitu *average*. Proses pengimplementasian *Aggregate Function* dengan cara menghitung *average* nilai setiap bidang konsentrasi berdasarkan hasil dari setiap kriteria.

Hasil perhitungan nilai rata-rata menggunakan *Aggregate Function* untuk bidang konsentrasi *Artificial Intelligence* sebesar 2,98, *Game Development* sebesar 2,52, *Cyber Security* sebesar 2,54, *Mobile Application Development* sebesar 3,10 dan *Data Analytics* sebesar 2,92. Untuk mendapatkan nilai rata-rata setiap bidang konsentrasi dapat dilakukan dengan cara menghitung *average* dari hasil kriteria 1, kriteria 2 dan kriteria 3. Hasil perhitungan kriteria 1 dan kriteria 2 seharusnya tidak boleh berbeda jauh karena sama-sama menggunakan data nilai mata kuliah. Namun terdapat faktor yang dapat mempengaruhi hasil dari kriteria 2 seperti informasi yang diberikan oleh kepala bidang (kabid) konsentrasi. Dari perhitungan

diatas dapat disimpulkan bahwa hasil perhitungan kriteria 1 dan kriteria 2 tidak jauh berbeda. Hasil kriteria 3 berbeda dengan kriteria 1 dan 2 karena pengukuran dilakukan dengan nilai kesukaan bukan dengan nilai mata kuliah. Setelah dilakukan perhitungan menggunakan *Aggregate Function* didapatkan hasil urutan rekomendasi konsentrasi yang paling sesuai untuk user adalah *Mobile Application Development*, *Artificial Intelligence*, *Data Analytics*, *Cyber Security* dan yang terakhir *Game Development*.

#### 4.3.5 Pengujian tingkat akurasi metode

Pengukuran tingkat akurasi metode dilakukan dengan menggunakan rumus akurasi *confusion matrix*. Data yang diperlukan adalah data hasil survei yang berisi konsentrasi real dari *user* dan data hasil prediksi konsentrasi yang didapat dari penerapan metode *Aggregate Function*. Setelah didapatkan data hasil survei dan data hasil prediksi dari sistem maka dapat ditentukan nilai TP, TN, FP, dan FN. Nilai TP dan TN akan bernilai 1 apabila konsentrasi real dan konsentrasi hasil prediksi sama, dan bernilai 0 apabila konsentrasi real dan konsentrasi hasil prediksi tidak sama. Sedangkan nilai FP dan FN akan bernilai 0 apabila konsentrasi real dan konsentrasi hasil prediksi sama, dan bernilai 1 apabila konsentrasi real dan konsentrasi hasil prediksi tidak sama. Nilai TP, TN, FP, dan FN inilah yang akan digunakan untuk mengukur akurasi. Pengujian ini dilakukan terhadap 30 *user* dari angkatan 2017 yang telah mengambil suatu konsentrasi. Hasil dari survei terhadap 30 *user* dinyatakan dalam bentuk tabel *confusion matrix* yang dapat dilihat pada Tabel 1.

**Tabel 1. Confusion Matrix**

| NRP       | Konsentrasi Real               | Konsentrasi Prediksi           | TP | TN | FP | FN |
|-----------|--------------------------------|--------------------------------|----|----|----|----|
| 325281121 | Mobile Application Development | Mobile Application Development | 1  | 1  | 0  | 0  |
| 325281137 | Game Development               | Cyber Security                 | 0  | 0  | 1  | 1  |
| 325281143 | Mobile Application Development | Mobile Application Development | 1  | 1  | 0  | 0  |
| 325281158 | Mobile Application Development | Mobile Application Development | 1  | 1  | 0  | 0  |
| 325281161 | Mobile Application Development | Mobile Application Development | 1  | 1  | 0  | 0  |
| 325281167 | Mobile Application Development | Mobile Application Development | 1  | 1  | 0  | 0  |
| 325281171 | Data Analytics                 | Enterprise Information System  | 0  | 0  | 1  | 1  |

|           |                                |                                |   |   |   |   |
|-----------|--------------------------------|--------------------------------|---|---|---|---|
| 325281194 | Game Development               | Game Development               | 1 | 1 | 0 | 0 |
| 325281197 | Artificial Intelligence        | Game Development               | 0 | 0 | 1 | 1 |
| 325281200 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281201 | Game Development               | Mobile Application Development | 0 | 0 | 1 | 1 |
| 325281207 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281211 | Cyber Security                 | Mobile Application Development | 0 | 0 | 1 | 1 |
| 325281217 | Data Analytics                 | Data Analytics                 | 1 | 1 | 0 | 0 |
| 325281238 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281246 | Mobile Application Development | Mobile Application Development | 1 | 1 | 0 | 0 |
| 325281247 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281267 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281268 | Data Analytics                 | Data Analytics                 | 1 | 1 | 0 | 0 |
| 325281269 | Mobile Application Development | Mobile Application Development | 1 | 1 | 0 | 0 |
| 325281145 | Game Development               | Game Development               | 1 | 1 | 0 | 0 |
| 325281187 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281165 | Enterprise Information System  | Enterprise Information System  | 1 | 1 | 0 | 0 |
| 325281198 | Game Development               | Game Development               | 1 | 1 | 0 | 0 |
| 325281254 | Mobile Application             | Mobile Application             | 1 | 1 | 0 | 0 |

|               | Developme<br>nt                          | Developme<br>nt                          |   |   |   |   |
|---------------|--|--|---|---|---|---|
| 32528114<br>4 | Cyber<br>Security                        | Mobile<br>Application<br>Developme<br>nt | 0 | 0 | 1 | 1 |
| 32528120<br>6 | Mobile<br>Application<br>Developme<br>nt | Game<br>Developme<br>nt                  | 0 | 0 | 1 | 1 |
| 32528124<br>4 | Artificial<br>Intelligence               | Data<br>Analytics                        | 0 | 0 | 1 | 1 |
| 32528123<br>0 | Mobile<br>Application<br>Developme<br>nt | Mobile<br>Application<br>Developme<br>nt | 1 | 1 | 0 | 0 |
| 32528115<br>2 | Enterprise<br>Information<br>System      | Enterprise<br>Information<br>System      | 1 | 1 | 0 | 0 |

Berdasarkan hasil pengujian terhadap 30 *user* didapatkan nilai TP berjumlah 22, TN berjumlah 22, FP berjumlah 8 dan FN berjumlah 8. Untuk mengukur tingkat akurasi metode dapat dilakukan pengukuran tingkat akurasi dari sistem dengan menggunakan rumus akurasi dari *confusion matrix*. Penerapan rumus akurasi sebagai berikut:

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% = \frac{44}{60} \times 100\% = 73.3\%$$

Dari hasil perhitungan menggunakan rumus akurasi untuk 30 *user*, didapatkan akurasi dari metode sebesar 73.3%.

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Dari hasil pembuatan sistem yang dilakukan, dapat ditarik kesimpulan, yaitu:

- Berdasarkan hasil pengujian terhadap 30 *user* dengan menggunakan *confusion matrix*, tingkat keakuratan metode *Aggregate Function* dalam sistem rekomendasi memperoleh angka 73.3%. Dari hasil perhitungan akurasi didapatkan nilai akurasi yang dihasilkan telah melebihi target yang diharapkan. Kesimpulan ini menjawab rumusan masalah no 1.

### 5.2 Saran

Saran yang dapat diberikan untuk peneliti agar dapat mengembangkan program lebih lanjut, adalah:

- Mengubah metode *User Based Collaborative Filtering* pada kriteria 1 menjadi *Hybrid Collaborative Filtering*.

- Mengubah cara untuk mengukur *passion* mahasiswa menggunakan cara lain yang lebih akurat.
- Menambah batasan minimal mata kuliah dalam 1 konsentrasi untuk dapat masuk ke perhitungan nilai rata-rata prediksi pada kriteria 1.
- Menambah bobot untuk setiap mata kuliah pada kriteria 2.

## 6. DAFTAR PUSTAKA

- [1] Ferio, G., Intan, R., & Rostianingsih, S. 2019. *Sistem Rekomendasi Mata Kuliah Pilihan Menggunakan Metode User Based Collaborative Filtering Berbasis Algoritma Adjusted Cosine Similarity*. *Jurnal Infra*, 7(1), 1–7.
- [2] Fitri, A., Putra, H., Mahmudy, W. F., & Setiawan, B. D. 2015. *Sistem rekomendasi mata kuliah pilihan mahasiswa dengan content-based filtering dan collaborative filtering* (studi kasus: universitas brawijaya). 17, 1–11.
- [3] Kaushik, S., & Tomar, P. 2015. *Evaluation of Similarity Functions by using User based Collaborative Filtering approach in Recommendation Systems*. *International Journal of Engineering Trends and Technology*, 21(4), 194–200. <https://doi.org/10.14445/22315381/ijett-v21p234>
- [4] Maulidah, M., Gata, W., Aulianita, R., & Agustyaningrum, C. I. 2020. Algoritma Klasifikasi Decision Tree Untuk Rekomendasi Buku Berdasarkan Kategori Buku. *Jurnal Ilmiah Ekonomi Dan Bisnis*, 13(2), 89–96.
- [5] Ricci, F., Rokach, L., & Shapira, B. 2011. *Recommender Systems Handbook*. In *Recommender Systems Handbook (Issue October)*. <https://doi.org/10.1007/978-0-387-85820-3>
- [6] Saputro, I. W., Sari, B. W. 2020. Chandra, W. N., Indrawan, G., & Sukajaya, I. N. 2016. Spam Filtering Dengan Metode Pos Tagger Dan Klasifikasi Naïve Bayes. *Jurnal Ilmiah Teknologi Informasi Asia*, 10(1), 47–55. Smyrniw, W. 2016. 1. Introduction 11. Ukrainian Science Fiction. <https://d.creativeinformationtechnologyjournal.com/>, 6(1), 1.
- [7] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. 2001. *Item-based collaborative filtering recommendation algorithms*. Proceedings of the 10th International Conference on World Wide Web, WWW 2001, December 2001, 285–295. <https://doi.org/10.1145/371920.372071>
- [8] Wiedjarnarko, S., Intan, R., & Setiawan, A. 2018. *Perencanaan dan Pembuatan Mobile Apps Informasi dan Prediksi Harga Emas dalam Rupiah melalui Agregasi Harga Emas dan Kurs Dollar*. *Jurnal Infra*, 6(1), 51–55. <http://publication.petra.ac.id/index.php/teknik-informatika/article/view/>
- [9] Yanti, N., Rahmi, R., & Ruliah. 2013. *Penerapan Algoritma Collaborative Filtering Untuk Rekomendasi Games Hardware*. *Jurnal Teknik Informatika Dan Sistem Informasi (JUTISI)*, 2(1), 305–314.