

# SKRIPSI

## SISTEM REKOMENDASI PROGRAM STUDI UNIVERSITAS KATOLIK PARAHYANGAN MENGGUNAKAN ALGORITMA *COLLABORATIVE FILTERING*



Anugrah Jaya Sakti

NPM: 2016730053

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN  
2020



# UNDERGRADUATE THESIS

## PARAHYANGAN CATHOLIC UNIVERSITY STUDY PROGRAM RECOMMENDATION SYSTEM USING *COLLABORATIVE FILTERING* ALGORITHM



Anugrah Jaya Sakti

NPM: 2016730053

DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY  
2020



# LEMBAR PENGESAHAN

SISTEM REKOMENDASI PROGRAM STUDI UNIVERSITAS KATOLIK  
PARAHYANGAN MENGGUNAKAN ALGORITMA *COLLABORATIVE FILTERING*

Anugrah Jaya Sakti

NPM: 2016730053

Bandung, «tanggal» «bulan» 2020

Menyetujui,

Pembimbing Utama

Pembimbing Pendamping

Husnul Hakim, M.T.

«Pembimbing Pendamping»

Ketua Tim Penguji

Anggota Tim Penguji

Rosa De Lima, M.Kom.

«penguji 2»

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng



## PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

**SISTEM REKOMENDASI PROGRAM STUDI UNIVERSITAS KATOLIK  
PARAHYANGAN MENGGUNAKAN ALGORITMA *COLLABORATIVE FILTERING***

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,  
Tanggal «tanggal» «bulan» 2020

Meterai Rp. 6000
---------------------

Anugrah Jaya Sakti  
NPM: 2016730053





## ABSTRAK

«Tuliskan abstrak anda di sini, dalam bahasa Indonesia»

**Kata-kata kunci:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Indonesia»



## ABSTRACT

«Tuliskan abstrak anda di sini, dalam bahasa Inggris»

**Keywords:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Inggris»



*«kepada siapa anda mempersembahkan skripsi ini...?»*



## KATA PENGANTAR

Segala puji bagi Allah SWT yang telah memberikan rahmat dan karunia-Nya kepada penulis, sehingga penulis dapat menyelesaikan skripsi dengan judul "Rekomendasi Program Studi di Perguruan Tinggi untuk Siswa SMA". Shalawat dan salam senantiasa tercurah kepada Rasulullah SAW. Penyusunan skripsi ini dimaksudkan untuk memenuhi syarat guna mencapai gelar sarjana di Program studi Teknik Informatika, Fakultas Teknologi Informasi dan Sains, Universitas Parahyangan Bandung. Penulis menyadari bahwa penulisan ini tidak dapat terselesaikan tanpa dukungan dari berbagai pihak, baik moril maupun materil. Oleh karena itu, penulis ingin mengucapkan ucapan terima kasih kepada :

1. Kedua orang tua, Bapak Ibrahim Warga Purawinata dan (Almh) Ibu Dede Komariah yang telah memberikan dukungan baik moril maupun materil serta doa.
2. Syarif Jordan .
3. Bapak Husnul Hakim selaku Dosen Pembimbing.
4. Ibu Flaviana selaku Wakil Dekan Bidang Kemahasiswaan dan Alumni yang membantu dalam pembuatan surat permohonan kepada Biro Admiristari. Akademik.
5. Seluruh Bapak/Ibu dosen Teknik Informatika yang telah memberikan pengetahuan.
6. Seluruh teman-teman yang telah memeberikan semangat.

Bandung, «bulan» 2020

Anugrah Jaya Sakti





# DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	2
1.3 Tujuan . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Metodologi . . . . .	3
1.6 Sistematika Pembahasan . . . . .	3
<b>2 LANDASAN TEORI</b>	<b>5</b>
2.1 Sistem Rekomendasi . . . . .	5
2.1.1 Fungsi Sistem Rekomendasi . . . . .	5
2.1.2 Sumber Data dan Pengetahuan . . . . .	6
2.1.3 Teknik Rekomendasi . . . . .	6
2.1.4 <i>Collaborative Filtering</i> . . . . .	7
2.1.5 Aplikasi dan Evaluasi . . . . .	8
2.2 <i>Cluster</i> . . . . .	9
2.2.1 K-Means . . . . .	9
2.3 <i>Library</i> PHP-ML . . . . .	9
2.3.1 Array Dataset . . . . .	10
2.3.2 Random Split . . . . .	10
2.4 Universitas Katolik Parahyangan . . . . .	10
2.4.1 Program Studi . . . . .	10
2.4.2 Syarat Masuk Program Studi . . . . .	15
2.4.3 Karakteristik Program Studi . . . . .	16
<b>3 ANALISIS</b>	<b>19</b>
3.1 Analisis Perangkat Lunak Sejenis . . . . .	19
3.2 Pemilihan Algoritma Sistem Rekomendasi . . . . .	22
3.3 <i>Preprocessing</i> Data Mahasiswa . . . . .	23
3.4 Contoh Perhitungan Pearson Correlation . . . . .	23
3.5 Contoh Perhitungan Metode Evaluasi Sistem Rekomendasi . . . . .	25
3.6 Analisis Kebutuhan Sistem . . . . .	26
3.6.1 Diagram <i>Use Case</i> . . . . .	27
3.6.2 Rancangan Basis Data . . . . .	29

<b>4</b>	<b>PERANCANGAN</b>	<b>31</b>
4.1	Perancangan Fisik Basis Data . . . . .	31
4.1.1	Perancangan Tabel . . . . .	31
4.2	Perancangan Algoritma . . . . .	32
4.2.1	K-Means . . . . .	32
4.2.2	Pearson Correlation Coefficient . . . . .	33
4.2.3	User-based Collaborative Filtering . . . . .	33
4.3	Perancangan Antar Muka . . . . .	33
<b>5</b>	<b>IMPLEMENTASI DAN PENGUJIAN</b>	<b>35</b>
5.1	Lingkungan Implementasi . . . . .	35
5.1.1	Lingkungan Perangkat Keras . . . . .	35
5.2	Lingkungan Perangkat Lunak . . . . .	35
5.3	Implementasi Tabel Basis Data . . . . .	35
5.4	Impelemtasi Antar Muka . . . . .	37
5.5	Pengujian Fungsional . . . . .	39
5.5.1	Pengujian Fungsional Pemilihan Jurusan SMA . . . . .	39
5.5.2	Pengujian Fungsional Pengisian Nilai . . . . .	39
5.6	Pengujian Eksperimental . . . . .	40
5.6.1	Metode Dasar . . . . .	40
5.6.2	Metode KMeans . . . . .	40
<b>6</b>	<b>KESIMPULAN DAN SARAN</b>	<b>43</b>
6.1	Kesimpulan . . . . .	43
6.2	Saran . . . . .	43
	<b>DAFTAR REFERENSI</b>	<b>45</b>
	<b>A KODE PROGRAM</b>	<b>47</b>

## DAFTAR GAMBAR

3.1	7 Dimensi Profil Siswa . . . . .	19
3.2	Tampilan setelah registrasi atau <i>login</i> . . . . .	20
3.3	Modul Pengenalan . . . . .	20
3.4	Modul Potensi Diri . . . . .	21
3.5	Modul Ukur Kemampuan Diri . . . . .	21
3.6	Hasil Rekomendasi . . . . .	22
3.7	Diagram <i>Use Case</i> Sistem Rekomendasi . . . . .	28
3.8	Diagram ERD Sistem Rekomendasi . . . . .	29
4.1	Halaman Index Sistem . . . . .	33
4.2	Halaman Pengisian Nilai IPA . . . . .	33
4.3	Halaman Pengisian Nilai IPS . . . . .	34
4.4	Halaman Hasil Rekomendasi . . . . .	34
5.1	Halaman Index Sistem . . . . .	37
5.2	Halaman Index Pengisian Nilai IPA . . . . .	38
5.3	Halaman Index Pengisian Nilai IPS . . . . .	38
5.4	Halaman Hasil Rekomendasi . . . . .	39
5.5	Hasil Pengujian Jurusan IPA dengan Metode Dasar . . . . .	40
5.6	Hasil Pengujian Jurusan IPS dengan Metode Dasar . . . . .	40
5.7	Hasil Pengujian Jurusan IPA dengan Metode KMeans . . . . .	41
5.8	Hasil Pengujian Jurusan IPS dengan Metode KMeans . . . . .	42



## DAFTAR TABEL

2.1	Tabel syarat program studi . . . . .	16
2.2	Tabel kriteria . . . . .	17
3.1	Contoh data mahasiswa dalam bentuk GPA . . . . .	23
3.2	Contoh data siswa dalam bentuk GPA . . . . .	24
3.3	Nilai kovariasi mahasiswa dan siswa . . . . .	24
3.4	Standar Deviasi Siswa . . . . .	24
3.5	Standar Deviasi Mahasiswa . . . . .	25
3.6	Contoh Perhitungan kesamaan . . . . .	25
3.7	Contoh hasil Prediksi . . . . .	25
3.8	Tabel Data MAE dan RMSE . . . . .	26
3.9	Pendefinisian Aktor . . . . .	27
3.10	Pendefinisian <i>Use Case</i> . . . . .	27
3.11	Skenario Memilih Jurusan SMA . . . . .	27
3.12	Skenario Mengisi Nilai Rapor . . . . .	28
4.1	Perancangan Tabel jurusan_sma . . . . .	31
4.2	Perancangan Tabel fakultas . . . . .	31
4.3	Perancangan Tabel program_studi . . . . .	31
4.4	Perancangan Tabel mahasiswa . . . . .	32
4.5	Perancangan Tabel mata_pelajaran . . . . .	32
4.6	Perancangan Tabel nilai . . . . .	32
5.1	Tabel Pengujian Fungsional Pemilihan SMA . . . . .	39
5.2	Tabel Pengujian Fungsional Pengisian Nilai . . . . .	40



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Salah satu tahapan pendidikan setelah lulus dari bangku sekolah menengah atas atau SMA adalah dengan melanjutkan studi ke perguruan tinggi baik perguruan tinggi negeri ataupun swasta. Salah satu hal yang perlu diperhatikan saat akan melanjutkan studi di perguruan tinggi adalah program studi apa yang akan dipilih. Program studi adalah kesatuan rencana belajar sebagai pedoman penyelenggaraan pendidikan akademik dan/atau profesional yang diselenggarakan atas dasar suatu kurikulum serta ditujukan agar mahasiswa dapat menguasai pengetahuan, keterampilan, dan sikap sesuai dengan sasaran kurikulum.

Kesalahan dalam memilih program studi memiliki dampak yang signifikan bagi kehidupan mahasiswa dimasa mendatang. Dampak bisa berupa masalah psikologi, mahasiswa akan merasa terpaksa saat belajar karena mempelajari sesuatu hal yang tidak sesuai minat. Selain masalah psikologi dampak lain yang bisa terjadi berupa masalah pada bidang akademik, prestasi seorang mahasiswa tidak akan maksimal, nilai mata kuliah kurang baik, dan mahasiswa yang salah dalam memilih jurusan memiliki kemungkinan yang lebih tinggi mengalami *Drop Out*.

Ketidakcocokan program studi dengan mahasiswa di Indonesia masih cukup tinggi. Berdasarkan buku Statistik Pendidikan Tinggi pada tahun 2017, terdapat 1.437.425 mahasiswa baru, 6.924.511 mahasiswa terdaftar, dan 1.046.141 mahasiswa lulus. Dengan kata lain ada 391.284 atau 27.22% mahasiswa yang tidak lulus. Jumlah mahasiswa *Drop Out* pada tahun 2017 adalah 195.176 dengan presentasi pada Perguruan Tinggi Negeri (PTN) sebesar 96% dan pada Perguruan Tinggi Swasta (PTS) sebesar 4%. Presentasi jumlah mahasiswa lulus tepat waktu merupakan salah satu faktor yang menentukan kualitas universitas (PP No. 66 Tahun 2010) selain itu, menurut Sudjito (2014): kecocokan program studi merupakan salah satu penentu keberhasilan studi dari seorang mahasiswa.

Maka dari itu diperlukan sebuah sistem rekomendasi yang dapat memberikan rekomendasi item berupa program studi yang sesuai dengan minat siswa SMA. Sistem rekomendasi adalah alat dan teknik perangkat lunak yang menyediakan saran untuk item yang akan digunakan oleh pengguna. Sistem rekomendasi berfokus pada item tertentu dan ditujukan untuk individu atau personal. Beberapa teknik yang biasa digunakan pada sistem rekomendasi, yaitu : *Content-based*, *Collaborative Filtering*, *Demographic*, *Knowledge-based*, *Community-based*, *Hybrid recommender systems*. Teknik yang akan digunakan pada sistem rekomendasi yang akan dibangun adalah *Collaborative Filtering*.

*Collaborative Filtering* merupakan teknik yang merekomendasikan item yang sesuai dengan kebutuhan pengguna berdasarkan *rating* tanpa memerlukan informasi mengenai item ataupun pengguna, contoh informasi yang dimaksud adalah deskripsi mengenai item atau pengguna. Secara sederhana *Collaborative Filtering* menghitung kesamaan atau similaritas antara pengguna aktif dengan beberapa pengguna yang memiliki selera atau minat yang serupa. Untuk menghitung similaritas digunakan metode *Pearson Correlation Coefficient*. *Pearson Correlation Coefficient*

bekerja dengan cara menghitung korelasi antara dua variabel dari masing-masing pengguna yang sedang dibandingkan. Semakin tinggi nilai korelasi yang dihasilkan maka mengidentifikasi kedua pengguna memiliki similaritas yang cukup tinggi.

Pada skripsi ini akan dibangun sebuah perangkat lunak sistem rekomendasi yang dapat memberikan rekomendasi item berupa program studi yang sesuai dengan minat siswa SMA. Sistem rekomendasi ini akan menggunakan algoritma *Collaborative Filtering* dengan model *Neighborhood* dengan pendekatan *User-based*. *User-based* memprediksi berdasarkan kesamaan *rating* pengguna dengan item.

*Rating* yang dalam kasus ini adalah indeks prestasi kumulatif (IPK) . Berdasarkan Pasal 23 ayat 5 Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor Tahun 2014 tentang Standar Nasional Pendidikan Tinggi yang berbunyi Hasil penilaian capaian pembelajaran lulusan pada akhir program studi dinyatakan dengan indeks prestasi kumulatif (IPK).

## 1.2 Rumusan Masalah

Berikut adalah rumusan masalah dari penulisan skripsi :

1. Bagaimana cara menilai kecocokan seorang calon mahasiswa terhadap suatu program studi ?
2. Bagaimana membangun perangkat lunak untuk memberikan rekomendasi program studi di perguruan tinggi yang cocok untuk calon mahasiswa ?
3. Bagaimana kualitas hasil rekomendasi dari perangkat lunak yang dibangun ?

## 1.3 Tujuan

Tujuan dari penulisan skripsi ini adalah sebagai berikut :

1. Mempelajari cara menilai kecocokan seorang mahasiswa terhadap suatu program studi.
2. Membangun perangkat lunak untuk memberikan rekomendasi program studi di perguruan tinggi yang cocok dengan calon mahasiswa.
3. Menguji hasil rekomendasi dari perangkat lunak yang sudah dibangun.

## 1.4 Batasan Masalah

Mengingat banyaknya perguruan tinggi dan program studi yang ada di Indonesia, maka perlu adanya batasan masalah yang jelas mengenai apa yang dibuat dan diselesaikan dalam penulisan skripsi ini. Berikut merupakan batasan-batasan masalah pada skripsi ini :

1. 15 Program studi Universitas Katolik Parahyangan (UNPAR).
2. Data mahasiswa UNPAR yang masuk melalui jalur Penelusuran Minat dan Kemampuan (PMDK) pada tahun 2013-2018 yang sudah lulus.
3. Hanya menggunakan nilai mata pelajaran Matematika, Bahasa Indonesia, Bahasa Inggris, Pendidikan Kewarganegaraan, Fisika, dan Kimia sebagai atribut.



## 1.5 Metodologi

1. Melakukan studi literatur mengenai sistem rekomendasi.
2. Mempelajari mengenai berbagai program studi dan karakteristiknya.
3. Mempelajari metode yang dapat digunakan untuk menghitung tingkat kecocokan calon mahasiswa dengan program studi.
4. Menganalisis hal-hal yang mempengaruhi kecocokan program studi dengan calon mahasiswa.
5. Mempelajari framework Laravel dan Bootstrap.
6. Membangun perangkat lunak sesuai dengan analisis dan perancangan yang dilakukan.
7. Melakukan pengujian kualitas hasil rekomendasi perangkat lunak yang dibangun.
8. Menulis dokumen skripsi.

## 1.6 Sistematika Pembahasan

1. Bab 1 menjelaskan mengenai latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan untuk sistem rekomendasi program studi di perguruan tinggi untuk anak SMA.
2. Bab 2 menjelaskan mengenai sistem rekomendasi dengan menggunakan *Collaborative Filtering*, teknik perhitungan kesamaan atau similaritas dengan *Pearson Correlation Coefficient*, teknik *clustering* menggunakan K-Means, penjelasan *Framework Laravel*, dan Program Studi yang berada di Universitas Katolik Parahyangan.
3. Bab 3
4. Bab 4
5. Bab 5



## BAB 2

### LANDASAN TEORI

#### 2.1 Sistem Rekomendasi

Sistem rekomendasi adalah alat dan teknik perangkat lunak yang menyediakan saran untuk item yang akan digunakan oleh pengguna. Saran terkait dengan berbagai proses pengambilan keputusan, seperti barang apa yang akan dibeli, musik apa yang akan didengarkan, atau berita online apa yang akan dibaca.

Sistem rekomendasi biasanya berfokus pada item tertentu seperti buku, musik, dll. Sistem rekomendasi ditujukan untuk individu atau personal yang kurang memiliki pengalaman pribadi. Contoh Sistem rekomendasi buku adalah *website* Amazon.com. Item yang ditawarkan sebagai daftar item peringkat. Sistem rekomendasi mencoba memprediksi produk dengan cara mengumpulkan referensi dari pengguna lainnya.

Pengembangan sistem rekomendasi dimulai dari pengamatan yang sederhana berupa rekomendasi yang diberikan oleh orang lain dalam membuat keputusan rutin sehari-hari bisa berupa buku, musik, film, rekrutmen karyawan, dll.

Sistem rekomendasi menghasilkan rekomendasi menggunakan berbagai jenis pengetahuan dan data tentang pengguna, item yang tersedia, dan transaksi sebelumnya, contohnya berupa e-commerce yang mengatasi masalah kelebihan informasi yang terjadi akibat transaksi pengguna sebelumnya.

##### 2.1.1 Fungsi Sistem Rekomendasi

Fungsi utama sistem rekomendasi adalah menemukan item yang relevan dengan kebutuhan pengguna. Selain untuk menemukan item yang relevan terdapat juga beberapa fungsing sistem rekomendasi, yaitu :

1. Meningkatkan jumlah penjualan barang  
Salah satu fungsi penting untuk sistem rekomendasi yang komersil. Peningkatan jumlah penjualan item ini disebabkan karena penjualan item dilakukan tepat sasaran kepada pembeli yang memang membutuhkan dan menginginkan item tersebut. Merekomendasikan item yang sesuai dengan kebutuhan atau minat pengguna.
2. Menjual barang-barang yang lebih beragam  
Memberikan rekomendasi item yang mungkin sulit ditemukan oleh pengguna jika tanpa menggunakan sistem rekomendasi.
3. Meningkatkan kepuasan pengguna  
Sistem rekomendasi yang dirancang dengan baik memberikan rekomendasi yang sesuai dengan kebutuhan pengguna sehingga pengguna akan merasa senang menggunakan sistem tersebut.

4. Meningkatkan kesetiaan pengguna  
Pengguna akan tetap menggunakan sebuah *website* jika sistem rekomendasi yang dihasilkan rekomendasi yang sesuai dengan kebutuhan pengguna.
5. Lebih mengerti apa yang diinginkan pengguna  
Sistem dapat memberikan hasil rekomendasi item yang sesuai dengan kebutuhan pengguna.

### 2.1.2 Sumber Data dan Pengetahuan

Sistem rekomendasi adalah sistem pemrosesan informasi yang secara aktif mengumpulkan berbagai jenis data untuk membangun rekomendasinya. Data utama berupa data item yang disarankan dan pengguna yang akan menerima rekomendasi. Data yang digunakan sistem rekomendasi mencakup pada tipe jenis objek, yaitu :

1. Item  
Item adalah objek yang direkomendasikan, item bisa ditandai oleh kompleksitasnya dan nilai atau kegunaannya. Bisa bernilai positif jika sesuai atau negatif jika tidak sesuai.
2. Pengguna  
Pengguna adalah objek yang menggunakan sistem, memiliki tujuan dan karakteristik beragam. Pengguna juga dapat dijelaskan oleh data pola perilaku (pola penelusuran web, atau pola pencarian perjalanan)
3. Transaksi  
Interaksi yang direkam antara pengguna sistem rekomendasi. Transaksi adalah data seperti log yang menyimpan informasi penting yang dihasilkan selama interaksi manusia-komputer dan berguna untuk algoritma pembuatan rekomendasi yang digunakan sistem. Bentuk dari peringkat yang populer di sistem rekomendasi :
  - Peringkat numerik 1 - 5
  - Peringkat ordinal (sangat setuju, setuju, netral, tidak setuju, dan sangat tidak setuju)
  - Peringkat biner, buruk (0) dan baik (1)
  - Peringkat unary menunjukkan bahwa pengguna telah mengamati atau membeli barang atau menilai barang secara positif

### 2.1.3 Teknik Rekomendasi

Berikut adalah teknik-teknik yang dapat digunakan pada sistem rekomendasi :

1. *Content-based*  
Sistem merekomendasikan item yang mirip berdasarkan item yang disukai pengguna di masa lalu. Kesamaan dihitung berdasarkan fitur (atribut) yang terkait dengan item. misal , review positif film komedi, maka akan direkomendasikan film di genre yang sama.
2. *Collaborative Filtering*  
Rekomendasi berdasarkan item yang disukai pengguna lain yang memiliki kesamaan. Implementasi paling sederhana, merekomendasikan item yang disukai pengguna lain dengan selera serupa di masa lalu. *Collaborative Filtering* populer dan banyak digunakan pada sistem rekomendasi. *Nearest neighbors* meningkatkan popularitas karena sederhana, efisien, dan kemampuan mereka untuk menghasilkan rekomendasi yang akurat dan menunjukkan ciri personal tertentu.

### 3. *Demographic*

Rekomendasi berdasarkan profil demografis pengguna. Asumsinya bahwa rekomendasi yang berbeda harus dihasilkan untuk demografis yang berbeda. Misalnya diarahkan ke web dengan bahasa atau negara pengguna.

### 4. *Knowledge-based*

Merekomendasikan item berdasarkan pengetahuan domain spesifik tentang fitur (atribut) item tertentu yang memenuhi kebutuhan atau referensi pengguna.

### 5. *Community-based*

Merekomendasikan item berdasarkan teman-teman pengguna. Bukti menunjukkan bahwa orang cenderung lebih mengandalkan rekomendasi dari teman-teman dari pada rekomendasi dari orang yang belum dikenal.

### 6. *Hybrid recomender systems*

Kombinasi dari beberapa teknik yang sudah disebutkan sebelumnya. Menggunakan teknik A dan B mencoba untuk menggunakan keunggulan A dan memperbaiki kelemahan B. Contoh, *Collaborative Filtering* memiliki kelemahan terhadap item yang tidak memiliki peringkat (tidak terdapat riwayat) bisa digabungkan dengan metode *Content-based*.

## 2.1.4 *Collaborative Filtering*

Dalam pengembangan sistem rekomendasi dapat menggunakan teknik *Collaborative Filtering*. *Collaborative Filtering* menghasilkan rekomendasi item yang spesifik untuk pengguna berdasarkan peringkat tanpa memerlukan informasi tambahan mengenai item ataupun pengguna. Gagasan utamanya adalah peringkat pengguna  $u$  untuk item  $i$  cenderung mirip dengan pengguna  $v$ , jika  $u$  dan  $v$  memberikan peringkat item lain dengan nilai yang sama.

Tantangan dalam membangun sistem rekomendasi menggunakan teknik *Collaborative Filtering* adalah sedikitnya jumlah data pengguna sebelumnya yang sudah memberikan peringkat kepada suatu item. Dalam *Collaborative Filtering* terdapat salah satu algoritma yaitu *Neighborhood-based Collaborative Filtering* atau yang dikenal dengan *Memory-base Collaborative Filtering*.

### *Neighborhood-based Collaborative Filtering*

*Neighborhood-based Collaborative Filtering* atau yang dikenal dengan *Memory-base Collaborative Filtering* adalah algoritma pertama yang dikembangkan untuk teknik *Collaborative Filtering*. Pada algoritma ini *rating user-item* disimpan dalam sistem secara langsung digunakan untuk memprediksi peringkat item baru, dapat dilakukan dengan *user-based model*.

### *User-based Neighborhood Model*

*User-based* bekerja dengan mengidentifikasi pengguna yang akan diberikan rekomendasi dengan pengguna lain yang memiliki kesamaan. Aktivitas pengguna yang memiliki kesamaan ini akan menjadi dasar dalam memberikan rekomendasi kepada pengguna lain. Aktivitas bisa berupa memberikan *rating* kepada item. Berikut adalah tahapan yang perlu dilakukan pada *User-based Neighborhood Model* :

1. Menghitung nilai rata-rata *rating* yang sudah diberikan oleh pengguna lain.
2. Menghitung kesamaan atau similaritas pengguna menggunakan *Pearson Correlation Coefficient*

$$sim(i, j) = Pearson(i, j) = \frac{\sum_{k \in I_i \cap I_j} (r_{i,k} - \mu_i) \cdot (r_{j,k} - \mu_j)}{\sqrt{\sum_{k \in I_i \cap I_j} (r_{i,k} - \mu_i)^2} \cdot \sqrt{\sum_{k \in I_i \cap I_j} (r_{j,k} - \mu_j)^2}} \quad (2.1)$$

Keterangan :

- $\text{sim}(i,j)$  = Kesamaan atau similaritas antara pengguna i dan pengguna j
  - $\Sigma_{k \in I_i \cap I_j}$  = Himpunan item pengguna i dan pengguna j yang saling beririsan
  - $r_{i,k}$  = Nilai yang diberikan pengguna i terhadap item k
  - $r_{j,k}$  = Nilai yang diberikan pengguna j terhadap item k
  - $\mu_i$  = Rata-rata nilai yang diberikan pengguna i
  - $\mu_j$  = Rata-rata nilai yang diberikan pengguna j
3. Memilih nilai kesamaan atau similaritas yang bernilai lebih besar dari 0. Nilai kesamaan atau similaritas memiliki rentan nilai -1, 0, dan +1 untuk pengguna yang akan diberikan rekomendasi. Jika hasil perhitungan mendekati -1, berarti pengguna tersebut kurang memiliki kesamaan dengan pengguna yang akan diberikan rekomendasi. Jika hasil perhitungan mendekati 0, berarti pengguna tersebut memiliki kesamaan yang cukup baik dengan pengguna yang akan diberikan rekomendasi. Jika hasil perhitungan mendekati +1, berarti pengguna tersebut memiliki kesamaan yang tinggi dengan pengguna yang akan diberikan rekomendasi.
4. Menghitung nilai prediksi dengan rumus *weighted sum* 2.2 :

$$r_{i,k} = \frac{\Sigma(\text{Sim}(i,j) * r_{j,k})}{\Sigma \text{Sim}(i,j)} \quad (2.2)$$

Keterangan :

- $r_{i,k}$  = Nilai prediksi pengguna i untuk item k
  - $\text{Sim}(i,j)$  = kesamaan atau similiaritas pengguna i dan pengguna j
  - $r_{j,k}$  = Penilaian pengguna j terhadap item k
5. Mengurutkan nilai prediksi dari yang terbesar ke terkecil.

### 2.1.5 Aplikasi dan Evalusi

#### Aplikasi

Faktor pertama yang harus dipertimbangkan adalah domain aplikasi yang akan dibangun karena memiliki efek yang besar pada algoritma yang akan digunakan. Kelas domain paling umum :

1. Entertainment : rekomendasi film dan musik
2. Content : personalisasi berita, dokukumen, dan web page
3. E-commerce : rekomendasi produk untuk di beli
4. Services : rekomendasi servis travel, hotel, dan rumah

## Evalusi

Sebuah sistem rekomendasi banyak digunakan untuk memberikan prediksi berupa saran item yang sesuai dengan minat pengguna. Prediksi yang diberikan sistem rekomendasi memiliki nilai keakuratan yang dapat berbeda sesuai dengan kasus yang dihadapi dan juga algoritma yang digunakan. Prediksi yang diberikan harus akurat, oleh karena itu diperlukan evaluasi pada sistem rekomendasi. Evaluasi dapat menggunakan tiga metode yaitu :

1. *Offline*

Metode *offline* dilakukan dengan cara menjalankan beberapa algoritma pada data yang sama dan membandingkan kinerjanya.

2. *Online*

Metode *online* dilakukan saat perangkat lunak sudah diluncurkan dan melibatkan pengguna nyata.

3. *Focused user study*

Metode *Focused user study* dilakukan saat metode *online* tidak layak dilakukan atau terlalu beresiko.

## 2.2 Cluster

*Clustering* adalah algoritma yang menganalisis objek data tanpa perlu label kelas. *Clustering* bisa digunakan untuk menghasilkan label kelas untuk sebuah kelompok data. Tujuan dari algoritma *cluster* adalah untuk meminimalkan jarak *intra-cluster* sekaligus memaksimalkan jarak *inter-cluster*. Kesamaan ditentukan dengan menggunakan ukuran jarak. Kelompok yang dihasilkan akan memiliki anggota yang memiliki kesamaan yang tinggi satu sama lain didalam kelompok yang sama dan berbeda dengan kelompok lain.

### 2.2.1 K-Means

K-Means adalah algoritma *clustering* yang termasuk kategori *partisial*. Kategori partisial adalah membagi item kedalam *non-overlapping cluster* sehingga setiap item hanya ada pada satu *cluster*. *Cluster* akan dibentuk sebanyak K dan untuk setiap *cluster* memiliki *centroid* awal. *Centroid* untuk setiap *cluster* adalah titik dimana jumlah jarak minimum dari semua item dalam *cluster*. Berikut merupakan tahapan-tahapan dalam pembuatan *cluster* :

1. Memilih secara acak K *centroid*
2. Menghitung jarak tiap objek ke titik *centroid* menggunakan *euclidean distance*
3. Menghitung *centroid* baru dari anggota yang berada didalam *centroid*
4. Lakukan tahap 1-3 hingga *centroid* konvergen

## 2.3 Library PHP-ML

PHP-ML adalah sebuah *library* yang khusus dibuat untuk *Machine Learning* dengan menggunakan bahasa pemrograman PHP. Terdapat lebih dari 20 algoritma yang bisa digunakan. *Library* ini bersifat *open source* yang berlisensi MIT. Versi PHP minimal untuk menggunakan *library* ini adalah PHP 7.1, penginstallan dapat menggunakan Composer.

### 2.3.1 Array Dataset

*Array Dataset* adalah bagian dari fitur *Dataset* yang disediakan oleh PHP-ML. *Array Dataset* adalah kelas yang berfungsi untuk menyimpan data sebagai tipe array dalam PHP. Menerapkan *interface Dataset* yang banyak digunakan di kelas lain. Kelas ini memiliki dua parameter yaitu : *samples* dan *labels*. *Samples* adalah array yang berisikan *sample*. *Labels* adalah array yang berisikan *label* setiap *sample*.

### 2.3.2 Random Split

*Random Split* adalah bagian dari fitur *Cross Validation* yang disediakan oleh PHP-ML. Kelas *Random Split* adalah salah satu metode paling sederhana dari *Cross Validation*. *Samples* dibagi menjadi dua kelompok yaitu : *train group* dan *test group*. Kelas ini memiliki tiga parameter yaitu : *dataset*, *testSize*, dan *seed*. *Dataset* adalah objek yang mengimplementasikan *interface Dataset*. *TestSize* adalah bilangan *float* yang menyatakan seberapa banyak anggota pada *test group* dengan nilai dasar 0.3 jika parameter tidak diisi. *Seed* untuk *random generator*.

## 2.4 Universitas Katolik Parahyangan

Perguruan tinggi adalah satuan pendidikan yang menyelenggarakan pendidikan tinggi yang dapat berbentuk akademi, politeknik, sekolah tinggi, institut, atau universitas. Pendidikan tinggi adalah kelanjutan pendidikan menengah yang diselenggarakan untuk menyiapkan peserta didik menjadi anggota masyarakat yang memiliki kemampuan akademik dan/atau profesional yang dapat menerapkan, mengembangkan dan/atau menciptakan ilmu pengetahuan, teknologi dan/atau kesenian.

Universitas Katolik Parahyangan adalah sebuah universitas atau Perguruan tinggi katolik pertama yang didirikan pada 17 Januari 1955. Saat ini terletak di Jalan Ciumbuleuit No.94, Bandung, Jawa Barat, Indonesia. Terdapat tujuh fakultas dengan total program studi yaitu tujuh belas dengan enam belas program studi sarjana dan satu program studi D3.

Terdapat beberapa jalur penerimaan mahasiswa baru yang dilakukan oleh Universitas Katolik Parahyangan. Jalur penerimaan diselenggarakan secara mandiri, berikut jalur penerimaan yang disediakan Universitas Katolik Parahyangan :

1. Penelusuran Minat dan Kemampuan (PMKD) atau jalur prestasi  
PMKD adalah satu jalur penerimaan mahasiswa baru yang dilaksanakan dengan seleksi berdasarkan pada nilai raport SMA di kelas X (Sepuluh) dan XI (Sebelas), tanpa ujian tertulis. Tujuan dari PMKD untuk menjaring siswa-siswa yang berprestasi. PMKD dilakukan hanya satu kali dalam satu tahun penerimaan.
2. Ujian Saringan Masuk (USM)  
USM adalah satu jalur penerimaan mahasiswa baru yang dilaksanakan dengan mengerjakan soal yang disediakan oleh Universitas Katolik Parahyangan. Terdapat dua tempat pelaksanaan untuk USM, pertama dilaksanakan di Universitas Katolik Parahyangan dan kedua dilaksanakan di sekolah-sekolah (*on-site test*). Tujuan dari USM untuk menjaring mahasiswa baru yang memiliki kemampuan akademik untuk mengikuti dan menyelesaikan pendidikan di Universitas Katolik Parahyangan sesuai dengan batas waktu (masa studi) yang ditetapkan.

### 2.4.1 Program Studi

Program studi adalah kesatuan rencana belajar sebagai pedoman penyelenggaraan pendidikan akademik dan/atau profesional yang diselenggarakan atas dasar suatu kurikulum serta ditujukan agar mahasiswa dapat menguasai pengetahuan, keterampilan, dan sikap sesuai dengan sasaran kurikulum. Kurikulum pendidikan tinggi adalah seperangkat rencana dan pengaturan mengenai



isi maupun bahan kajian dan pelajaran serta cara penyampaian dan penilaiannya yang digunakan sebagai pedoman penyelenggaraan kegiatan belajar - mengajar di perguruan tinggi.

Terdapat tujuh fakultas yang ada di Universitas Katolik Parahyangan, yaitu :

1. Fakultas Ekonomi
2. Fakultas Hukum
3. Fakultas Ilmu Sosial dan Ilmu Politik
4. Fakultas Teknik
5. Fakultas Falsafah dan Peradaban
6. Fakultas Teknologi Industri
7. Fakultas Teknologi Informasi dan Sains

### **Fakultas Ekonomi**

Terdapat empat program studi pada fakultas Ekonomi, yaitu : Ekonomi Pembangunan, Manajemen, Akuntansi, Manajemen Perusahaan. Manajemen Perusahaan merupakan program studi D3 yang ada di Universitas Katolik Parahyangan. Berikut merupakan penjelasan program studi yang ada pada Fakultas Ekonomi :

1. Ekonomi Pembangunan

Mempelajari persoalan pembangunan ekonomi yang sudah, sedang, dan akan terjadi di negara berkembang. Menganalisis isu perekonomian untuk mencari dan menemukan solusi dari berbagai persoalan ekonomi secara kritis, kreatif, dan inovatif. Program studi Ekonomi Pembangunan mempersiapkan mahasiswanya untuk menjadi perencana bidang pembangunan ekonomi. Ekonomi Pembangunan adalah cabang ilmu ekonomi. Mempelajari pembangunan industri, perbankan, keuangan, dan bisnis. Berkutat dengan analisis berbagai isu perekonomian untuk mendapatkan solusi dari persoalan ekonomi.

Terdapat tiga peminatan pada program studi Ekonomi Pembangunan, yaitu :

- Ekonomi Industri dan Perdagangan
- Ekonomi Kawasan dan Lingkungan
- Ekonomi Moneter dan Keuangan

2. Manajemen

Mempelajari bagaimana mengelola suatu perusahaan atau organisasi. Fokus pada kegiatan mengelola, merencanakan, dan mengatur semua proses dalam perusahaan untuk mencapai tujuan.

Terdapat satu peminatan pada program studi Manajemen, yaitu :

- Manajemen

3. Akuntansi

Mempelajari mengenai keuangan dan ilmu ekonomi, Mahasiswa pada program studi Akuntansi akan memiliki pengetahuan dan penguasaan materi tentang keuangan dan ilmu ekonomi.

Mampu mengelola keuangan bisnis.

Terdapat satu peminatan pada program studi Akuntansi, yaitu :

- Akuntansi

### **Fakultas Hukum**

Terdapat satu program studi pada Fakultas Hukum, yaitu : Ilmu Hukum.

#### **1. Ilmu Hukum**

Mempelajari tentang hukum baik praktek maupun teori. Hukum mengatur bagaimana manusia bertindak dan bertingkah laku agar tidak merugikan orang lain. Mendalami konsep, teori, dan beberapa kasus hukum yang terjadi.

Terdapat satu peminatan pada program studi Ilmu Hukum, yaitu :

- Ilmu Hukum

### **Fakultas Ilmu Sosial dan Ilmu Politik**

Terdapat tiga program studi pada Fakultas Ilmu Sosial dan Ilmu Politik, yaitu : Ilmu Administrasi Publik, Ilmu Administrasi Bisni, dan Ilmu Hubungan Internasional.

#### **1. Ilmu Administrasi Publik**

Mempelajari seluk beluk pemerintahan, masyarakat, dan kebijakan publik, sistem pemerintahan, pembuatan kebijakan hingga pengimplementasian dan evaluasi, pelayanan masyarakat, dan segala sesuatu yang berkaitan dengan birokrasi.

Terdapat satu peminatan pada program studi Ilmu Administrasi Publik, yaitu :

- Ilmu Administrasi Publik

#### **2. Ilmu Administrasi Bisni**

Mempelajari mengenai kegiatan operasional bisnis dan perusahaan, yaitu : pemasaran (marketing), pengelolaan keuangan, pengelolaan personalia (SDM), hingga kegiatan produksi. Mempelajari untuk membuat produk sendiri, bukan membuat, menjual, dan mendapatkan keuntungan, tetapi menciptakan value pada produk yang dipasarkan. Mempelajari urusan klarikal kantor, mengelola sarana dan prasarana kantor, memproses data secara akurat, dan mengelola informasi yang berhubungan dengan pekerjaan kantor. Program studi ini cocok dengan orang yang memiliki ketertarikan dalam bidang pengurusan dokumen.

Terdapat dua peminatan pada program studi Ilmu Administrasi Bisnis, yaitu :

- *General Business*
- *Digital Business*

### 3. Ilmu Hubungan Internasional

Mempelajari mengenai interaksi, relasi, dan komunikasi yang terjadi secara internasional. Tidak hanya mempelajari hubungan diplomasi satu negara dengan negara lain, tapi juga konflik, kesejahteraan, ekonomi, dan perdamaian dunia. Beberapa kajian diplomasi dan negosiasi, politik luar negeri, perdagangan luar negeri, politik internasional, ekonomi internasional, hukum internasional, globalisasi, dll. Diasah mengenai isu-isu global, tokoh-tokoh, dan organisasi internasional yang berpengaruh, dan kerjasama internasional.

Terdapat satu peminatan pada program studi Ilmu Administrasi Bisnis, yaitu :

- Ilmu Hubungan Internasional

## **Fakultas Teknik**

Terdapat dua program studi pada Fakultas Teknik, yaitu : Teknik Sipil dan Arsitektur.

### 1. Teknik Sipil

Mempelajari proses merancang, membangun, dan merenovasi gedung serta infrastruktur lain, seperti jalan, jembatan, bendungan, dan infrastruktur lainnya. Memahami unsur-unsur bangunan seperti beton, baja, aspal, dan lain-lain. Mempelajari perancangan struktur bangunan yang kuat, layak, dan efisien.

Terdapat satu peminatan pada program studi Teknik Sipil, yaitu :

- Teknik Sipil

### 2. Arsitektur

Mempelajari desain dan rancangan konstruksi bangunan. Lebih menuangkan ide, konsep, dan desain di atas kertas, sedangkan realisasi akan dikerjakan oleh teknik sipil. Harus mempelajari kekuatan bangunan (firmitasi), estetika atau keindahan bangunan (venustas), dan fungsi bangunan (utilitas).

Terdapat satu peminatan pada program studi Arsitektur, yaitu :

- Arsitektur

## **Fakultas Filsafah dan Peradaban**

Terdapat satu program studi pada Fakultas Filsafah dan Peradaban, yaitu : Ilmu Filsafat.

### 1. Ilmu Filsafat

Filsafat sebagai induk semua ilmu, filsafat lebih mempelajari tentang permasalahan mendasar manusia dan hubungannya dengan realita. Bersifat abstrak dan memerlukan pemahaman yang mendasar. Kajian utamanya yaitu tujuan hidup, esensi manusia, moralitas, dan hati nurani. Mempelajari pemikiran para filsuf. Membantu berpikir secara terstruktur dan mampu memproses informasi secara jernih.

Terdapat dua peminatan pada program studi Ilmu Filsafat, yaitu :

- Filsafat Keilahian
- Filsafat Budaya

### **Fakultas Teknologi Industri**

Terdapat tiga program studi pada Fakultas Teknologi Industri, yaitu : Teknik Industri, Teknik Kimia, dan Teknik Elektro.

#### **1. Teknik Industri**

Mempelajari proses industri baik dari sisi manajemen ataupun teknik. Turunan dari teknik mesin. Mempelajari disiplin ilmu lain seperti matematika, fisika, fisiologi, dan manajemen saintifik. Teknik Industri berfokus pada perancangan, peningkatan, dan pemasangan sistem terintegrasi yang membutuhkan manusia, material, peralatan, dan energi. Memiliki tiga bidang dan satu sistem manufaktur (mempelajari peningkatan kualitas, produktivitas, dan efisiensi sistem produk), dua manajemen industri (mempelajari manajemen keuangan, operasional, manajemen inovasi, perencanaan dan pengendalian produksi, dan ekonomi teknik), dan tiga sistem industri dan tekno ekonomi, seperti logistik, statistik, penelitian operasional, dan sistem basis data.

Terdapat satu peminatan pada program studi Teknik Industri, yaitu :

- Teknik Industri

#### **2. Teknik Kimia**

Cabang ilmu teknik yang mempelajari bagaimana proses dan cara mengubah bahan baku/mentah dan bahan kimia menjadi sebuah produk yang lebih bernilai secara komersial maupun perubahan sifat fisik dan kimia bahan mentah. Dididik untuk merencanakan dan merancang alat-alat proses, mengoperasikan, mengendalikan dan memelihara pabrik/industri, mengkontruksi pendirian suatu pabrik, mengadakan penelitian dan pengembangan proses, serta merencanakan serta mengelola penjualan dan pelayanan.

Terdapat satu peminatan pada program studi Teknik Kimia, yaitu :

- Teknik Kimia

#### **3. Teknik Elektro**

Mempelajari sifat-sifat elektron yang kita kenal sebagai listrik, mempelajari aplikasi dan pemanfaatan listrik dalam kehidupan sehari-hari, serta teknologi yang terkait. Cakupannya meliputi pembangkit tenaga listrik, sistem jaringan distribusi, pemanfaatan oleh pengguna akhir.

Terdapat satu peminatan pada program studi Teknik Elektro, yaitu :

- Mekatronika

### **Fakultas Teknologi Informasi dan Sains**

Terdapat tiga program studi pada Fakultas Teknologi Informasi dan Sains, yaitu : Matematika, Fisika, dan Teknik Informatika.

#### **1. Matematika**

Mempelajari matematika murni seperti aljabar, geometri, dan analisis matematika; statistika; komputasi; aktuarial; dan riset operasi.

Terdapat dua peminatan pada program studi Matematika, yaitu :

- Aktuaria
- Matematika Terapan

## 2. Fisika

Mempelajari gejala alam yang tidak hidup atau materi dalam lingkup ruang dan waktu, mempelajari perilaku dan sifat materi dalam bidang yang sangat beragam (partikel submikroskopis - perilaku materi alam semesta sebagai satu kesatuan kosmos). Ilmu fisika sangat mendukung perkembangan teknologi, yaitu industri, komunikasi, rekayasa, kimia, dan kedokteran.

Terdapat satu peminatan pada program studi Fisika, yaitu :

- Fisika

## 3. Teknik Informatika

Mempelajari dan menerapkan prinsip-prinsip ilmu komputer dan analisa matematis untuk desain, pengembangan, pengujian, evaluasi perangkat lunak, sistem operasi, dan kerja komputer. Menghasilkan ide kreatif, merealisasikan ide, mendiferensiasikan berbagai macam fungsi, dan menciptakan struktur instruksi yang sangat detail dalam bahasa pemrograman untuk mengajarkan komputer apa yang harus dilakukan.

Terdapat dua peminatan pada program studi Teknik Informatika, yaitu :

- *Data Science*
- *Computer Science*

### 2.4.2 Syarat Masuk Program Studi

Berikut merupakan syarat untuk program studi yang ada di Universitas Katolik Parahyangan :

Program Studi	Syarat Jurusan	USM	PMDK	Syarat Khusus
Ekonomi Pembangunan	IPA IPS	Matematika Bahasa Inggris	Matematika Bahasa Indonesia Bahasa Inggris	
Manajemen	IPA IPS	Matematika Bahasa Inggris	Matematika Bahasa Inggris	
Akuntansi	IPA IPS	Matematika Bahasa Inggris	Matematika Bahasa Inggris	
Ilmu Hukum	IPA IPS Bahasa	Matematika Bahasa Inggris	Matematika Bahasa Inggris Pendidikan Kewarganegaraan	
Ilmu Administrasi Publik	IPA IPS Bahasa SMK	Matematika Bahasa Inggris	Matematika Bahasa Inggris	
Ilmu Administrasi Bisnis	IPA IPS Bahasa SMK	Matematika Bahasa Inggris	Matematika Bahasa Inggris	

Ilmu Hubungan Internasional	IPA IPS Bahasa	Matematika Bahasa Inggris	Matematika Bahasa Inggris Uraian Bahasa Inggris	
Teknik Sipil	IPA	Matematika Bahasa Inggris Fisika	Matematika Bahasa Inggris Fisika	
Arsitektur	IPA	Matematika Bahasa Inggris Gambar	Matematika Bahasa Inggris Gambar	
Ilmu Filsafat	IPA IPS Bahasa SMK	Matematika Bahasa Inggris Wawancara	Bahasa Inggris Bahasa Indonesia	
Teknik Industri	IPA	Matematika Matematika	Bahasa Inggris Bahasa Inggris	
Teknik Kimia	IPA	Matematika Bahasa Inggris Fisika	Matematika Bahasa Inggris Fisika Kimia	Tidak buta warna
Teknik Elektro	IPA	Matematika Bahasa Inggris Fisika	Matematika Bahasa Inggris Fisika	Tidak buta warna
Matematika	IPA	Matematika Bahasa Inggris	Matematika Bahasa Inggris	
Fisika	IPA	Matematika Bahasa Inggris Fisika	Matematika Bahasa Inggris	
Teknik Informatika	IPA	Matematika Bahasa Inggris	Matematika Bahasa Inggris	

Tabel 2.1: Tabel syarat program studi

### 2.4.3 Karakteristik Program Studi

Berikut merupakan kriteria untuk calon mahasiswa sesuai dengan program studi :

Program Studi	Karakteristik
Ekonomi Pembangunan	Tertarik dengan Ilmu Ekonomi Tertarik dengan perhitungan Berpikir kritis Senang menganalisis Mampu memecahkan masalah
Manajemen	Keterampilan komunikasi Senang menganalisis Senang memecahkan masalah
Akuntansi	Tertarik dengan akuntansi Memiliki kemampuan berhitung yang kuat dan teliti Senang menganalisis

Ilmu Hukum	Tertarik dengan hukum Teliti dan berpikir kritis Keterampilan komunikasi Mampuan menganalisis
Ilmu Administrasi Publik	Terstruktur Senang menganalisis Senang memecahkan masalah
Ilmu Administrasi Bisnis	Memiliki minat yang tinggi untuk usaha Kemampuan komunikasi Kemampuan berhitung Terstruktur
Ilmu Hubungan Internasional	Tertarik dengan interaksi internasional Kemampuan berbahasa Inggris Berwawasan luas Kemampuan komunikasi
Teknik Sipil	Senang berhitung Terstruktur
Arsitektur	Tertarik dengan desain dan rancangan bangunan Tertarik dengan menggambar dan seni Tertarik dengan humaniora, sains, dan teknologi
Ilmu Filsafat	Tipe pemikir Berwawasan luas Berpikir Rasional Berpikir kritis
Teknik Industri	Senang berhitung Terstruktur
Teknik Kimia	Tertarik dengan Kimia Senang berhitung Terstruktur Tidak buta warna
Teknik Elektro	Tidak buta warna Senang berhitung Terstruktur Teliti
Matematika	Tertarik dengan Matematika Senang memecahkan masalah Terstruktur Teliti
Fisika	Senang berhitung Senang menganalisis Mampu memecahkan masalah Teliti
Teknik Informatika	Tertarik dengan teknologi Senang menganalisis Senang memecahkan masalah Senang berhitung

Tabel 2.2: Tabel kriteria



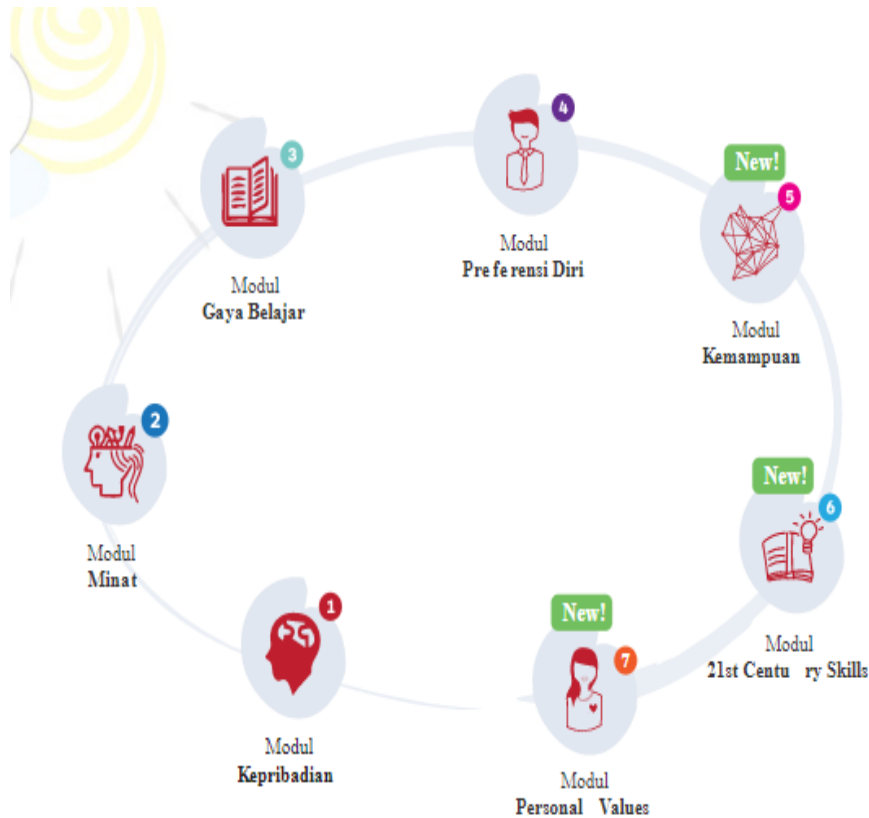


## BAB 3

### ANALISIS

#### 3.1 Analisis Perangkat Lunak Sejenis

Salah satu *website* yang dapat memberikan rekomendasi program studi adalah <https://rencanamu.id>. Sistem tersebut dikembangkan menggunakan riset ilmiah, Rencanamu mengukur 7 dimensi profil siswa sebagai landasan dalam rekomendasi, perencanaan kuliah dan karier yang terintegrasi, berkesinambungan dan menyeluruh. Gambar 3.1 menunjukkan 7 dimensi profil siswa.

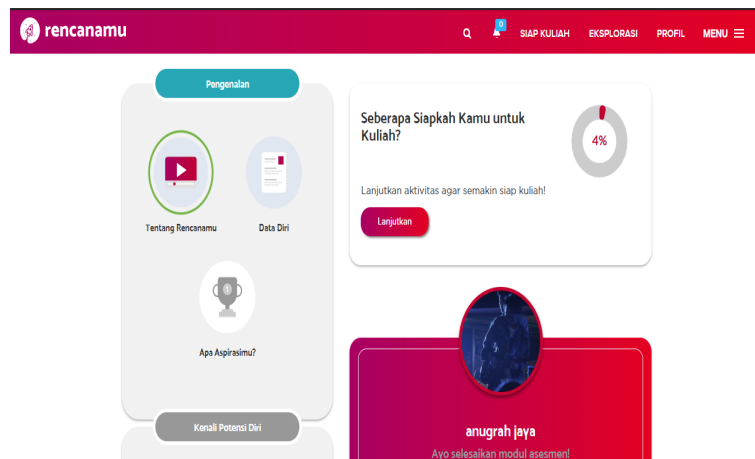


Gambar 3.1: 7 Dimensi Profil Siswa

Pada sistem ini, telah dilakukan beberapa analisis dan hasilnya sebagai berikut :

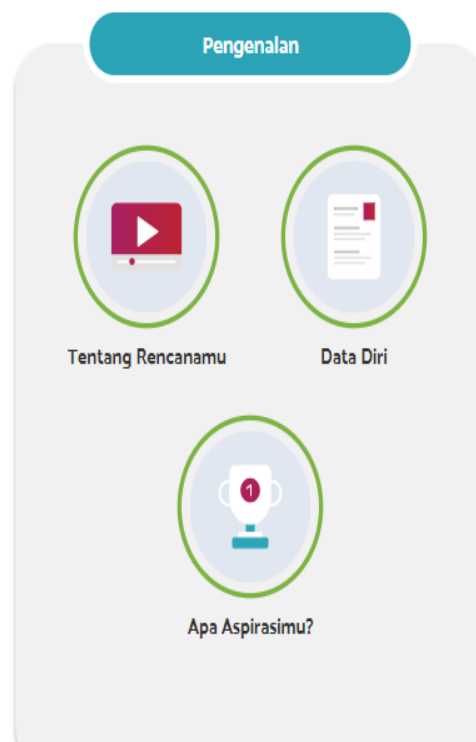
1. Website <https://rencanamu.id> adalah sebuah *platform* persiapan kuliah dan karier *online* berbasis data didukung oleh teknologi *People Science* untuk membantu siswa dalam merancang dan mempersiapkan masa depan mereka.
2. Perlu melakukan registrasi atau *login* kedalam sistem.

3. Gambar 3.2 merupakan tampilan awal setelah registrasi atau *login*.

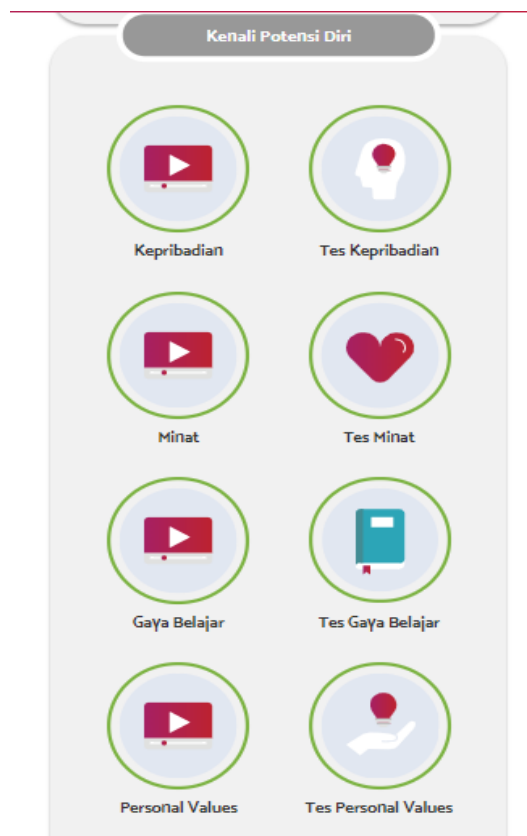


Gambar 3.2: Tampilan setelah registrasi atau *login*

4. Gambar 3.3, Gambar 3.4, dan Gambar 3.5 adalah beberapa modul yang harus dikerjakan.



Gambar 3.3: Modul Pengenalan

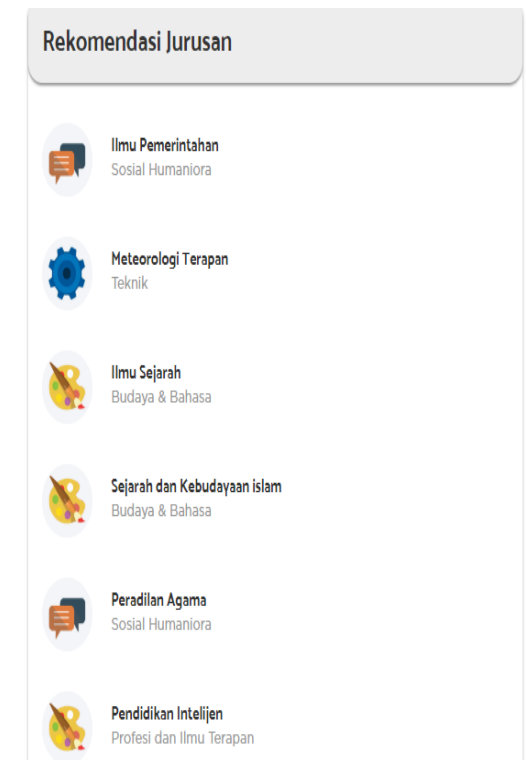


Gambar 3.4: Modul Potensi Diri



Gambar 3.5: Modul Ukur Kemampuan Diri

5. Gambar 3.6 adalah contoh hasil rekomendasi yang diberikan sistem berdasarkan modul yang sudah dikerjakan.



Gambar 3.6: Hasil Rekomendasi

Website <https://rencanamu.id> memiliki kesamaan dengan sistem yang dibangun yaitu memberikan rekomendasi program studi untuk anak SMA. Perbedaannya pada website <https://rencanamu.id> tidak menampilkan prediksi IPK dan harus mengisi beberapa modul untuk mendapatkan rekomendasi program studi.

### 3.2 Pemilihan Algoritma Sistem Rekomendasi

Berdasarkan teori 2.1.3 yang menjelaskan mengenai teknik-teknik yang dapat digunakan untuk membangun sistem rekomendasi, teknik *collaborative filtering* adalah teknik yang dapat digunakan untuk memberikan rekomendasi berupa program studi kepada calon mahasiswa berdasarkan kesamaan dengan pengguna lain. Berikut merupakan beberapa hal mengapa memilih teknik *collaborative filtering* :

1. *Collaborative filtering* menghasilkan rekomendasi item yang spesifik untuk pengguna berdasarkan peringkat tanpa memerlukan informasi tambahan mengenai item ataupun pengguna.

Teknik *collaborative filtering* memiliki beberapa kekurangan diantaranya :

1. Rekomendasi yang diberikan mengambil data yang cukup banyak dari basis data sehingga membutuhkan memori yang besar.
2. Tidak bisa memberikan rekomendasi untuk item yang tidak pernah diberikan *rating* oleh pengguna.
3. Pengguna harus memberikan *rating* untuk beberapa atribut agar bisa diberikan rekomendasi.

Pada sistem yang dibangun, akan memberikan rekomendasi berdasarkan nilai raport beberapa mata pelajaran siswa pada kelas 10 dan 11 yang digunakan untuk PMDK. Mata pelajaran yang digunakan adalah Matematika, Bahasa Indonesia, Bahasa Inggris, Fisika, Kimia, dan Pendidikan Kewarganegaraan. Rekomendasi program studi berdasarkan asal jurusan saat SMA, misalnya siswa IPA akan diberikan rekomendasi program studi IPA.

### 3.3 *Preprocessing* Data Mahasiswa

Data mahasiswa yang digunakan adalah data mahasiswa Universitas Katolik Parahyangan dengan jalur penerimaan Penelurusan dan Kemampuan atau PMDK pada tahun 2013-2018. Pada data yang digunakan terdapat beberapa atribut yang tidak dapat digunakan seperti No.PMB, kota asal sekolah, dan provinsi asal sekolah. Atribut yang tidak dapat digunakan akan dihapus dan data akan dipisahkan menjadi dua *file* mahasiswa dan nilai untuk setiap program studi yang ada. *preprocessing* dilakukan menggunakan Python. Berikut langkah-langkah dalam *preprocessing* :

1. Membaca file .csv yang berisikan data mahasiswa pada fakultas tertentu.
2. Membuat dataframe untuk menampung data mahasiswa dan nilai.
3. Menginisialisasikan batas *looping*, *id\_user*, *id\_nilai*, dan asal jurusan.
4. Menambahkan data mahasiswa berupa NPM, *id\_prodi*, asal jurusan, dan IPK pada dataframe mahasiswa.
5. Mengubah *range* nilai menjadi GPA (*Grade Point Average*) dan menghitung nilai rata-rata untuk setiap nilai mata pelajaran.
6. Menambahkan data GPA, rata-rata nilai, dan *id\_user* pada dataframe nilai.
7. Menyimpan dataframe mahasiswa dan nilai menjadi .csv.

Hasil *file* .csv nantinya akan *diimport* pada basis data yang akan digunakan pada sistem.

### 3.4 Contoh Perhitungan Pearson Correlation

Berdasarkan 2.1.4 terdapat langkah-langkah perhitungan dari *user-based*. Tabel 3.1 merupakan contoh data mahasiswa.

MP/Semester	101	102	111	112	AVG
Matematika	2.6	2.9	2.95	2.75	2.8
Bahasa Indonesia	0	0	0	0	0
Bahasa Inggris	2.95	3	2.85	2.95	2.9375
PKN	0	0	0	0	0

Tabel 3.1: Contoh data mahasiswa dalam bentuk GPA

Berikut merupakan contoh langkah-langkah perhitungan :

1. Menghitung nilai rata-rata *rating*.

MP/Semester	101	102	111	112	Rumus	AVG
Matematika	2.9	3.4	3.4	2.9	$\frac{2.9+3.4+3.4+2.9}{4}$	3.15
Bahasa Indonesia	2.95	2.9	3.9	3.4	$\frac{2.95+2.9+3.9+3.4}{4}$	3.2875
Bahasa Inggris	3.3	3.35	3.25	2.9	$\frac{3.3+3.35+3.25+2.9}{4}$	3.2
PKN	3.4	2.9	3.35	2.35	$\frac{3.4+2.9+3.35+2.35}{4}$	3

Tabel 3.2: Contoh data siswa dalam bentuk GPA

## 2. Menghitung kesamaan atau similaritas.

No	Rumus	Matematika	Rumus	Bahasa Inggris
1	$(2.9 - 3.15)*$ $(2.6 - 2.8)$	0.5	$(3.3 - 3.2)*$ $(2.95 - 2.9375)$	0.00125
2	$(3.4 - 3.15)*$ $(2.9 - 2.8)$	0.025	$(3.35 - 3.2)*$ $(3 - 2.9375)$	0.009375
3	$(3.4 - 3.15)*$ $(2.95 - 2.8)$	0.0375	$(3.25 - 3.2)*$ $(2.85 - 2.9375)$	-0.004375
4	$(2.9 - 3.15)*$ $(2.75 - 2.8)$	0.0125	$(2.9 - 3.2)*$ $(2.95 - 2.9375)$	-0.00375
Sigma	$0.5 + 0.025 +$ $0.0375 + 0.0125$	0.125	$0.00125 + 0.009375 +$ $-0.004375 + -0.00375$	0.0025
Hasil			$0.125 + 0.0025$	0.1274

Tabel 3.3: Nilai kovariasi mahasiswa dan siswa

No	Rumus	Matematika	Rumus	Bahasa Inggris
1	$(2.9 - 3.15)^2$	0.0625	$(3.3 - 3.2)^2$	0.01
2	$(3.4 - 3.15)^2$	0.0625	$(3.35 - 3.2)^2$	0.0225
3	$(3.4 - 3.15)^2$	0.0625	$(3.25 - 3.2)^2$	0.0025
4	$(2.9 - 3.15)^2$	0.0625	$(2.9 - 3.2)^2$	0.09
Sigma	$0.0625+0.0625+$ $0.0625+0.0625$	0.25	$0.01+0.0225+$ $0.0025+0.09$	0.125
Hasil			$\sqrt{0.25 + 0.125}$	0.612372436

Tabel 3.4: Standar Deviasi Siswa

No	Rumus	Matematika	Rumus	Bahasa Inggris
1	$(2.6 - 2.8)^2$	0.04	$(2.95 - 2.9375)^2$	0.00015625
2	$(2.9 - 2.8)^2$	0.01	$(3 - 2.9375)^2$	0.00390625
3	$(2.95 - 2.8)^2$	0.0225	$(2.85 - 2.9375)^2$	0.00765625
4	$(2.75 - 2.8)^2$	0.0225	$(2.95 - 2.9375)^2$	0.00015625
Sigma	0.04+0.01+ 0.0225+0.0225	0.075	0.0.000156250.00390625+ 0.00765625+0.00015625	0.011875
Hasil			$\sqrt{0.075 + 0.011875}$	0.294745653

Tabel 3.5: Standar Deviasi Mahasiswa

No	Rumus	Kesamaan	IPK
1	$\frac{0.1275}{0.612372436 \cdot 0.294745653}$	0.706394228	3.11
2	$\frac{0.0125}{0.612372436 \cdot 0.2343242}$	0.08711185	2.9
3	$\frac{0.2}{0.612372436 \cdot 0.543242}$	0.601202838	3
4	$\frac{0.125}{0.612372436 \cdot 0.432343}$	0.472134729	3.2
5	$\frac{0.05}{0.612372436 \cdot 0.242345}$	0.336914969	3.4

Tabel 3.6: Contoh Perhitungan kesamaan

- Memilih nilai kesamaan atau similaritas yang bernilai lebih dari 0.  
Berdasarkan langkah nomor 2, maka nilai kesamaan pada tabel 3.6 semuanya dapat digunakan untuk prediksi.
- Menghitung nilai prediksi.

No	Kesamaan	Rumus	Kesamaan*IPK
1	0.706394228	$0.706394228 \cdot 3.11$	2.196886049
2	0.08711185	$0.08711185 \cdot 2.9$	0.252624364
3	0.601202838	$0.601202838 \cdot 3$	1.803608514
4	0.472134729	$0.472134729 \cdot 3.2$	1.510831133
5	0.336914969	$0.336914969 \cdot 3.4$	1.145510893
Sigma	2.203758614	-	6.909460954
Hasil		$\frac{6.909460954}{2.203758614}$	3.13530752

Tabel 3.7: Contoh hasil Prediksi

### 3.5 Contoh Perhitungan Metode Evaluasi Sistem Rekomendasi

Berdasarkan penjelasan mengenai sistem rekomendasi yang dibahas pada bab 2 bagian 2.1.5. Salah satu bagian terpenting adalah evaluasi. Evaluasi pada sistem rekomendasi dilakukan untuk

mendapatkan akurasi dari hasil prediksi yang diberikan. Akurasi merupakan salah satu aspek yang sering dijadikan acuan untuk rekomendasi yang digunakan. Dalam melakukan pengujian akurasi bisa menggunakan metode *Mean Absolute Error* (MAE) dan *Root Mean Square Error* (RMSE). Berikut merupakan contoh penerapan kedua metode yang akan disajikan dalam tabel ?? dan tabel ?? :

No	Item	Pengguna	Penilaian Asli $r_{u,i}$	Prediksi Sistem $\hat{r}_{u,i}$	$r_{u,i} - \hat{r}_{u,i}$	$ r_{u,i} - \hat{r}_{u,i} $	$(r_{u,i} - \hat{r}_{u,i})^2$
1	110	1	3.4	3.5	-0.1	0.1	0.01
2	120	30	3.3	3.1	0.2	0.2	0.04
3	130	56	3	2.8	0.2	0.2	0.04
4	200	65	2.9	3.1	-0.2	0.2	0.04
5	310	76	3.1	2.8	0.3	0.3	0.09
6	320	87	3.2	3	0.2	0.2	0.04
7	330	99	2.8	3	-0.2	0.2	0.04
8	410	102	3.4	3.5	-0.1	0.1	0.01
9	420	167	3.1	3.5	-0.4	0.4	0.16
10	510	189	2.8	3.1	-0.3	0.3	0.09
11	610	298	3.1	2.9	0.2	0.2	0.04
12	620	344	3.4	2.9	0.5	0.5	0.25
13	630	365	3.1	3	0.1	0.1	0.01
14	710	465	2.9	3	-0.1	0.1	0.01
15	720	477	3.4	3.5	-0.1	0.1	0.01
16	730	480	3.4	3.6	-0.2	0.2	0.04
Jumlah						3.4	0.92

Tabel 3.8: Tabel Data MAE dan RMSE

Berdasarkan data pada tabel 3.8 jika dihitung dengan persamaan MAE yaitu :

$$MAE = \frac{1}{n} * \sum |r_{u,i} - \hat{r}_{u,i}| \quad (3.1)$$

maka akan didapatkan hasil pengimplementasian dari rumus MAE sebagai berikut :

$$MAE = \frac{1}{16} * 3.4 = 0.2125 \quad (3.2)$$

Berdasarkan data pada tabel 3.8 jika dihitung dengan persamaan RMSE yaitu :

$$RMSE = \sqrt{\frac{1}{n} * \sum (r_{u,i} - \hat{r}_{u,i})^2} \quad (3.3)$$

maka akan didapatkan hasil pengimplementasian dari rumus RMSE sebagai berikut :

$$RMSE = \sqrt{\frac{1}{16} * 0.91} = 0.0575 \quad (3.4)$$

### 3.6 Analisis Kebutuhan Sistem

Pada sistem yang akan dibangun, memiliki kebutuhan-kebutuhan perangkat lunak seperti : *Use Case* dan Rancangan Basis Data.



### 3.6.1 Diagram *Use Case*

Pada sistem yang akan dibangun terdapat satu aktor yaitu Siswa/i. Siswa/i ini adalah calon mahasiswa kelas XI yang merupakan target dari sistem yang akan dibangun. Terdapat 4 langkah yang harus dilakukan, yaitu :

#### 1. Pendefinisian Aktor

No	Aktor	Deskripsi
1	Siswa/i	Siswa/i adalah orang yang akan diberikan rekomendasi program studi yang ada di Universitas Parahyangan.

Tabel 3.9: Pendefinisian Aktor

#### 2. Pendefinisian *Use Case*

No	<i>Use Case</i>	Deskripsi
1	Memilih Jurusan SMA	Merupakan proses untuk memilih jurusan saat SMA.
2	Mengisi Nilai Rapor	Merupakan proses untuk mengisi nilai beberapa nilai mata pelajaran sesuai dengan jurusan saat SMA.

Tabel 3.10: Pendefinisian *Use Case*

#### 3. Pembuatan *Use Case* Skenario

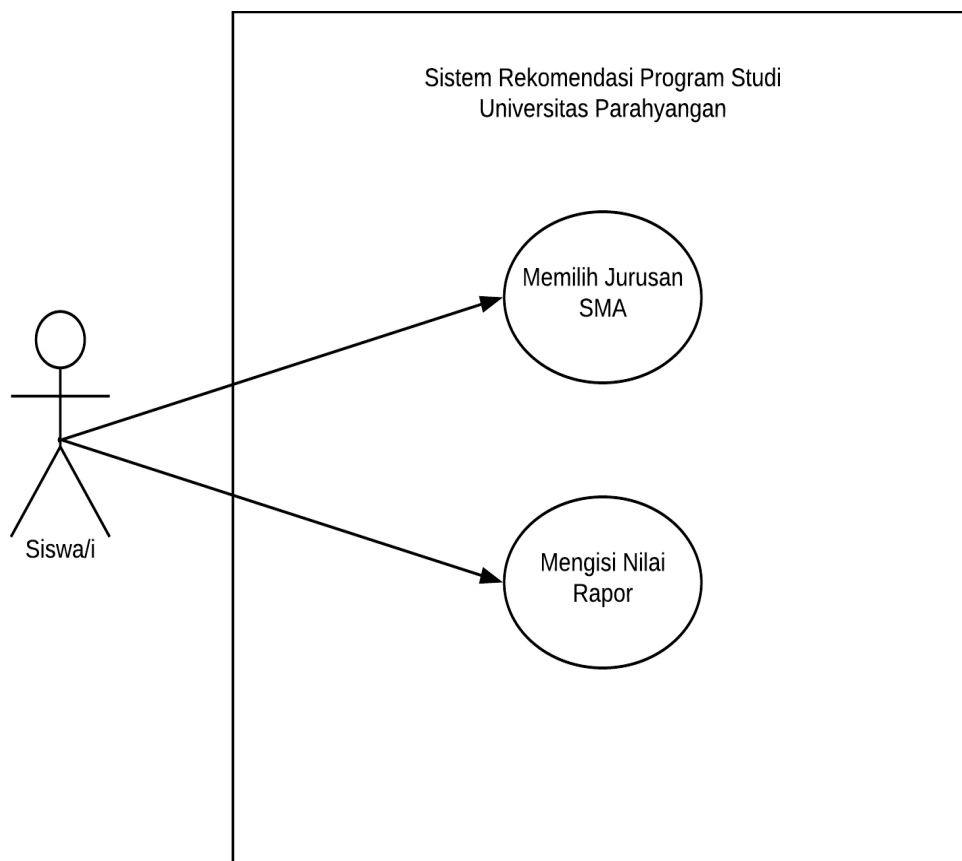
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Memilih jurusan saat SMA.	
	2. Mengarahkan kepada form sesuai jurusan SMA.

Tabel 3.11: Skenario Memilih Jurusan SMA

Skenario Normal	
1. Mengisi nilai sesuai nilai rapor.	
	2. Memeriksa valid tidaknya data yang dimasukkan.
	3. Memeriksa range nilai.
4. Klik tombol submit.	
	5. Mengarahkan kepada halaman hasil rekomendasi.
Skenario Alternatif	
1. Mengisi nilai yang tidak valid.	
	2. Memeriksa valid tidaknya data yang dimasukkan.
	3. Memberikan pesan data tidak valid.
4. Mengisi nilai sesuai nilai rapor yang valid.	
	5. Memeriksa <i>range</i> nilai.

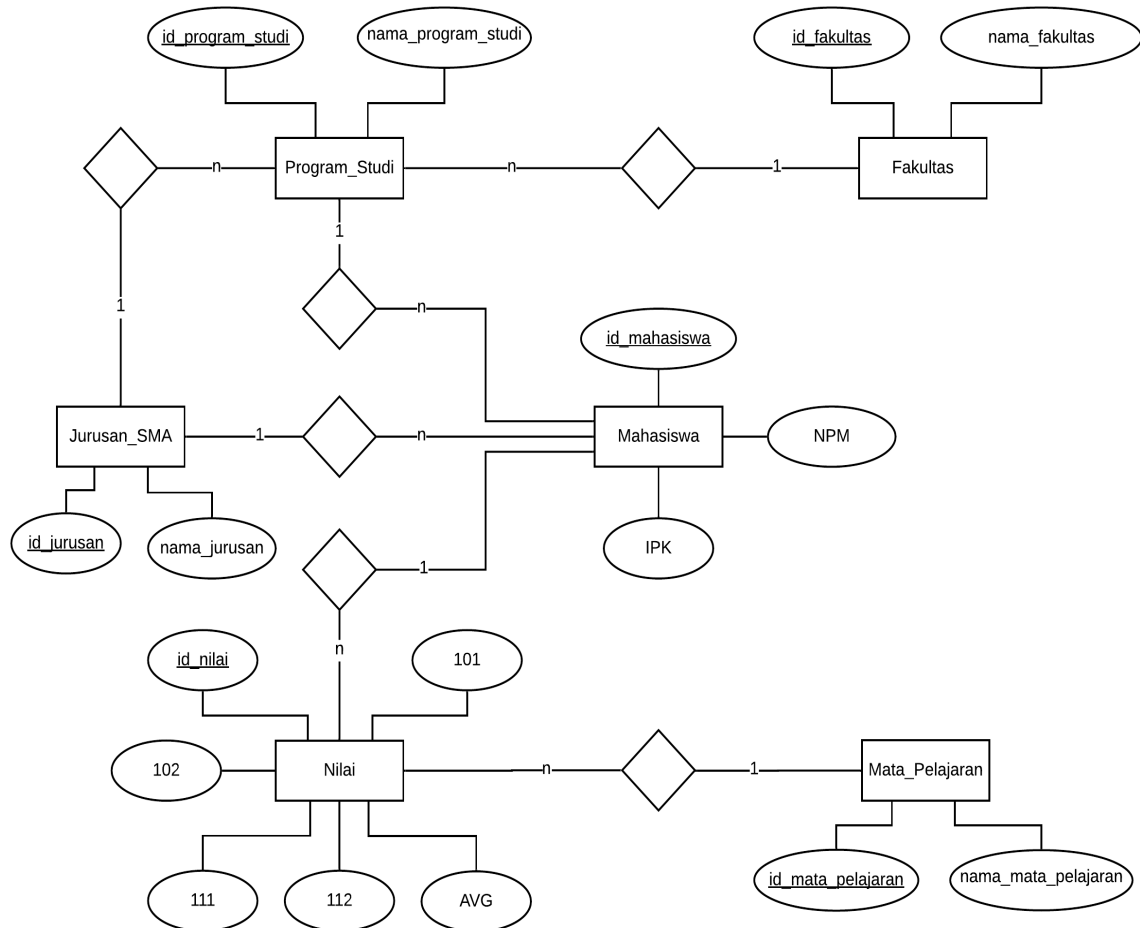
	6. Memberikan pesan <i>range</i> tidak sesuai.
7. Mengisi nilai sesuai <i>range</i> .	
	8. Memeriksa <i>range</i> nilai.
9. Klik tombol submit.	
	10. Mengarahkan kepada <i>page</i> hasil rekomendasi.

Tabel 3.12: Skenario Mengisi Nilai Rapor

4. Menggambarkan Diagram *Use Case*Gambar 3.7: Diagram *Use Case* Sistem Rekomendasi

### 3.6.2 Rancangan Basis Data

#### Diagram ERD



Gambar 3.8: Diagram ERD Sistem Rekomendasi

Berikut merupakan entitas dan atribut gambar 3.8 yang akan digunakan pada sistem yang akan dibangun :

1. Jurusan\_SMA memiliki atribut id\_jurusan dan nama jurusan.
2. Fakultas memiliki atribut id\_fakultas dan nama\_fakultas.
3. Program\_Studi memiliki atribut id\_program\_studi dan nama\_program\_studi.
4. Mahasiswa memiliki atribut id\_mahasiswa, NPM, dan IPK.
5. Mata\_Pelajaran memiliki atribut id\_mata\_pelajaran dan nama\_mata\_pelajaran.
6. Nilai memiliki atribut id\_nilai, 101, 102, 111, 112, dan AVG.



## BAB 4

### PERANCANGAN

#### 4.1 Perancangan Fisik Basis Data

##### 4.1.1 Perancangan Tabel

Tabel data yang akan digunakan untuk sistem rekomendasi program studi Universitas Katolik Parahyangan sesuai diagram ERD pada gambar 3.8 akan dirancang sesuai pada tabel beriku :

###### 1. Jurusan SMA

No	Nama Atribut	Tipe Data	Keterangan
1	id_jurusan	INT	NOT NULL, Primary Key
2	nama_jurusan	VARCHAR(25)	NOT NULL

Tabel 4.1: Perancangan Tabel jurusan\_sma

###### 2. Fakultas

No	Nama Atribut	Tipe Data	Keterangan
1	id_fakultas	INT	NOT NULL, Primary Key
2	nama_fakultas	VARCHAR(50)	NOT NULL

Tabel 4.2: Perancangan Tabel fakultas

###### 3. Program Studi

No	Nama Atribut	Tipe Data	Keterangan
1	id_program_studi	INT	NOT NULL, Primary Key
2	nama_program_studi	VARCHAR(50)	NOT NULL
3	id_fakultas	INT	NOT NULL, Foreign Key dari fakultas
4	id_jurusan	INT	NOT NULL, Foreign Key dari jurusan_sma

Tabel 4.3: Perancangan Tabel program\_studi

###### 4. Mahasiswa

No	Nama Atribut	Tipe Data	Keterangan
1	id_mahasiswa	INT	NOT NULL, Primary Key
2	NPM	VARCHAR(10)	NOT NULL
3	IPK	DOUBLE	NOT NULL
4	id_jurusan	INT	NOT NULL, Foreign Key dari jurusan
5	id_program_studi	INT	NOT NULL, Foreign Key dari program_studi

Tabel 4.4: Perancangan Tabel mahasiswa

## 5. Mata Pelajaran

No	Nama Atribut	Tipe Data	Keterangan
1	id_mata_pelajaran	INT	NOT NULL, Primary Key
2	nama_mata_pelajaran	VARCHAR(20)	NOT NULL

Tabel 4.5: Perancangan Tabel mata\_pelajaran

## 6. Nilai

No	Nama Atribut	Tipe Data	Keterangan
1	id_nilai	INT	NOT NULL, Primary Key
2	id_mata_pelajaran	INT	Foreign Key dari mata_pelajaran
3	id_mahasiswa	INT	NOT NULL, Foreign Key dari mahasiswa
4	101	DOUBLE	Nilai kelas 10 semester 1
5	102	DOUBLE	Nilai kelas 10 semester 2
6	111	DOUBLE	Nilai kelas 11 semester 1
7	112	DOUBLE	Nilai kelas 11 semester 2
8	AVG	DOUBLE	Rata-rata nilai

Tabel 4.6: Perancangan Tabel nilai

## 4.2 Perancangan Algoritma

Pada subbab ini akan berisikan perancangan algoritma yang digunakan pada sistem. Sistem menggunakan beberapa algoritma seperti K-Means untuk membuat kelompok mahasiswa yang memiliki karakteriksi yang sama dengan calon mahasiswa, Pearson Correlation Coefficient untuk menghitung kesamaan atau similaritas, dan User-base Collaborative Filtering untuk menghitung prediksi nilai IPK.

### 4.2.1 K-Means

---

**Algorithm 1** test
 

---

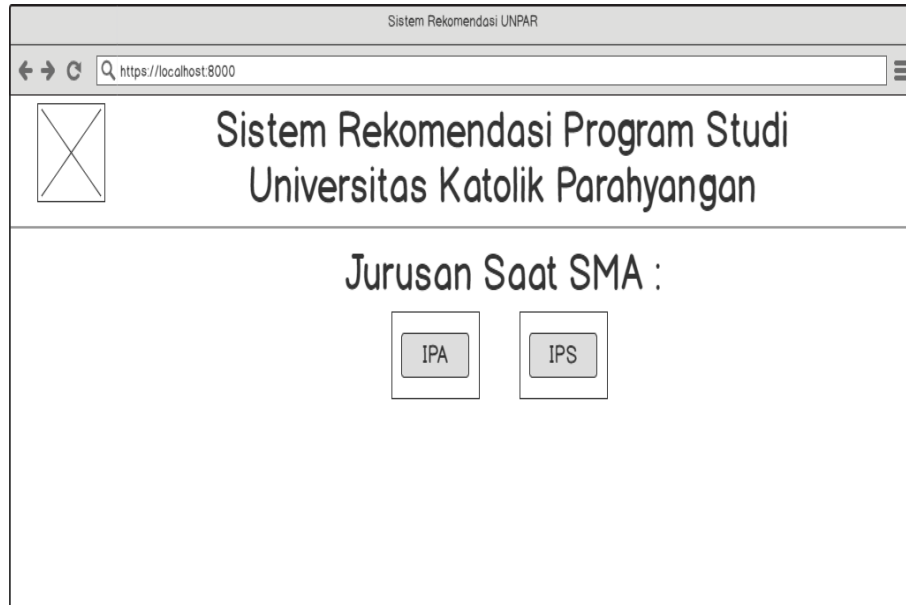
#### 4.2.2 Pearson Correlation Coefficient

#### 4.2.3 User-based Collaborative Filtering

### 4.3 Perancangan Antar Muka

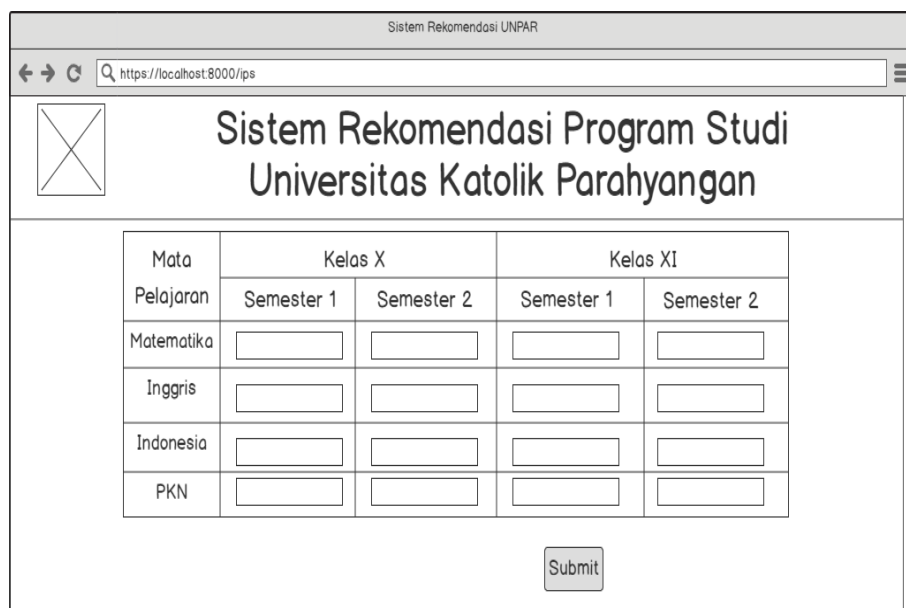
Pada subbab ini akan berisikan perancangan antar muka untuk sistem rekomendasi, berikut merupakan hasil perancangan :

1. Halaman index saat siswa/i mengakses sistem



Gambar 4.1: Halaman Index Sistem

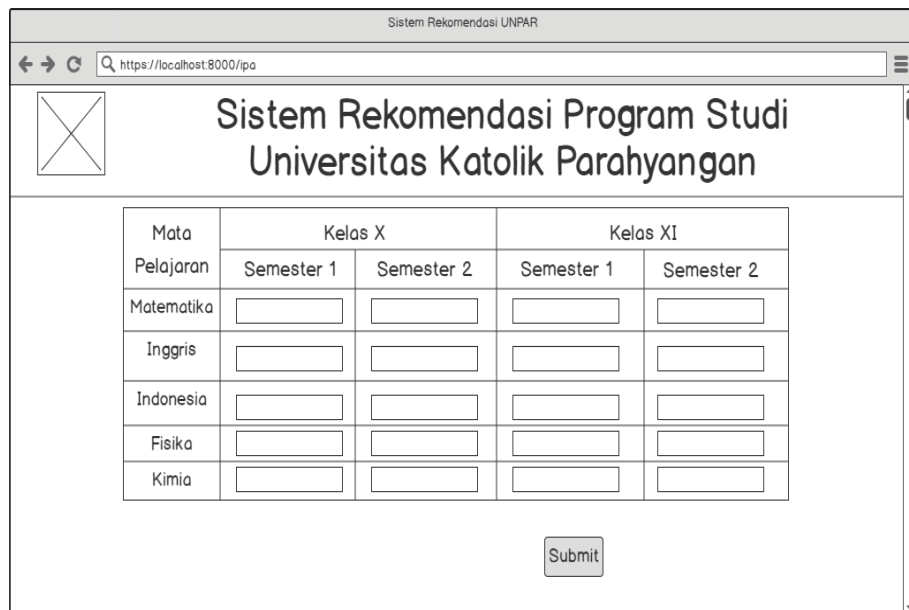
2. Halaman pengisian nilai siswa/i IPA



Mata Pelajaran	Kelas X		Kelas XI	
	Semester 1	Semester 2	Semester 1	Semester 2
Matematika	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Inggris	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Indonesia	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
PKN	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Gambar 4.2: Halaman Pengisian Nilai IPA

## 3. Halaman pengisian nilai siswa/i IPS



Mata Pelajaran	Kelas X		Kelas XI	
	Semester 1	Semester 2	Semester 1	Semester 2
Matematika	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Inggris	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Indonesia	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Fisika	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Kimia	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Gambar 4.3: Halaman Pengisian Nilai IPS

## 4. Halaman hasil rekomendasi



No	Fakultas	Program Studi	IPK
1	Teknologi Informasi dan Sains	Matematika	3.6
2	Teknologi Informasi dan Sains	Teknik Informatika	3.4
3	Fakultas Teknik	Teknik Sipil	3.3
4	Fakultas Teknik	Arsitektur	3.2
5	Fakultas Teknik Industri	Teknik Industri	3

Gambar 4.4: Halaman Hasil Rekomendasi



## BAB 5

### IMPLEMENTASI DAN PENGUJIAN

#### 5.1 Lingkungan Implementasi

Pada subbab ini akan dipaparkan perangkat keras dan perangkat lunak yang digunakan dalam membangun sistem rekomendasi program studi Universitas Katolik Parahyangan.

##### 5.1.1 Lingkungan Perangkat Keras

1. Processor : Intel Core i5-7200
2. Memory : 12 GB
3. Harddisk : 1 T
4. VGA : NVIDIA GeForce 940MX

#### 5.2 Lingkungan Perangkat Lunak

1. Web Server : Apache 2.4.41
2. Tools : XAMPP 3.2.4 dan Visual Studio Code 1.44
3. Bahasa Pemrograman : PHP 7.4.1
4. Database management system : MySQL
5. Operating System : Windows 10

#### 5.3 Implementasi Tabel Basis Data

Dibawah ini merupakan implementasi tabel basis data yang digunakan pada sistem rekomendasi program studi Universitas Katolik Parahyangan.

##### 1. Tabel Jurusan SMA

Tabel jurusan SMA digunakan untuk menyimpan seluruh data jurusan SMA yang digunakan pada sistem rekomendasi.

Listing 5.1: Implementasi tabel jurusan SMA

```
1 CREATE TABLE 'jurusan_sma' (  
2     'id_jurusan' int(10) UNSIGNED NOT NULL,  
3     'nama_jurusan' varchar(25) COLLATE utf8mb4_unicode_ci NOT NULL  
4 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

## 2. Tabel Fakultas

Tabel fakultas digunakan untuk menyimpan seluruh fakultas yang ada di Universitas Katolik Parahyangan.

Listing 5.2: Implementasi tabel

```
1 CREATE TABLE 'fakultas' (
2     'id_fakultas' INT(10) UNSIGNED NOT NULL,
3     'nama_fakultas' VARCHAR(50) COLLATE utf8mb4_unicode_ci NOT NULL
4 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

## 3. Program Studi

Tabel program studi digunakan untuk menyimpan seluruh program studi yang ada di Universitas Katolik Parahyangan.

Listing 5.3: Implementasi tabel

```
1 CREATE TABLE 'program_studi' (
2     'id_program_studi' INT(10) UNSIGNED NOT NULL,
3     'nama_program_studi' VARCHAR(50) COLLATE utf8mb4_unicode_ci NOT NULL,
4     'id_fakultas' INT(10) UNSIGNED NOT NULL,
5     'id_jurusan' INT(10) UNSIGNED NOT NULL
6 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

## 4. Mahasiswa

Tabel mahasiswa digunakan untuk menyimpan nilai seluruh mahasiswa yang sudah lulus dari Universitas Katolik Parahyangan.

Listing 5.4: Implementasi tabel

```
1 CREATE TABLE 'mahasiswa' (
2     'id_mahasiswa' INT(10) UNSIGNED NOT NULL,
3     'NPM' VARCHAR(10) COLLATE utf8mb4_unicode_ci NOT NULL,
4     'IPK' DOUBLE(3,2) NOT NULL,
5     'id_jurusan' INT(10) UNSIGNED NOT NULL,
6     'id_program_studi' INT(10) UNSIGNED NOT NULL
7 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

## 5. Mata Pelajaran

Tabel mata pelajaran digunakan untuk menyimpan mata pelajaran yang digunakan pada PMDK di Universitas Katolik Parahyangan.

Listing 5.5: Implementasi tabel

```
1 CREATE TABLE 'mata_pelajaran' (
2     'id_mata_pelajaran' INT(10) UNSIGNED NOT NULL,
3     'nama_mata_pelajaran' VARCHAR(20) COLLATE utf8mb4_unicode_ci NOT NULL
4 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

## 6. Nilai

Tabel nilai digunakan untuk menyimpan nilai mahasiswa pada kelas X dan XI untuk semester 1 dan 2 pada saat SMA.

Listing 5.6: Implementasi tabel

```
1 CREATE TABLE 'nilai' (
2     'id_nilai' INT(10) UNSIGNED NOT NULL,
```

```
3      'id_mata_pelajaran' int(10) UNSIGNED NOT NULL,  
4      'id_mahasiswa' int(10) UNSIGNED NOT NULL,  
5      '101' double(5,2) NOT NULL,  
6      '102' double(5,2) NOT NULL,  
7      '111' double(5,2) NOT NULL,  
8      '112' double(5,2) NOT NULL,  
9      'AVG' double(5,2) NOT NULL  
10 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

## 5.4 Impelementasi Antar Muka

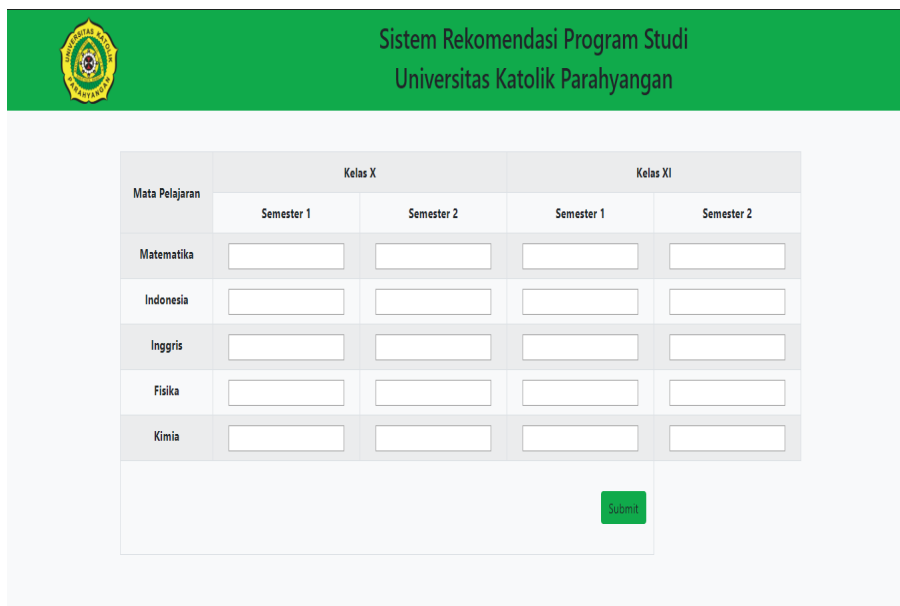
Pada subab ini akan ditampilkan antar muka yang digunakan pada sistem rekomendasi program studi Universitas Katolik Parahyangan.

1. Halaman index saat siswa/i mengakses sistem



Gambar 5.1: Halaman Index Sistem

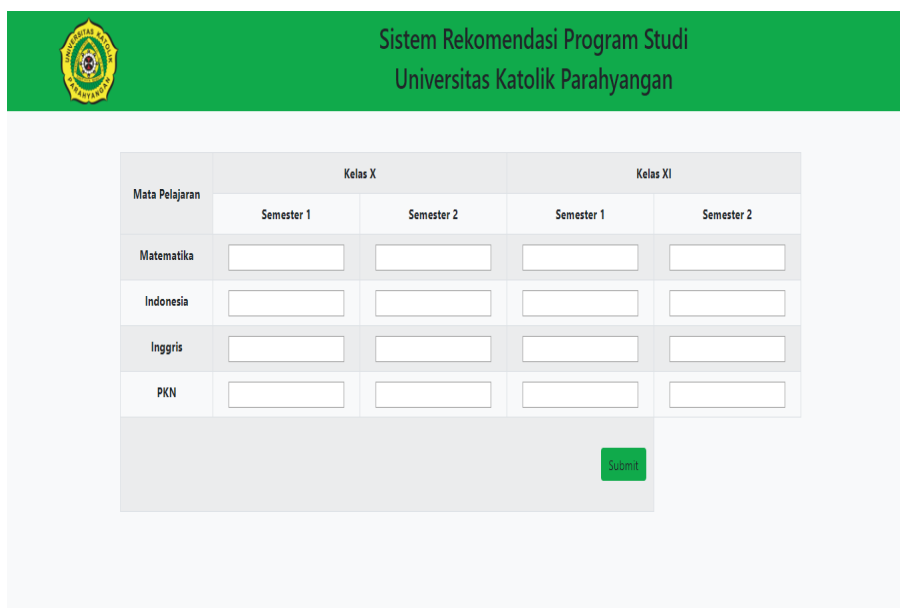
2. Halaman pengisian nilai siswa/i IPA



Mata Pelajaran	Kelas X		Kelas XI	
	Semester 1	Semester 2	Semester 1	Semester 2
Matematika	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Indonesia	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Inggris	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Fisika	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Kimia	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Gambar 5.2: Halaman Index Pengisian Nilai IPA

## 3. Halaman pengisian nilai siswa/i IPS



Mata Pelajaran	Kelas X		Kelas XI	
	Semester 1	Semester 2	Semester 1	Semester 2
Matematika	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Indonesia	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Inggris	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
PKN	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Gambar 5.3: Halaman Index Pengisian Nilai IPS

## 4. Halaman hasil rekomendasi



No	Fakultas	Program Studi	IPK
----	----------	---------------	-----

Gambar 5.4: Halaman Hasil Rekomendasi

## 5.5 Pengujian Fungsional

Pengujian ini dilakukan untuk menguji fitur-fitur yang ada pada sistem rekomendasi program studi Universitas Katolik Parahyangan agar dapat berjalan dengan baik.

### 5.5.1 Pengujian Fungsional Pemilihan Jurusan SMA

Pengujian ini dilakukan pada fitur pemilihan jurusan saat SMA oleh siswa/i yang menjadi target sistem.

No	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian	Status
1	Memilih jurusan saat SMA	Sistem mengarahkan kepada form sesuai jurusan SMA	Sistem mengarahkan kepada form sesuai jurusan SMA	Sesuai

Tabel 5.1: Tabel Pengujian Fungsional Pemilihan SMA

### 5.5.2 Pengujian Fungsional Pengisian Nilai

Pengujian ini dilakukan pada fitur pengisian nilai mata pelajaran sesuai dengan jurusan saat SMA pada kelas X dan XI.

No	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian	Status
1	Mengisi nilai sesuai nilai rapor	Memeriksa valid tidaknya data yang dimasukkan dan memeriksa <i>range</i> nilai	Memeriksa valid tidaknya data yang dimasukkan dan memeriksa <i>range</i> nilai	Sesuai
2	Klik tombol <i>submit</i>	Mengarahkan kepada halaman hasil rekomendasi	Mengarahkan kepada halaman hasil rekomendasi	Sesuai
3	Mengisi nilai yang tidak valid	Memberikan pesan data tidak valid	Memberikan pesan data tidak valid	Sesuai

Tabel 5.2: Tabel Pengujian Fungsional Pengisian Nilai

## 5.6 Pengujian Eksperimental

Pada subbab ini, akan dilakukan pengujian sistem rekomendasi program studi Universitas Katolik Parahyangan. Pengujian dilakukan menggunakan *Mean Absolute Error* (MAE), *Root Mean Square Error* (RMSE), dan eksekusi waktu program. Data yang digunakan pada pengujian adalah seluruh data mahasiswa yang dibagi menjadi dua yaitu *train set* sebesar 70% dan *test set* sebesar 30%. Pengujian dilakukan dengan dua metode, yaitu :

### 5.6.1 Metode Dasar

Metode dasar ini dilakukan dengan cara menguji secara langsung *test set* kedalam sistem dengan menggunakan *train set*, gambar 5.5 dan gambar 5.6 merupakan hasil perhitungan untuk *Mean Absolute Error* (MAE), *Root Mean Square Error* (RMSE), dan eksekusi waktu program.

#### 1. IPA

No	MAE	RMSE	Time (Mic Sec)
1	0.27545454545455	0.34951109947755	0.0025620460510254

Gambar 5.5: Hasil Pengujian Jurusan IPA dengan Metode Dasar

#### 2. IPS

No	MAE	RMSE	Time (Mic Sec)
1	0.2944	0.36557571035286	0.0030970573425293

Gambar 5.6: Hasil Pengujian Jurusan IPS dengan Metode Dasar

### 5.6.2 Metode KMeans

Metode KMeans dilakukan dengan membuat kelompok dari *train set* dan mengelompokkan *test set* masuk kedalam kelompok mana, untuk mengurangi perhitungan kemiripan atau similaritas dengan pengguna yang memiliki kemiripan kecil. Pada pengujian ini dilakukan dengan nilai k 2 sampai 10 dan dilakukan pengulangan dalam membentuk kelompok sebanyak 30 kali. Gambar 5.7 dan

gambar 5.8 merupakan hasil perhitungan untuk *Mean Absolute Error* (MAE), *Root Mean Square Error* (RMSE), dan eksekusi waktu program.

#### 1. IPA

K	MAE	RMSE	Time (Mic Sec)
2	0.26676046176046	0.34804989151157	100.413022995
3	0.26316406462552	0.35059754079784	124.37367892265
4	0.26202499975993	0.35287057706885	135.68786978722
5	0.25704977141468	0.3445728235525	122.34353613853
6	0.25571329058145	0.3450314717282	135.42677283287
7	0.24646572081595	0.33287068283497	96.311838150024
8	0.24141842111954	0.32382977337753	100.59392499924
9	0.24610569388862	0.33354556355214	92.318086147308
10	0.24276286139087	0.3280378768029	92.794052839279

Gambar 5.7: Hasil Pengujian Jurusan IPA dengan Metode KMeans

#### 2. IPS

K	MAE	RMSE	Time (Mic Sec)
2	0.30913885943775	0.37518995373096	206.63060998917
3	0.30286638256251	0.36952562008495	174.55692005157
4	0.29623629882973	0.36337981974997	176.9764289856
5	0.29501845923321	0.36219610210439	208.81446313858
6	0.28981979928315	0.35760140231183	178.8926858902
7	0.28707882623017	0.3558129888324	176.53319191933
8	0.28336300178783	0.35293578942195	169.50131511688
9	0.27898667029878	0.34849401658213	147.50484895706
10	0.27735784990023	0.34686343917974	173.98230099678

Gambar 5.8: Hasil Pengujian Jurusan IPS dengan Metode KMeans

Dari hasil pengujian baik menggunakan metode 5.6.1 dan metode 5.6.2 dengan nilai MAE berada di 0.2 dan RMSE berada di 0.3, yang membedakan hanya pada waktu eksekusi program, dimana dengan menggunakan kmeans membutuhkan waktu lebih lama.



## BAB 6

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Berikut ini merupakan kesimpulan yang didapatkan berdasarkan penelitian yang sudah dilakukan :

1. Sistem rekomendasi program studi Universitas Katolik Parahyangan sudah dapat dikembangkan dan memberikan rekomendasi kepada pengguna khususnya siswa/i SMA pada kelas 11 yang ingin melanjutkan pendidikan di Universitas Katolik Parahyangan.
2. Sistem rekomendasi program studi Universitas Katolik Parahyangan yang dikembangkan dengan teknik *user-based collaborative filtering* memberikan rekomendasi berdasarkan *rating* yang diberikan pengguna lain yang telah lulus dari Universitas Katolik Parahyangan.
3. Hasil pengujian menggunakan metode *Mean Absolute Error* (MAE), *Root Mean Square Error* (RMSE), dan eksekusi waktu program pada metode dasar dan metode Kmeans dengan nilai MAE berada di 0.2, nilai RMSE berada di 0.3, dan waktu eksekusi program lebih lama pada metode KMeans, dikarenakan diperlukan membuat kelompok pada data *train set*.

#### 6.2 Saran

Berdasarkan hasil penelitian yang dilakukan, penulis dapat memberikan beberapa saran sebagai berikut :

1. Sistem rekomendasi program studi Universitas Katolik Parahyangan menggunakan data mahasiswa lulus pada jalur PMDK, untuk itu perlu ditambahkan data mahasiswa lulus pada jalur USM, dengan harapan semakin banyak jumlah pengguna yang memiliki kesanamaan dengan siswa/i SMA yang menggunakan sistem.
2. Sistem rekomendasi program studi Universitas Katolik Parahyangan menggunakan data mahasiswa yang berasal dari Biro Teknologi Informasi (BTI) Universitas Katolik Parahyangan. Penulis berharap agar format penyimpanan nilai baik untuk jalur penerimaan PMDK dan UMS sama, dengan harapan mudah untuk digunakan pada sistem yang sudah dibangun.
3. Teknik pengelompokkan yang digunakan adalah KMeans dimana saat pengujian membutuhkan waktu yang lebih lama, metode ini memberikan masalah pada waktu eksekusi program. Berdasarkan masalah ini, penulis berharap sistem dapat menggunakan metode lain untuk mengoptimalkan waktu eksekusi program.



## DAFTAR REFERENSI



# LAMPIRAN A

## KODE PROGRAM

Listing A.1: SiswaController.php

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 use App\Http\Controllers\UserBasedModelController;
8 use App\Http\Controllers\MahasiswaController;
9 use App\Http\Controllers\KMeansController;
10
11 class SiswaController extends Controller
12 {
13     private $mata_pelajaran = array(
14         "mtk" => 1,
15         "ind" => 2,
16         "ing" => 3,
17         "fsk" => 4,
18         "gbr" => 5,
19         "pkn" => 6,
20         "kma" => 7,
21     );
22
23     function index(Request $request)
24     {
25         // untuk penampung input dari form
26         $data = $request->input();
27         // untuk menampung input yang sudah diolah, agar mudah digunakan
28         $siswa = $this->dataSiswa($data);
29
30         // inisialisasi controller mahasiswa
31         $mahasiswa = new MahasiswaController();
32         // data mahasiswa
33         $mhs = $mahasiswa->index($siswa["btn"])->toArray();
34         // inisialisasi kmeans
35         $kmeans = new KMeansController(2, $mhs);
36
37         // hitung jarak siswa dengan centroid
38         // mengembalikan siswa masuk dalam cluster mana
39         $cluster = $kmeans->hitungJarakSiswa($siswa);
40
41         // mengubah data mhs dari seluruh mhs
42         // menjadi anggota satu cluster dengan siswa
43         $mhs = $kmeans->getCluster($cluster);
44
45         // inisialisasi userBasedModel
46         $userBasedModel = new UserBasedModelController($mhs, $siswa);
47
48         $result = $userBasedModel->getResult();
49
50         return view('/result', ['result' => $result]);
51     }
52
53     // mengubah data siswa dari form
54     // menjadi array dengan format mengikuti array mahasiswa
55     private function dataSiswa($data)
56     {
57         $i = 1;
58         $result = array();
59         $result["nilai"] = array();
60         foreach ($data as $key => $value) {
61             if ($key == "_token") {
62                 $result[$key] = $value;
63             } else {
64                 if ($i == 1) {
65                     // key untuk mata pelajaran
66                     $k = substr($key, 0, 3);
67                     // temporary array
68                     $temp = [];
69                     // masukan data (nilai) ke temp
70                     array_push($temp, ((int) $value / 20) - 1);
71                     $i++;
72                 } else {
73                     // masukan data nilai ke temp
74                     array_push($temp, ((int) $value / 20) - 1);
75                     $i++;
76                 }
77             }
78         }
79     }
80 }
```

```

76         if ($i == 5) {
77             // avg nilai
78
79             $avg = array_sum($temp) / count($temp);
80             array_push($temp, $avg);
81
82             // replace index ke-4 dengan AVG
83             $temp = $this->replaceKey($temp, 4, "AVG");
84
85             array_push($temp, $this->mata_pelajaran[$k]);
86
87             $temp = $this->replaceKey($temp, 5, "id_mata_pelajaran");
88
89             // masukan data ke result
90             // $result[$this->mata_pelajaran[$k]] = $temp;
91             array_push($result["nilai"], $temp);
92             // print($k." ");
93             $i = 1;
94         }
95     }
96 }
97 }
98 }
99
100 if (!empty($data["btnIPA"])) {
101     $result["btn"] = "IPA";
102 } else if (!empty($data["btnIPS"])) {
103     $result["btn"] = "IPS";
104 }
105 return $result;
106 }
107
108 private function replaceKey($temp, $oldKey, $newKey)
109 {
110     $temp[$newKey] = $temp[$oldKey];
111     unset($temp[$oldKey]);
112
113     return $temp;
114 }
115 }

```

Listing A.2: KMeansController.php

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class KMeansController extends Controller
8 {
9     public $currCentroid;
10    private $prevCentroid;
11    private $k;
12    public $cluster;
13    private $mahasiswa;
14    private $J0, $J1;
15
16    function __construct($k, $dataMahasiswa)
17    {
18        $this->k = $k;
19        $this->mahasiswa = $dataMahasiswa;
20        $this->inisialisasiCluster();
21        $this->currCentroid = array();
22
23        $this->J0 = 100;
24
25        $this->pilihCentroid();
26
27        $this->hitungJarakMhs();
28        // $this->printHitungJarak($jarak);
29        // $this->printCluster();
30
31        $status = true;
32        $idx = 0;
33        while ($status) {
34            $this->hitungCentroidBaru();
35            $idx++;
36            // print($idx);
37            // echo "<br>";
38            $status = $this->cekBatas();
39            $this->hitungJarakMhs();
40        }
41    }
42
43    private function inisialisasiCluster()
44    {
45        $this->cluster = array();
46        // inisialisasi untuk res anggota cluster
47        for ($i = 0; $i < $this->k; $i++) {
48            $this->cluster[$i] = array();
49        }
50    }
51
52    // fungsi untuk memilih secara acak mahasiswa mana
53    // yang akan dijadikan centroid
54    private function pilihCentroid()
55    {

```

```

56 |     for ($i = 0; $i < $this->k; $i++) {
57 |         // random secara acak key(idx) mahasiswa
58 |         $key = rand(0, 1739);
59 |         // cek apakah ada key(idx) pada mahasiswa
60 |         if (array_key_exists($key, $this->mahasiswa)) {
61 |             // cek apakah ada key(idx) pada currCentroid
62 |             if (!array_key_exists($key, $this->currCentroid)) {
63 |                 array_push($this->currCentroid, $this->mahasiswa[$key]);
64 |             } else {
65 |                 $i--;
66 |             }
67 |         } else {
68 |             $i--;
69 |         }
70 |     }
71 |
72 |     // untuk testing
73 |     // array_push($this->currCentroid, $this->mahasiswa[1]);
74 |     // // print_r($this->dataMahasiswa[1]);
75 |     // // echo "<br>";
76 |     // // echo "<br>";
77 |
78 |     // array_push($this->currCentroid, $this->mahasiswa[5]);
79 |     // // print_r($this->dataMahasiswa[5]);
80 |     // // echo "<br>";
81 |     // // echo "<br>";
82 | }
83 |
84 | // fungsi untuk menghitung jarak untuk untuk
85 | // mata pelajaran mtk (1) dan ing (3)
86 | private function hitungJarakMhs()
87 | {
88 |     $this->J1 = 0;
89 |     // looping sebanyak mahasiswa
90 |     foreach ($this->mahasiswa as $keyMhs => $valueMhs) {
91 |         // array sementara untuk menentukan masuk cluster mana
92 |         $tempCluster = array();
93 |         // array yang berisikan nilai satu mahasiswa
94 |         $nilaiMhs = $valueMhs['nilai'];
95 |         // looping sebanyak nilai mahasiswa
96 |         foreach ($nilaiMhs as $keyNilaiMhs => $valueNilaiMhs) {
97 |             // array untuk menampung jarak
98 |             $arrJarak = array();
99 |             // looping sebanyak centroid
100 |            foreach ($this->currCentroid as $keyCen => $valuCen) {
101 |                // penampung jarak
102 |                $jarak = 0;
103 |                // array yang berisikan nilai pada centroid
104 |                $nilaiCen = $valuCen['nilai'];
105 |                // looping sebanyak nilai centroid
106 |                foreach ($nilaiCen as $keyNilaiCen => $valueNilaiCen) {
107 |                    // cek apakah pada mata pelajaran yang sama atau tidak
108 |                    if (
109 |                        $valueNilaiMhs['id_mata_pelajaran'] == 1 && $valueNilaiCen['id_mata_pelajaran'] == 1
110 |                        ||
111 |                        $valueNilaiMhs['id_mata_pelajaran'] == 3 && $valueNilaiCen['id_mata_pelajaran'] == 3
112 |                    ) {
113 |                        // hitung jarak dengan euclidian distance
114 |                        $jarak = $this->euclidianDistance($valueNilaiMhs, $valueNilaiCen);
115 |                    } else if ($valueNilaiMhs['id_mata_pelajaran'] < $valueNilaiCen['id_mata_pelajaran']) {
116 |                        break;
117 |                    }
118 |                }
119 |                // memasukkan jarak antara mhs(nilai) dengan centroid(nilai)
120 |                // index 0 nilai dengan mata pelajaran mtk (1)
121 |                // index 1 nilai dengan mata pelajaran ing (3)
122 |                array_push($arrJarak, $jarak);
123 |            }
124 |            // cek apakah tempCluster kosong
125 |            if (empty($tempCluster)) {
126 |                array_push($tempCluster, $arrJarak);
127 |            } else {
128 |                // menghitung jarak sebenarnya
129 |                // dari dua nilai
130 |                for ($i = 0; $i < $this->k; $i++) {
131 |                    $tempCluster[0][$i] += $arrJarak[$i];
132 |                    $tempCluster[0][$i] = sqrt($tempCluster[0][$i]);
133 |                }
134 |            }
135 |        }
136 |        // menentukan mhs masuk pada cluster mana
137 |        $c = array_keys($tempCluster[0], min($tempCluster[0]))[0];
138 |
139 |        $this->J1 += $tempCluster[0][$c];
140 |
141 |        array_push($tempCluster[0], $c, $valueMhs['id_mahasiswa']);
142 |
143 |        // mengubah key index array
144 |        $tempCluster[0]['id_mahasiswa'] = $tempCluster[0][$this->k + 1];
145 |        unset($tempCluster[0][$this->k + 1]);
146 |
147 |        // memasukkan mhs ke array hasil
148 |        array_push($this->cluster[$c], $valueMhs);
149 |    }
150 | }
151 |
152 | public function hitungJarakSiswa($siswa)
153 | {
154 |     $nilaiSiswa = $siswa['nilai'];

```

```

155 $tempCluster = array();
156 // looping untuk nilai siswa
157 foreach ($nilaiSiswa as $keyNilaiSiswa => $valueNilaiSiswa) {
158     $sarrJarak = array();
159     // looping sebanyak centroid
160     foreach ($this->currCentroid as $keyCen => $valuCen) {
161         // penampung jarak
162         $jarak = 0;
163         // array yang berisikan nilai pada centroid
164         $nilaiCen = $valuCen['nilai'];
165         // looping sebanyak nilai centroid
166         foreach ($nilaiCen as $keyNilaiCen => $valueNilaiCen) {
167             // cek apakah pada mata pelajaran yang sama atau tidak
168             if (
169                 $valueNilaiSiswa['id_mata_pelajaran'] == 1 && $valueNilaiCen['id_mata_pelajaran'] == 1
170                 ||
171                 $valueNilaiSiswa['id_mata_pelajaran'] == 3 && $valueNilaiCen['id_mata_pelajaran'] == 3
172             ) {
173                 // hitung jarak dengan euclidian distance
174                 $jarak = $this->euclidiandistance($valueNilaiSiswa, $valueNilaiCen);
175             } else if ($valueNilaiSiswa['id_mata_pelajaran'] < $valueNilaiCen['id_mata_pelajaran']) {
176                 break;
177             }
178         }
179         // memasukkan jarak antara mhs(nilai) dengan centroid(nilai)
180         // index 0 nilai dengan mata pelajaran mtk (1)
181         // index 1 nilai dengan mata pelajaran ing (3)
182         array_push($sarrJarak, $jarak);
183     }
184     // cek apakah tempCluster kosong
185     if (empty($tempCluster)) {
186         array_push($tempCluster, $sarrJarak);
187     } else {
188         // menghitung jarak sebenarnya
189         // dari dua nilai
190         for ($i = 0; $i < $this->k; $i++) {
191             $tempCluster[0][$i] += $sarrJarak[$i];
192             $tempCluster[0][$i] = sqrt($tempCluster[0][$i]);
193         }
194     }
195 }
196 // menentukan mhs masuk pada cluster mana
197 $res = array_keys($tempCluster[0], min($tempCluster[0]))[0];
198
199 return $res;
200 }
201
202 // parameter berisikan array of nilai satu mata pelajaran
203 private function euclidiandistance($mhs, $centroid)
204 {
205     // asumsi itung yang beririsan aja
206     $result = 0;
207     for ($i = 0; $i < 4; $i++) {
208         $result += pow($mhs[$i] - $centroid[$i], 2);
209     }
210     $result += pow($mhs['AVG'] - $centroid['AVG'], 2);
211     return $result;
212 }
213
214 private function hitungCentroidBaru()
215 {
216     // mengisi centroid sebelumnya dengan centroid saat ini
217     $this->prevCentroid = $this->currCentroid;
218
219     // reset nilai curr centroid
220     $this->resetCentroid();
221     // print_r($this->currCentroid);
222
223     // looping sebanyak centroid
224     foreach ($this->currCentroid as $keyCen => $valuCen) {
225         // array yang berisikan nilai pada centroid
226         $nilaiCen = $valuCen['nilai'];
227         // looping sebanyak nilai centroid
228         foreach ($nilaiCen as $keyNilaiCen => $valueNilaiCen) {
229             // penampung untuk anggota pada cluster tertentu (sesuai index/key)
230             $anggota = $this->cluster[$keyCen];
231             if (count($anggota) != 0) {
232                 foreach ($anggota as $keyAnggota => $valueAnggota) {
233                     // penampung untuk nilai anggota
234                     $nilaiAnggota = $valueAnggota['nilai'];
235                     foreach ($nilaiAnggota as $keyNilaiAnggota => $valueNilaiAnggota) {
236                         if (
237                             $valueNilaiCen['id_mata_pelajaran'] == 1 && $valueNilaiAnggota['id_mata_pelajaran'] == 1
238                             ||
239                             $valueNilaiCen['id_mata_pelajaran'] == 3 && $valueNilaiAnggota['id_mata_pelajaran'] == 3
240                         ) {
241                             // update nilai 101, 102, 111, 112
242                             for ($i = 0; $i < 4; $i++) {
243                                 $nilaiLama = $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen][$i];
244                                 $nilaiBaru = $anggota[$keyAnggota]['nilai'][$keyNilaiAnggota][$i];
245
246                                 $this->updateNilai($keyCen, $keyNilaiCen, $nilaiLama, $nilaiBaru, $i);
247                             }
248
249                             // update nilai avg
250                             $nilaiLama = $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen]['AVG'];
251                             $nilaiBaru = $anggota[$keyAnggota]['nilai'][$keyNilaiAnggota]['AVG'];
252                             $this->updateNilai($keyCen, $keyNilaiCen, $nilaiLama, $nilaiBaru, 'AVG');
253                         }

```



```

254         }
255     }
256     } else {
257         // random nilai baru
258         $this->randomNilaiBaru($keyCen, $keyNilaiCen);
259         // print_r($this->currCentroid[$keyCen]);
260         // echo "<br>";
261         // echo "<br>";
262     }
263 }
264 }
265
266 $this->hitungRata2();
267 // print_r($this->currCentroid);
268 }
269
270 private function resetCentroid()
271 {
272     foreach ($this->currCentroid as $keyCen => $valuCen) {
273         // array yang berisikan nilai pada centroid
274         $nilaiCen = $valuCen['nilai'];
275         // looping sebanyak nilai centroid
276         foreach ($nilaiCen as $keyNilaiCen => $valueNilaiCen) {
277             for ($i = 0; $i < 4; $i++) {
278                 $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen][$i] = 0;
279             }
280             // update nilai avg
281             $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen]['AVG'] = 0;
282         }
283     }
284 }
285
286 private function updateNilai($keyCen, $keyNilaiCen, $nilaiLama, $nilaiBaru, $i)
287 {
288     $nilai = $nilaiLama + $nilaiBaru;
289     $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen][$i] = $nilai;
290 }
291
292 private function hitungRata2()
293 {
294     foreach ($this->currCentroid as $keyCen => $valuCen) {
295         // array yang berisikan nilai pada centroid
296         $nilaiCen = $valuCen['nilai'];
297         // looping sebanyak nilai centroid
298         $anggota = $this->cluster[$keyCen];
299         $count = count($anggota);
300         if ($count != 0) {
301             foreach ($nilaiCen as $keyNilaiCen => $valueNilaiCen) {
302                 for ($i = 0; $i < 4; $i++) {
303                     $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen][$i] = $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen][$i] / $count;
304                 }
305                 // update nilai avg
306                 $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen]['AVG'] = $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen]['AVG'] / $count;
307             }
308         }
309     }
310 }
311
312 private function randomNilaiBaru($keyCen, $keyNilaiCen)
313 {
314     for ($i = 0; $i < 4; $i++) {
315         $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen][$i] = rand(1, 3) + rand(1, 10) / 10;
316     }
317     // update nilai avg
318     $this->currCentroid[$keyCen]['nilai'][$keyNilaiCen]['AVG'] = rand(1, 3) + rand(1, 10) / 10;
319 }
320
321 public function cekBatas()
322 {
323     $batas = abs($this->J0 - $this->J1);
324
325     if ($batas < 0.1) {
326         return false;
327     }
328
329     $this->J0 = $this->J1;
330
331     return true;
332 }
333
334 public function getCluster($idx)
335 {
336     return $this->cluster[$idx];
337 }
338
339 private function printHitungJarak($jarak)
340 {
341     foreach ($jarak as $keyJarak => $valueJarak) {
342         print_r($valueJarak);
343         echo "<br>";
344         echo "<br>";
345     }
346 }
347
348 private function printCluster()
349 {
350

```

```

351     foreach ($this->cluster as $keyCluster => $valueCluster) {
352         print("CLUSTER_" . $keyCluster);
353         echo "<br>";
354         echo "<br>";
355         foreach ($valueCluster as $v) {
356             print_r($v);
357             echo "<br>";
358             echo "<br>";
359         }
360     }
361 }
362 }

```

Listing A.3: UserBasedModelController.php

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class UserBasedModelController extends Controller
8 {
9     private $pearsonCorrelation;
10    private $prediction;
11    private $result;
12
13    // mode 0 untuk pengguna
14    // mode 1 untuk pengujian
15    public function __construct($mahasiswa, $siswa, $mode = 0)
16    {
17        // $this->mahasiswa = $mhs;
18        // $this->siswa = $siswa;
19
20        // inisialisasi prediction controller
21        $this->prediction = new PredictionController();
22
23        if ($mode == 0) {
24            // inisialisasi pearson correlation controller
25            $this->pearsonCorrelation = new PearsonCorrelationController();
26            // menghitung kesamaan atau similaritas
27            $pearson = $this->calculateSimilarity($mahasiswa, $siswa);
28
29            // menghitung prediksi IPK untuk siswa
30            $this->result = $this->calculatePredict($pearson);
31        } else if ($mode == 1) {
32            $this->pearsonCorrelation = new PearsonCorrelationPengujianController();
33        }
34    }
35
36    public function calculateSimilarity($mahasiswa, $siswa)
37    {
38        return $this->pearsonCorrelation->calculatePearson($mahasiswa, $siswa);
39    }
40
41    public function calculatePredict($pearson)
42    {
43        return $this->prediction->calculatePredict($pearson);
44    }
45
46    public function getResult()
47    {
48        return $this->result;
49    }
50 }

```

Listing A.4: PearsonCorrelationController.php

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class PearsonCorrelationController extends Controller
8 {
9
10    private $sdSiswa;
11
12    function __construct()
13    {
14        $this->sdSiswa = array();
15    }
16
17    // untuk menghitung kovariansi satu mahasiswa
18    // dengan satu siswa
19    private function calculateCovariance($mhs, $siswa)
20    {
21        $res = 0;
22        // nilai 1 mhs
23        $nilaiMhs = $mhs['nilai'];
24        $nilaiSiswa = $siswa['nilai'];
25        // looping sebanyak nilai
26        foreach ($nilaiSiswa as $nSiswa) {
27            $idMP = $nSiswa['id_mata_pelajaran'];
28            // looping sebanyak nilai mahasiswa pada index
29            foreach ($nilaiMhs as $nMhs) {

```

```

30 |         // hanya menghitung mata pelajaran yang beririsan
31 |         if ($idMP == $mhs['id_mata_pelajaran']) {
32 |             for ($i = 0; $i < 4; $i++) {
33 |                 // mahasiswa * siswa
34 |                 $res += ($mhs[$i] - $mhs["AVG"]) * ($siswa[$i] - $siswa["AVG"]);
35 |             }
36 |         } else if ($idMP < $mhs['id_mata_pelajaran']) {
37 |             break;
38 |         }
39 |     }
40 | }
41 | return $res;
42 | }
43 |
44 | // mengitung standar deviasi untuk satu mahasiswa
45 | // dan satu siswa
46 | private function calculateStandarDeviation($mhs, $siswa)
47 | {
48 |     $res = array();
49 |
50 |     $sdMhs = 0;
51 |     $sdSiswa = 0;
52 |     // nilai 1 mhs
53 |     $nilaiMhs = $mhs['nilai'];
54 |     $nilaiSiswa = $siswa['nilai'];
55 |     // looping sebanyak nilai
56 |     foreach ($nilaiSiswa as $nSiswa) {
57 |         $idMP = $nSiswa['id_mata_pelajaran'];
58 |         // looping sebanyak nilai mahasiswa pada index
59 |         foreach ($nilaiMhs as $nMhs) {
60 |             // hanya menghitung mata pelajaran yang beririsan
61 |             if ($idMP == $nMhs['id_mata_pelajaran']) {
62 |                 for ($i = 0; $i < 4; $i++) {
63 |                     $sdMhs += pow($nMhs[$i] - $nMhs["AVG"], 2);
64 |                     $sdSiswa += pow($nSiswa[$i] - $nSiswa["AVG"], 2);
65 |                 }
66 |             } else if ($idMP < $nMhs['id_mata_pelajaran']) {
67 |                 break;
68 |             }
69 |         }
70 |     }
71 |     array_push($res, sqrt($sdMhs), sqrt($sdSiswa));
72 |
73 |     return $res;
74 | }
75 |
76 | // menghitung kemiripan dengan perason
77 | // $mahasiswa -> seluruh mahasiswa sesuai dengan jurusan SMA
78 | // $siswa ->
79 | public function calculatePearson($mahasiswa, $siswa)
80 | {
81 |     $res = array();
82 |     foreach ($mahasiswa as $mhs) {
83 |         $covariance = $this->calculateCovariance($mhs, $siswa);
84 |         $sd = $this->calculateStandarDeviation($mhs, $siswa);
85 |         $sdMhs = $sd[0]; // standar deviasi untuk mahasiswa
86 |         $sdSiswa = $sd[1]; // standar deviasi untuk siswa
87 |
88 |         $id_prodi = $mhs['id_program_studi'];
89 |         $IPK = $mhs['IPK'];
90 |
91 |         $sim = $covariance / ($sdMhs * $sdSiswa);
92 |         // atur threshold
93 |         if ($sim > 0) {
94 |             // inisialisasi array agar tidak null
95 |             $res[$mhs['id_mahasiswa']] = array();
96 |             array_push($res[$mhs['id_mahasiswa']], $sim, $id_prodi, $IPK);
97 |         }
98 |     }
99 |     return $res;
100 | }
101 | }

```

Listing A.5: PredictionController.php

```

1 | <?php
2 |
3 | namespace App\Http\Controllers;
4 |
5 | use Illuminate\Http\Request;
6 |
7 | class PredictionController extends Controller
8 | {
9 |     private $programStudi;
10 |    private $fakultas;
11 |
12 |    public function __construct()
13 |    {
14 |        $this->programStudi = new ProgramStudiController();
15 |        $this->fakultas = new FakultasController();
16 |    }
17 |
18 |    public function calculatePredict($pearson)
19 |    {
20 |        $res = array();
21 |
22 |        // a = Sigma(sim * IPK)
23 |        $a = 0;

```

```

24 // b = Sigma(sim)
25 $b = 0;
26 // pred = a/b
27 // id_mhs -> sim, id_prodi, IPK, avgMhs, avgSiswa
28 foreach ($pearson as $id_mhs => $value) {
29     $a += $value[0] * $value[2];
30     $b += $value[0];
31
32     $next = next($pearson);
33
34     if ($next != null) {
35         // program studi mhs sekarang berbeda dengan mhs selanjutnya
36         if ($value[1] != $next[1]) {
37             $res = $this->insertData($res, $a, $b, $value[1]);
38
39             $a = 0;
40             $b = 0;
41         }
42     } else if ($next == null) {
43         $res = $this->insertData($res, $a, $b, $value[1]);
44     }
45 }
46
47 // // penampung untuk nilai prediksi IPK
48 $score = array_column($res, 0);
49 // sort berdasarkan nilai prediksi ipk terbesar
50 array_multisort($score, SORT_DESC, $res);
51
52 return $res;
53 }
54
55 private function insertData($res, $a, $b, $id_prodi)
56 {
57     $pred = $a / $b;
58     $namaFakultas = $this->fakultas->getNamaFakultas($id_prodi);
59     $namaProdi = $this->programStudi->getNamaProgramStudi($id_prodi);
60     $res[$id_prodi] = array();
61     // dibalik
62     array_push($res[$id_prodi], $pred, $namaFakultas, $namaProdi);
63
64     return $res;
65 }
66 }

```