



TUGAS AKHIR - IF184802

PEMODELAN DAN VISUALISASI REKOMENDASI KONTEN BELAJAR MENGGUNAKAN ALGORITMA CONTENT BASED FILTERING

FINA FITRI YUNITA
NRP 05111640000067

Dosen Pembimbing
Dr. Umi Laili Yuhana S.Kom., M.Sc.
Nurul Fajrin Ariyani, S.Kom., M.Sc.

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020



TUGAS AKHIR - IF184802

**PEMODELAN DAN VISUALISASI REKOMENDASI
KONTEN BELAJAR MENGGUNAKAN
ALGORITMA *CONTENT BASED FILTERING***

FINA FITRI YUNITA
NRP 05111640000067

Dosen Pembimbing
Dr. Umi Laili Yuhana S.Kom., M.Sc.
Nurul Fajrin Ariyani, S.Kom., M.Sc.

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

[Halaman ini sengaja dikosongkan]



UNDERGRADUATE THESIS - IF184802

MODELLING AND VISUALIZATION OF COURSE RECOMMENDATION USING CONTENT BASED FILTERING ALGORITHM

FINA FITRI YUNITA
NRP 0511164000067

Advisor
Dr. Umi Laili Yuhana S.Kom., M.Sc.
Nurul Fajrin Ariyani, S.Kom., M.Sc.

INFORMATICS ENGINEERING DEPARTMENT
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

PEMODELAN DAN VISUALISASI REKOMENDASI KONTEN BELAJAR MENGGUNAKAN ALGORITMA CONTENT BASED FILTERING

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer pada
Rumpun Mata Kuliah Rekayasa Perangkat Lunak
Program Studi S-1 Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember


Oleh :

FINA FITRI YUNITA
NRP : 05111640000067

Disetujui oleh Dosen Pembimbing Tugas Akhir :


Dr. Umi Laili Yuhana S.Kom., M.Sc.

NIP: 197906262005012002


.....
(pembimbing 1)

Nurul Fajrin Ariyani, S.Kom., M.Sc.

NIP: 198607222015042003


.....
(pembimbing 2)

SURABAYA
JUNI 2020

[Halaman ini sengaja dikosongkan]

**PEMODELAN DAN VISUALISASI REKOMENDASI KONTEN
BELAJAR MENGGUNAKAN ALGORITMA CONTENT BASED
FILTERING**

Nama Mahasiswa : FINA FITRI YUNITA
NRP : 0511154000067
Jurusan : Teknik Informatika ITS
Dosen Pembimbing I : Dr. Umi Laili Yuhana S.Kom., M.Sc.
Dosen Pembimbing II : Nurul Fajrin Ariyani, S.Kom., M.Sc.

Abstrak

Dalam belajar mandiri secara daring, tentu tidak semua siswa mengetahui pelajaran apa yang relevan dengan pelajaran yang telah diselesaikan. Dengan tidak adanya kehadiran guru dalam menemani siswa belajar secara mandiri, maka dibutuhkan suatu tools untuk memberikan siswa bantuan dalam memilih konten belajar.

Berdasarkan permasalahan tersebut, pembuatan Plugin Rekomendasi Course diusulkan sebagai solusi untuk membantu memberikan rekomendasi konten belajar. Dengan memanfaatkan Moodle, yaitu salah satu platform e-learning, plugin yang dibangun dapat dimanfaatkan oleh pengajar untuk mempermudah siswa, dalam melakukan belajar mandiri secara daring di rumah.

Tugas Akhir ini berhasil membangun dan mengimplementasikan sebuah plugin yang dapat digunakan sebagai pemberi rekomendasi course sebagai konten belajar dalam platform e-learning.

Kata kunci: E-learning, Moodle, Rekomendasi, Course, Plugin

[Halaman ini sengaja dikosongkan]

MODELING AND VISUALIZATION OF LEARNING CONTENT RECOMMENDATIONS USING CONTENT BASED FILTERING ALGORITHM

Nama Mahasiswa : FINA FITRI YUNITA
NRP : 0511154000067
Jurusan : Teknik Informatika ITS
Dosen Pembimbing I : Dr. Umi Laili Yuhana S.Kom., M.Sc.
Dosen Pembimbing II : Nurul Fajrin Ariyani, S.Kom., M.Sc.

Abstract

In online self-study, certainly not all students know what lessons are relevant to the completed lessons. With the absent of teacher in the learning process, a tool is needed to give students assistance in choosing learning content.

Therefore, the creation of the Plugin Recommendation Course is proposed as a solution to help recommend learning content. By utilizing Moodle, which is one of the e-learning platforms, plugins that are built can be utilized by teachers to facilitate students, in conducting independent learning online at home. Not only at home, these tools can also be used in learning other things.

This Final Project successfully builds and implements a plugin that can be used as a recommendation course provider as learning content in the e-learning platform.

Keywords: : *E-learning, Moodle, Recommendation, Course, Plugin*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Alhamdulillahirobbil ‘alamiin, puji syukur kepada Tuhan Yang Maha Esa atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul:

PEMODELAN DAN VISUALISASI REKOMENDASI KONTEN BELAJAR MENGGUNAKAN ALGORITMA CONTENT BASED FILTERING

Melalui lembar ini, penulis ingin menyampaikan ucapan terimakasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah SWT atas segala nikmat dan rahmat yang telah diberikan selama ini serta senantiasa menemani dan memberi kemudahan kepada penulis dalam menyelesaikan Tugas Akhir.
2. Kedua Orang Tua Bapak Farid Fanani dan Ibu Yayuk Purwaningsih, adik Diana Evita Dewi yang telah memberikan doa dan dukungan selama ini.
3. Ibu Dr. Umi Laili Yuhana S.Kom., M.Sc selaku dosen pembimbing I yang selalu memberikan motivasi dan membimbing penulis selama pengerjaan Tugas Akhir.
4. Ibu Nurul Fajrin Ariyani, S.Kom., M.Sc selaku dosen pembimbing II yang senantiasa memberikan masukan, arahan, dan bantuan sehingga penulis dapat menyelesaikan Tugas Akhir ini.
5. Bapak dan Ibu dosen Teknik Informatika ITS yang telah membina dan memberikan ilmu kepada penulis selama menempuh studi di Teknik Informatika ITS.
6. Teman-teman seperjuangan Tugas Akhir; April, Mutia, Fira, Rifka, Yasinta, Opi, Sari, dan Farida yang saling menyemangati selama mengerjakan Tugas Akhir
7. Teman-teman BPH Schematics 2017/2018 yang telah

memberikan kesempatan penulis untuk mendapatkan pengalaman lebih selama 1 tahun kepengurusan.

8. Teman-teman angkatan 2016 yang telah memberikan motivasi selama penulis berkuliah di Informatika ITS.
9. Serta pihak lain yang namanya tidak dapat penulis sebutkan satu-persatu.

Penulis menyadari sepenuhnya bahwa Tugas Akhir ini masih memiliki kekurangan. Oleh karena itu, dengan tangan terbuka, penulis menerima segala saran dan kritik dari pembaca untuk perbaikan ke depannya.

Surabaya, 2020

Fina Fitri Yunita

DAFTAR ISI

Abstrak	vii
Abstract.....	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xvi
DAFTAR TABEL	xix
DAFTAR KODE SUMBER	xx
1 BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah	2
1.4. Tujuan	3
1.5. Manfaat	3
1.6. Metodologi.....	3
1.7. Sistematika Penulisan	4
2 BAB II TINJAUAN PUSTAKA.....	7
2.1. Sistem Rekomendasi.....	7
2.1.1. Collaborative Filtering	7
2.1.2. Content Based Recommendation	7
2.2. E-Learning	8
2.3. Moodle	8
2.4. TF-IDF	9
2.5. Cosine Similarity	10

2.6.	PHP.....	10
3	BAB III ANALISIS DAN PERANCANGAN.....	13
3.1.	Analisis.....	13
3.1.1.	Analisis Permasalahan.....	13
3.1.2.	Deskripsi Umum Perangkat Lunak.....	14
3.1.3.	Spesifikasi Kebutuhan Perangkat Lunak.....	19
3.2.	Perancangan.....	25
3.2.1.	Perancangan Arsitektur Sistem.....	25
3.2.2.	Perancangan Antarmuka Pengguna.....	27
3.2.3.	Perancangan Data.....	30
4	BAB IV IMPLEMENTASI.....	35
4.1.	Lingkungan Implementasi.....	35
4.2.	Implementasi Antarmuka Sistem.....	35
4.2.1.	Halaman Menambahkan Fitur Rekomendasi Course	36
4.2.2.	Halaman Melihat Hasil Rekomendasi Course	37
4.3.	Implementasi Kasus Penggunaan.....	37
4.3.1.	Implementasi Menambahkan Fitur Rekomendasi Course	37
4.3.2.	Implementasi Melihat Hasil Rekomendasi Course	39
5	BAB V PENGUJIAN DAN EVALUASI	47
5.1	Lingkungan Pengujian.....	47
5.2	Penambahan Plugin Pemberi Rekomendasi Course Pada Moodle.....	48
5.3	Pengujian Moodle Tanpa Plugin dan Moodle dengan Plugin	51

5.4	Pengujian Fungsionalitas	54
5.4.1	Pengujian Menambahkan Fitur Rekomendasi Course	54
5.4.2	Pengujian Melihat Hasil Rekomendasi Course...	58
5.5	Evaluasi Pengujian Fungsionalitas.....	61
6	BAB VI KESIMPULAN DAN SARAN	63
4	65
6.1	Kesimpulan	65
6.2	Saran	66
7	DAFTAR PUSTAKA.....	67
8	LAMPIRAN.....	69
9	BIODATA PENULIS.....	71

DAFTAR GAMBAR

Gambar 3.1 Deskripsi Umum Sistem	15
Gambar 3.2 Rancangan Model Perhitungan Rekomendasi <i>Course</i> Dalam Moodle	16
Gambar 3.3 Diagram Aktivitas UC001	21
Gambar 3.4 Sequence Diagram Aktivitas UC001	22
Gambar 3.5 Diagram Aktivitas UC001	24
Gambar 3.6 Sequence Diagram Aktivitas UC002.....	25
Gambar 3.7 Ilustrasi Arsitektur Sistem Web.....	26
Gambar 3.8 Rancangan Antarmuka Menambahkan Fitur Rekomendasi <i>Course</i>	28
Gambar 3.9 Rancangan Halaman Antarmuka Melihat Rekomendasi <i>Course</i>	29
Gambar 3.10 CDM Sistem	30
Gambar 3.11 PDM Sistem.....	31
Gambar 4.1 Implementasi Antarmuka Menambahkan Fitur Rekomendasi <i>Course</i>	36
Gambar 4.2 Implementasi Melihat Hasil Rekomendasi Course. 37	37
Gambar 5.1 Halaman <i>Dashboard</i> Moodle	48
Gambar 5.2 Halaman <i>Site Administration</i>	49
Gambar 5.3 Halaman <i>Plugin Installer</i>	49
Gambar 5.4 Halaman <i>Upload File Plugin</i>	50
Gambar 5.5 Halaman <i>Plugin Installer</i>	50
Gambar 5.6 Moodle Tanpa <i>Plugin Course Recommender</i>	51
Gambar 5.7 Moodle dengan <i>Plugin Course Recommender</i>	52
Gambar 5.8 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran KPK dan FPB	56
Gambar 5.9 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran Pecahan.....	56
Gambar 5.10 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran Pengumpulan dan Penyajian Data..	57
Gambar 5.11 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran Bilangan Bulat.....	57

Gambar 5.12 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran KPK dan FPB.....	59
Gambar 5.13 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran Pecahan	59
Gambar 5.14 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran Pengumpulan dan Penyajian Data.....	60
Gambar 5.15 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran Bilangan Bulat.....	60

DAFTAR TABEL

Tabel 3.1 Contoh data dari mdl_index dengan course Statistika	17
Tabel 3.2 Spesifikasi Karakteristik Aktor pada Sistem	19
Tabel 3.3 Kebutuhan Fungsional Sistem	20
Tabel 3.4 Spesifikasi Kasus Menambahkan Fitur Rekomendasi Course	20
Tabel 3.5 Spesifikasi Kasus Penggunaan Melihat Rekomendasi Course	22
Tabel 3.6 Penjelasan Antarmuka Menambahkan Fitur Rekomendasi Course	28
Tabel 3.7 Penjelasan Antarmuka Melihat Rekomendasi Course	29
Tabel 3.7 Deskripsi Rinci Tabel mdl_course	31
Tabel 3.9 Deskripsi Rinci Tabel Jenis mdl_index	32
Tabel 3.10 Deskripsi Rinci Tabel mdl_vektor	33
Tabel 3.11 Deskripsi Rinci Tabel mdl_similarity	33
Tabel 5.1 Lingkungan Pengujian Fungsionalitas Perangkat Lunak	47
Tabel 5.2 Hasil Pengujian kepada Pengajar	53
Tabel 5.3 Pengujian Menambahkan Fitur Rekomendasi Course	54
Tabel 5.4 Pengujian Melihat Hasil Rekomendasi Course	58
Tabel 5.5 Evaluasi Pengujian Fungsionalitas	63
Tabel 8.1 Kuesioner Pengujian Terhadap Pengajar	69

DAFTAR KODE SUMBER

Kode Sumber 4.1.Fungsi Menambahkan Fitur Rekomendasi <i>Course</i>	38
Kode Sumber 4.2. Fungsi Melihat Hasil Rekomendasi <i>Course</i> .	45

BAB I

PENDAHULUAN

Bab pendahuluan membahas garis besar penyusunan Tugas Akhir yang meliputi latar belakang, tujuan pembuatan, rumusan dan batasan permasalahan, metodologi penyusunan Tugas Akhir, dan sistematika penulisan.

1.1. Latar Belakang

Sistem e-learning merupakan penerapan konsep pendidikan dalam proses belajar mengajar yang memanfaatkan teknologi informasi. Salah satu framework e-learning adalah Moodle. Dalam Online Learning Management System ini, siswa dapat mengakses materi yang telah diupload oleh guru secara online, sehingga siswa dapat melakukan pembelajaran secara daring sebagai penunjang pembelajaran formal di sekolah.

Dengan wadah pembelajaran yang sudah tersedia, tentu akan memudahkan siswa untuk belajar secara mandiri. Namun, dengan kurangnya asistensi dari guru, tidak dipungkiri siswa akan kebingungan dengan materi apa yang akan dipelajari berikutnya. Hal ini dikarenakan, pada platform e-learning tersebut masih belum ada pemberian rekomendasi konten belajar, sehingga ketika siswa telah menyelesaikan satu bab pelajaran, tidak ada rekomendasi bab selanjutnya dan siswa dapat memilih pelajaran selanjutnya secara random apabila tanpa bimbingan dari guru. Oleh karena hal di atas, maka pada tugas akhir ini akan dibuat suatu plugin pada Moodle, yang dapat memberikan rekomendasi konten belajar yang berkaitan dengan materi yang baru saja diselesaikan.

Dalam algoritma sistem rekomendasi sendiri, ada beberapa macam. Yang cukup dikenal diantaranya adalah Collaborative Filtering, dan Content Based Filtering. Collaborative Filtering adalah pemberian rekomendasi didasarkan dengan *user behaviour*, dimana atribut yang digunakan bukanlah konten dari item itu sendiri, namun menggunakan riwayat rating dari user

tersebut maupun user lain. Sedangkan pengertian dari Content Based Filtering adalah pemberian rekomendasi berdasarkan data atau atribut dari sebuah konten, yang kemudian akan dihitung kemiripannya dengan konten yang sesuai. Perhitungan kemiripan tersebut dapat menggunakan euclidian distance maupun cosine similarity. Berdasarkan 2 tipe rekomendasi diatas, metode yang paling tepat untuk digunakan dalam pemberian rekomendasi materi/konten belajar adalah metode Content Based Filtering, dimana pemberian rekomendasi didasarkan dari atribut dan data dari course dan subjek pelajaran. Metode ini diambil dengan pertimbangan, sistem akan menampilkan rekomendasi materi yang memiliki kemiripan dengan materi yang sedang diambil. Dan tidak didasarkan dari perilaku dan aktivitas user.

1.2. Rumusan Masalah

Perumusan masalah yang terdapat pada Tugas Akhir ini, antara lain adalah:

1. Bagaimana cara mengekstraksi atribut deskripsi subjek pelajaran?
2. Bagaimana cara merancang dan membangun aplikasi e-learning yang memiliki sistem rekomendasi?
3. Bagaimana cara membuat plugin didalam Moodle?

1.3. Batasan Masalah

Batasan masalah yang terdapat pada Tugas Akhir ini, sebagai berikut:

1. Framework yang digunakan menggunakan Moodle 3.7
2. Konten belajar yang digunakan menggunakan mata pelajaran Matematika SD kelas 4, 5, dan 6
3. Rekomendasi berdasarkan kemiripan deskripsi topik, bukan learning path

1.4. Tujuan

Tujuan dari pembuatan tugas akhir ini adalah untuk membuat plugin pada Moodle yang dapat memberikan rekomendasi konten belajar.

1.5. Manfaat

Manfaat dari pembuatan tugas akhir ini adalah siswa terbantu untuk memilih konten belajar, sehingga proses pembelajaran menjadi lebih efektif dan efisien.

1.6. Metodologi

Tahap yang dilakukan untuk menyelesaikan Tugas Akhir ini adalah sebagai berikut:

1. Penyusunan Proposal Tugas Akhir

Proposal Tugas Akhir ini berisi tentang deskripsi pendahuluan dari Tugas Akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan Tugas Akhir, rumusan masalah yang diangkat, batasan masalah untuk Tugas Akhir, tujuan dari pembuatan Tugas Akhir, dan manfaat dari hasil pembuatan Tugas Akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan Tugas Akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan Tugas Akhir mulai dari penyusunan proposal hingga penyusunan buku Tugas Akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan Tugas Akhir.

2. Studi literatur

Pada studi literatur akan dipelajari beberapa referensi yang relevan terhadap tugas akhir yang dikerjakan. Literasi yang didapatkan berasal dari jurnal yang kredibel, dokumentasi website resmi, dan artikel yang dapat dipercaya. Hal-hal yang dipelajari yaitu tentang sistem

rekomendasi, Moodle, Content Based Filtering, TF-IDF, Cosine Similarity.

3. Analisis dan desain sistem

Tahap ini meliputi perumusan kebutuhan fungsional, kebutuhan non-fungsional, kasus penggunaan, diagram aktivitas, diagram kelas, rancangan antarmuka pengguna serta pembuatan rancangan basis data.

4. Implementasi

Pada tahap implementasi, perangkat lunak yang digunakan dalam pengerjaan tugas akhir ini adalah pemrograman PHP untuk pembuatan plugin bertipe blocks pada Moodle.

5. Uji Coba dan Evaluasi

Pada tahapan ini dilakukan uji coba pada perangkat lunak dengan pengujian berdasarkan *usability* dan *maintenance*.

6. Penyusunan Buku Tugas Akhir

Pada tahapan ini disusun buku yang membuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

1.7. Sistematika Penulisan

Buku Tugas Akhir ini terdiri atas beberapa bab yang tersusun secara sistematis, yaitu sebagai berikut.

1. Bab I. Pendahuluan

Bab pendahuluan berisi penjelasan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan Tugas Akhir.

2. Bab II. Tinjauan Pustaka

Bab tinjauan pustaka berisi penjelasan mengenai dasar teori yang mendukung pengerjaan Tugas Akhir.

3. Bab III. Analisis dan Perancangan

Bab ini berisi tentang desain sistem, rancangan basis data, diagram kasus penggunaan, diagram aktivitas dan rancangan antarmuka pengguna.

4. Bab IV. Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa tampilan antarmuka yang telah dibuat dan dapat berfungsi untuk mengakomodir kebutuhan fungsional yang ada.

5. Bab V. Uji Coba dan Evaluasi

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

6. Bab VI. Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

[Halaman ini sengaja dikosongkan]

BAB II

TINJAUAN PUSTAKA

Bab tinjauan pustaka berisi mengenai penjelasan teori yang berkaitan dengan implementasi perangkat lunak. Penjelasan tersebut bertujuan untuk memberikan gambaran mengenai sistem yang akan dibangun dan berguna sebagai pendukung dalam pengembangan perangkat lunak.

2.1. Sistem Rekomendasi

Sistem Rekomendasi merupakan sebuah sistem yang mengacu pada memprediksi sejumlah item maupun data untuk pengguna di masa mendatang, kemudian dijadikan rekomendasi item paling teratas. Salah satu alasan mengapa perlu digunakannya sistem rekomendasi karena pengguna memiliki banyak pilihan untuk digunakan karena prevalensi internet. [1]

Meskipun jumlah informasi yang tersedia meningkat, masalah baru muncul karena para pengguna kesulitan memilih item yang ingin mereka lihat. Di sinilah recommendation system masuk. Diantara banyak rekomendasi sistem, ada 2 macam rekomendasi sistem yang dilihat berdasarkan sudut pandangnya.

2.1.1. Collaborative Filtering

Collaborative Filtering memanfaatkan transaksi suatu produk / item yang didasarkan kepada perilaku / kebiasaan si pengguna. Tujuannya agar pengguna yang sama dan item yang serupa dapat disukai oleh pengguna sebagai rekomendasi pilihan. [1]

2.1.2. Content Based Recommendation

Content Based Recommendation memanfaatkan informasi beberapa item / data untuk direkomendasikan kepada pengguna sebagai referensi yang terkait dengan informasi yang digunakan sebelumnya. Tujuan dari content based recommendation agar dapat memprediksi persamaan dari sejumlah informasi yang didapat dari pengguna.

Contoh, seorang pengguna sedang menonton video di Youtube. Konten yang dilihat oleh pengguna, yaitu tentang sepak bola. Youtube secara sistem akan merekomendasikan si pengguna untuk melihat video lain yang berhubungan dengan konten sepak bola. [1]

Dalam pembuatannya, content based filtering menggunakan konsep perhitungan vector, TF-IDF, dan Cosine Similarity yang intinya dikonversikan dari data / text menjadi berbentuk vektor.

Jenis filter ini tidak melibatkan pengguna lain jika bukan diri kami sendiri. Berdasarkan dari apa yang kita sukai, algoritma hanya akan memilih item dengan konten yang mirip untuk direkomendasikan kepada kita.

Dalam hal ini akan ada lebih sedikit keragaman dalam rekomendasi, tetapi ini akan berfungsi baik pengguna memberikan rating atau menilai sesuatu atau tidak. [2]

2.2. *E-Learning*

E-learning adalah suatu sistem atau konsep pendidikan yang memanfaatkan teknologi informasi dalam proses belajar mengajar. Beberapa pengertian e-learning diantaranya adalah sebuah pembelajaran yang disusun dengan tujuan menggunakan sistem elektronik atau komputer sehingga mampu mendukung proses pembelajaran (Michael, 2013:27). [3]

2.3. *Moodle*

MOODLE (singkatan dari Modular Object-Oriented Dynamic Learning Environment) adalah paket perangkat lunak yang diproduksi untuk kegiatan belajar berbasis internet dan situs web yang menggunakan prinsip social constructionist pedagogy. MOODLE merupakan salah satu aplikasi dari konsep dan mekanisme belajar mengajar yang memanfaatkan teknologi informasi, yang dikenal dengan konsep pembelajaran elektronik atau e-learning. Moodle dapat digunakan secara bebas sebagai produk sumber terbuka (open source) di bawah lisensi GNU.

Moodle dapat diinstal di komputer dan sistem operasi apapun yang bisa menjalankan PHP dan mendukung database SQL. [4]

Pada Moodle sendiri terdapat plugin dengan berbagai tipe. Plugin sendiri merupakan folder yang terdiri dari skrip php. Dengan adanya plugin kita dapat mengembangkan Moodle tanpa harus memodifikasi aplikasi core Moodle itu sendiri. Terdapat 24 tipe plugin dalam Moodle, satu diantaranya adalah tipe *Blocks*, dimana tipe ini akan memperbolehkan sebuah plugin untuk menampilkan informasi atau *tools* yang dapat diletakkan pada halaman layar Moodle.

2.4. TF-IDF

Term Frequency — Inverse Document Frequency atau TF — IDF adalah suatu metode algoritma yang berguna untuk menghitung bobot setiap kata yang umum digunakan. Metode ini juga terkenal efisien, mudah dan memiliki hasil yang akurat. Metode ini akan menghitung nilai Term Frequency (TF) dan Inverse Document Frequency (IDF) pada setiap token (kata) di setiap dokumen dalam korpus. Secara sederhana, metode TF-IDF digunakan untuk mengetahui berapa sering suatu kata muncul di dalam dokumen.

Metode Term Frequency-Inverse Document Frequency (TF-IDF) adalah cara pemberian bobot hubungan suatu kata (term) terhadap dokumen. TF-IDF ini adalah sebuah ukuran statistik yang digunakan untuk mengevaluasi seberapa penting sebuah kata di dalam sebuah dokumen atau dalam sekelompok kata. Untuk dokumen tunggal tiap kalimat dianggap sebagai dokumen. Frekuensi kemunculan kata di dalam dokumen yang diberikan menunjukkan seberapa penting kata itu di dalam dokumen tersebut. Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Bobot kata semakin besar jika sering muncul dalam suatu dokumen dan semakin kecil jika muncul dalam banyak dokumen.

Pada algoritma TF-IDF digunakan rumus untuk menghitung bobot (W) masing masing dokumen terhadap kata kunci dengan rumus yaitu : [5]

$$W_{dt} = tf_{dt} * idf_t \quad (1)$$

Dimana :

W_{dt} = bobot dokumen ke d terhadap kata ke-t

tf_{dt} = banyak kata yang dicari pada sebuah dokumen

idf_t = *Inversed Document Frequency* ($\log(N/df)$)

df = banyak dokumen yang mengandung kata yang dicari

2.5. Cosine Similarity

Metode cosine similarity merupakan metode untuk menghitung kesamaan antara dua buah objek yang dinyatakan dalam dua buah vector dengan menggunakan keywords (kata kunci) dari sebuah dokumen sebagai ukuran. Cosine Similarity dapat diimplementasikan untuk menghitung nilai kemiripan antar kalimat dan menjadi salah satu teknik untuk mengukur kemiripan teks yang populer. [6]

$$Similarity = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i x B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (2)$$

Keterangan :

- A = vektor
- B = vector
- A_i = bobot *term i* dalam blok A_i
- B_i = bobot *term i* dalam blok B_i
- i = jumlah *term* dalam kalimat
- n = jumlah vektor

2.6. PHP

PHP merupakan bahasa pemrograman script server-side yang digunakan untuk mengembangkan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum. PHP di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang

dikelola oleh The PHP Group. Situs resmi PHP beralamat di <http://www.php.net>.

PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client). Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll.

Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: Hypertext Preprocessor. PHP dapat digunakan dengan gratis (free) dan bersifat Open Source. PHP dirilis dalam lisensi PHP License, sedikit berbeda dengan lisensi GNU General Public License (GPL) yang biasa digunakan untuk proyek Open Source. [7]

[Halaman ini sengaja dikosongkan]

BAB III

ANALISIS DAN PERANCANGAN

Bab ini membahas analisis dan perancangan dari perangkat lunak yang dibangun pada Tugas Akhir ini. Perancangan sistem akan mencantumkan analisis kebutuhan yang diperlukan di perangkat lunak setelah mendapatkan proses bisnis. Perancangan sistem ini akan direpresentasikan dengan diagram *Unified Modelling Language* (disebut UML).

3.1. Analisis

Subbab ini membahas analisis kebutuhan perangkat lunak, yang meliputi analisis perancangan perangkat lunak secara garis besar, deskripsi umum, dan kebutuhan perangkat lunak.

3.1.1. Analisis Permasalahan

Pada era globalisasi ini, penggunaan internet sebagai sarana penunjang untuk melakukan kegiatan belajar mengajar sudah semakin meningkat, penggunaan *e-learning* seringkali digunakan oleh pengajar untuk memasukkan materi pelajaran sebagai sarana pendukung . Selain itu dengan adanya *e-learning*, dapat mendukung terjadinya belajar secara mandiri oleh siswa dengan aktivitas yang lengkap. Namun, saat ini dari sistem *e-learning* berbasis Moodle masih belum ada pengembangan *plugin* untuk memberikan rekomendasi materi, yang dapat mendukung terjadinya pembelajaran mandiri oleh siswa.

Untuk mengatasi permasalahan tersebut, pada Tugas Akhir ini akan dilakukan pembuatan *plugin* untuk memberikan rekomendasi konten belajar pada Moodle secara otomatis agar siswa dapat terbantu untuk memilih konten belajar, sehingga proses pembelajaran menjadi efektif dan efisien. *Plugin* ini dikembangkan melalui Moodle dikarenakan Moodle merupakan aplikasi *Course Manajemen Sistem* (CMS) berbasis web yang biasanya digunakan pada *e-learning*.

Dengan Moodle, portal *e-learning* dapat dimodifikasi sesuai dengan kebutuhan. Dalam pengembangan *plugin* kali ini,

ditujukan untuk memberikan rekomendasi konten belajar dengan materi matematika kelas 4, 5, dan 6. Maka dari itu, Moodle yang akan dibuat merupakan Moodle dengan materi matematika, dimana setiap bab pelajaran dalam buku akan dijadikan sebagai *course*, dan plugin akan dibangun dengan menghitung kemiripan antar *course* didalam Moodle.

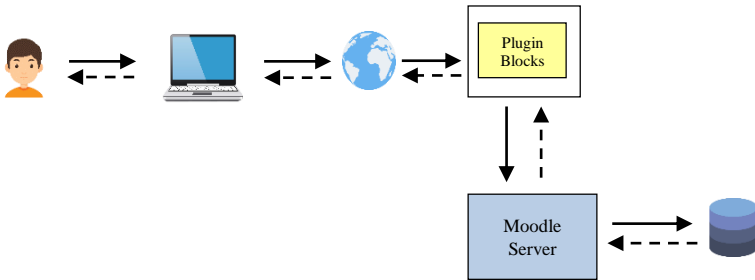
Content Based Filtering merupakan algoritma yang digunakan dalam pembangunan plugin, dimana sistem mengacu pada kemiripan antar dokumen dalam memberikan rekomendasi. Sehingga, pembangunan plugin pada Tugas Akhir kali ini akan memberikan rekomendasi dengan menghitung kemiripan dari sisi konten deskripsinya.

Dalam pembuatannya, *plugin* ini dikembangkan pada *plugin* bertipe *blocks* dimana *plugin* ini menyediakan tampilan data yang berguna di situs Moodle untuk siswa. Sedangkan untuk struktur *database plugin* didefinisikan dalam berkas *install.xml* pada folder *db* di *plugin report*. Setelah itu dilakukan pengolahan deskripsi setiap *course* agar bisa dihitung kemiripannya menggunakan *cosine similarity*. Administrator dapat melakukan instalasi plugin. Dengan adanya plugin ini, diharapkan siswa dapat melihat rekomendasi *course* untuk memudahkan pembelajaran secara efektif dan efisien.

Tugas Akhir ini hanya menjelaskan secara detail perhitungan rekomendasi konten belajar.

3.1.2. Deskripsi Umum Perangkat Lunak

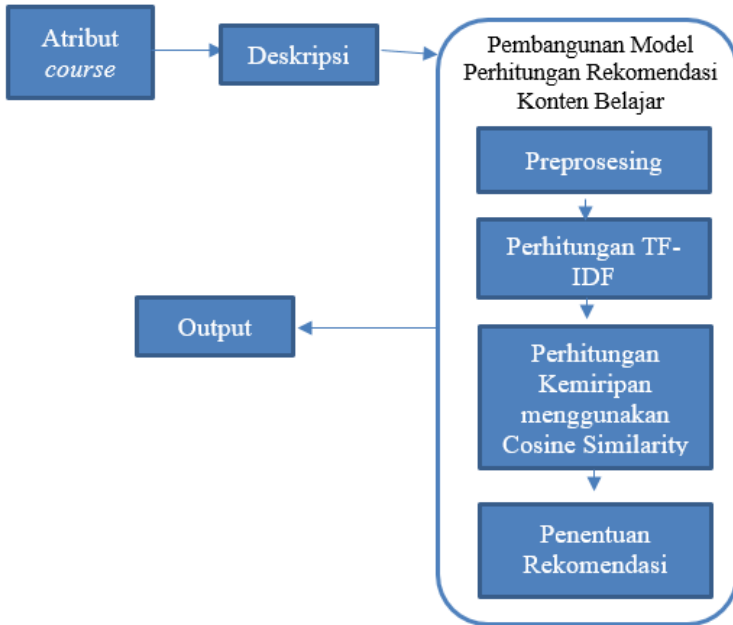
Pada Tugas Akhir ini akan dibuat sebuah *plugin* rekomendasi konten belajar pada Moodle. Pada Gambar 3.1 dijelaskan bahwa informasi yang diakses oleh pengguna pada *plugin blocks* merupakan hasil dari pengolahan data yang diambil dari *database*.



Gambar 3.1 Deskripsi Umum Sistem

Sedangkan rancangan perhitungan rekomendasi konten belajar dalam Moodle dimulai dengan mengolah deskripsi pada tiap konten. Kemudian dilakukan pengolahan data sampai menghasilkan nilai kemiripan antar deskripsi. Rancangan model perhitungan digambarkan pada

Gambar 3.2.



Gambar 3.2 Rancangan Model Perhitungan Rekomendasi Course Dalam Moodle

3.1.2.1. Perhitungan Rekomendasi Konten Belajar untuk Course

Dalam menghitung rekomendasi, digunakan algoritma *content based filtering* dimana algoritma tersebut merekomendasikan konten yang memiliki kesamaan akan konten yang disukai pengguna. Maka akan dibuat rekomendasi yang mencari kemiripan *course* yang sedang diambil oleh siswa. Dimana dalam studi kasus ini, *course* yang akan digunakan adalah topik/materi dari bab pelajaran matematika kelas 4, 5, dan 6.

Sebelum melakukan proses perhitungan rekomendasi konten belajar untuk *course*, terlebih dahulu dilakukan proses pengolahan data *course*, hal ini dilakukan karena di dalam Moodle belum menyediakan data yang sesuai kebutuhan Tugas Akhir ini. Dalam Moodle, data data *course* tersimpan didalam database pada table *mdl_course*. Tabel ini memiliki atribut *id_course* sebagai *primary key*, *category*, *sortorder*, *fullname*, *shortname*, *idnumber*, *summary*, *summaryformat*, *format*, *showgrades*, *newsitems*, *startdate*, *enddate*, *marker*, *maxbytes*, *legacyfiles*, *showreports*, *visible*, *visibleold*, *groupmode*, *groupmodeforce*, *defaultgroupingid*, *lang*, *calendartype*, *theme*, *timecreated*, *timemodified*, *requested*, *enablecompletion*, *completionnotify* dan *cacherev*.

3.1.2.1.1. Menghitung Bobot TF-IDF

Menghitung bobot TF-IDF dilakukan berdasarkan persamaan:

$$Wdt = tfdt * idft \quad (1)$$

Dimana :

Wdt = bobot dokumen ke d terhadap kata ke-t

Tfdt = banyak kata pada sebuah dokumen

Idft = *Inversed Document Frequency* ($\log(N/df)$)

df = banyak dokumen yang mengandung kata yang dicari

untuk mendapatkan TF, setiap kata pada deskripsi *course* dihitung berapa kali kemunculannya, kemudian dikalikan dengan log dari total dokumen dibagi dengan dokumen yang mengandung kata tersebut.

Tabel 3.1 Contoh data dari mdl_index dengan course Statistika

kata	tf	tf*log(N/df)
statistika	2	479,579
dalam hidupan	1	23,979
seharihari	1	23,979

sangat	1	23,979
dibutuhkan	1	23,979
misalnya	2	34,095
bagaimana	1	170,475
suatu	2	34,095
data	3	389,785
terkumpul	1	23,979
disajikan	1	23,979
dalam	2	157,691
bentuk	1	10,116
lebih	1	23,979
mudah	1	23,979
dibaca	1	23,979
dipahami	1	23,979
dunia	1	23,979
pendidikan	1	23,979
penerapan	1	23,979
tentang	1	10,116
penyajian	3	511,424
banyak	2	20,232
siswa	3	719,369

3.1.2.1.2. Menghitung Cosine Similarity

Untuk menghitung cosine similarity, menggunakan persamaan :

$$\text{Similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} \frac{\sum_{i=1}^n A_i x B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (2)$$

Keterangan :
 A = course A
 B = course B
 Ai = bobot *term i* dalam blok Ai

B_i = bobot *term* i dalam blok B_i
 i = jumlah *term* dalam kalimat
 n = jumlah vector

dari persamaan diatas, $\sum_{i=1}^n A_i \times B_i$ adalah perkalian skalar antara bobot setiap term pada course A, dengan bobot setiap term dari course B, sedangkan persamaan $\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}$ adalah perkalian dari panjang vector A dan B yang telah di akar kuadratkan.

3.1.3. Spesifikasi Kebutuhan Perangkat Lunak

Subbab ini membahas spesifikasi kebutuhan perangkat lunak dari hasil analisis yang telah dilakukan. Bagian ini berisi kebutuhan perangkat lunak yang direpresentasikan dalam bentuk kebutuhan fungsional, kebutuhan non-fungsional, diagram kasus penggunaan, dan diagram aktivitas.

3.1.3.1. Aktor

Aktor adalah pihak-pihak yang terlibat dan berinteraksi langsung dengan sistem. Aktor yang berada pada sistem ini dibahas lebih rinci pada Tabel 3.2.

Tabel 3.2 Spesifikasi Karakteristik Aktor pada Sistem

No	Aktor	Tugas	Hak Akses
1	Pengajar	<ul style="list-style-type: none"> Membuat <i>course</i> Menambahkan Plugin 	<ul style="list-style-type: none"> Mengelola (melihat, menambah, mengubah, menghapus) data <i>course</i> Menambahkan plugin
2	Siswa	<ul style="list-style-type: none"> Melihat <i>course</i> 	<ul style="list-style-type: none"> Melihat data <i>course</i>

3.1.3.2. Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan layanan yang harus dimiliki oleh perangkat lunak, reaksi terhadap suatu masukan, hasil

yang dilakukan pada situasi khusus. Kebutuhan fungsional dari perangkat lunak untuk aplikasi ini dijelaskan pada Tabel 3.3.

Tabel 3.3 Kebutuhan Fungsional Sistem

Kode	Kebutuhan Fungsional	Deskripsi
F01	Menambahkan fitur rekomendasi pada <i>course</i>	Menambahkan fitur rekomendasi <i>course</i> pada <i>course</i> yang tersedia pada Moodle
F02	Melihat rekomendasi <i>course</i>	Melihat rekomendasi <i>course</i> yang ditampilkan

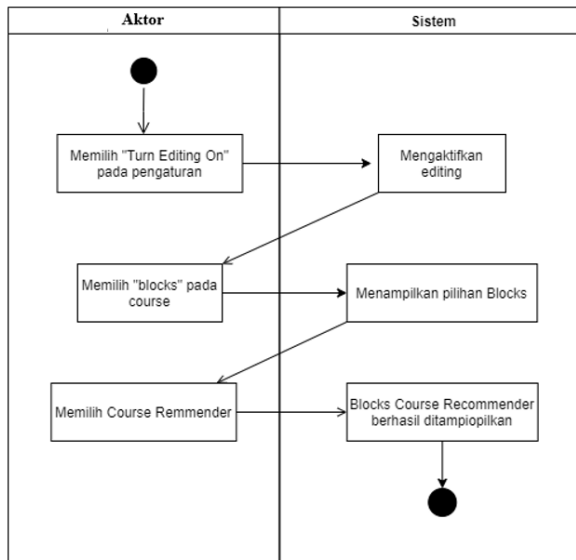
3.1.3.2.1. Kasus Penggunaan Menambahkan Fitur Rekomendasi Course

Kasus penggunaan ini diakses ketika pengajar hendak membuat *course* yang didalamnya ingin ditambahkan fitur rekomendasi untuk mempermudah siswa dalam melakukan aktifitas pembelajaran. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.4.

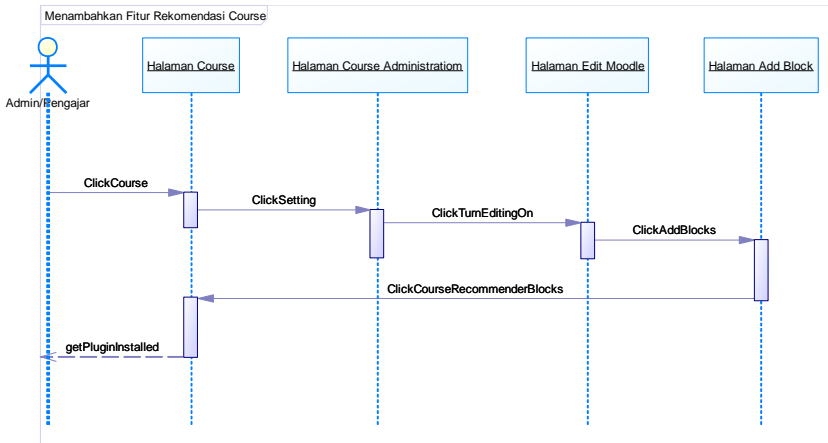
Tabel 3.4 Spesifikasi Kasus Menambahkan Fitur Rekomendasi Course

Nama	Menambahkan Fitur Rekomendasi <i>Course</i>	
Nomor	UC001	
Deskripsi	Pengajar dapat menambahkan fitur rekomendasi <i>course</i>	
Aktor	Pengajar	
Kondisi Awal	Fitur rekomendasi <i>course</i> belum dapat ditampilkan	
Kondisi Akhir	Fitur rekomendasi <i>course</i> sudah dapat diakses	
Alur Normal		
	Aktor	Sistem
1.	Administrator memilih “Turn editing on” pada pengaturan	

	2. Sistem mengaktifkan mode editing
3. Administrator memilih "Blocks" pada <i>course</i>	
	4. Sistem menampilkan pilihan Blocks
5. Administrator memilih "Course Recommender"	
	6. Blocks Course Recommender berhasil ditampilkan
Alur Alternatif	
-	



Gambar 3.3 Diagram Aktivitas UC001



Gambar 3.4 Sequence Diagram Aktivitas UC001

Skenario aktivitas dari kasus penggunaan selanjutnya digambarkan ke dalam diagram aktivitas yang dapat dilihat pada Gambar 3.3 dan Sequence Diagram dapat dilihat pada Gambar 3.4.

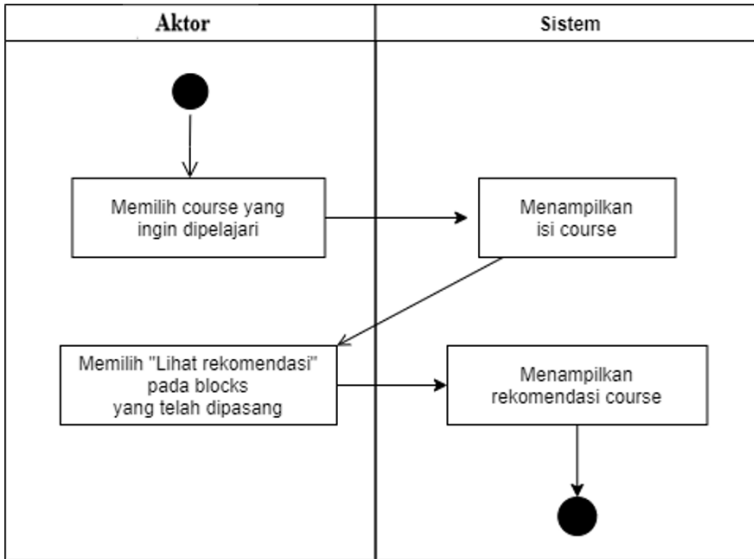
3.1.3.2.2. Kasus Penggunaan Melihat Rekomendasi Course

Kasus penggunaan melihat rekomendasi *course* adalah kasus penggunaan yang memungkinkan pengguna untuk melihat rekomendasi *course* dari pemasangan plugin yang telah ditambahkan sebelumnya. Spesifikasi kasus penggunaan dapat dilihat pada Tabel 3.5 dan diagram aktivitas pada Tabel 3.5.

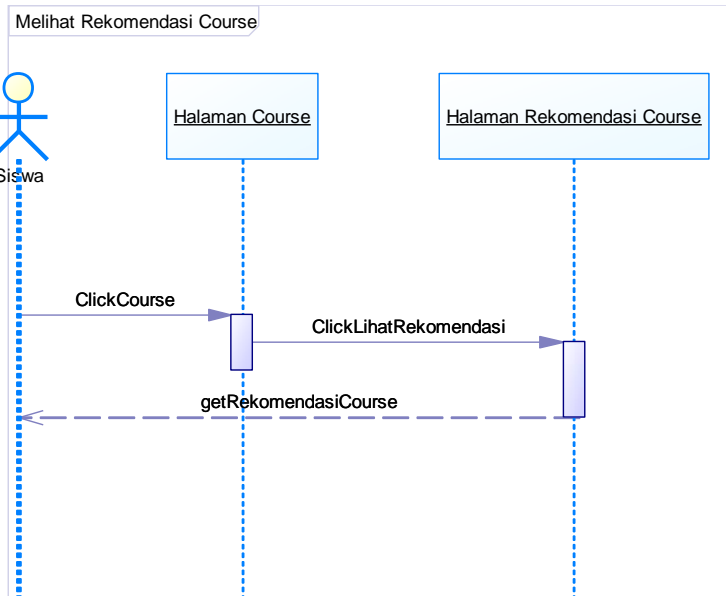
Tabel 3.5 Spesifikasi Kasus Penggunaan Melihat Rekomendasi Course

Nama	Melihat Rekomendasi <i>Course</i>
Nomor	UC002
Deskripsi	Pengguna dapat melihat rekomendasi <i>course</i>
Aktor	Siswa

Kondisi Awal	Plugin Blocks Course Recommender telah terpasang pada course yang ada.
Kondisi Akhir	Rekomendasi <i>course</i> dapat dilihat oleh pengguna
Alur Normal	
Aktor	Sistem
1. Siswa memilih course yang akan dipelajari 3. Siswa memilih “Lihat rekomendasi” pada blocks yang telah dipasang	2. Sistem menampilkan isi course 4. Sistem menampilkan rekomendasi <i>course</i>
Alur Alternatif	
-	
Aktor	Sistem
-	-



Gambar 3.5 Diagram Aktivitas UC001



Gambar 3.6 Sequence Diagram Aktivitas UC002

Berdasarkan skenario kasus penggunaan pada Tabel 3.5 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus melihat rekomendasi *course* dapat dilihat pada **Error! Reference source not found.** dan Sequence Diagram dapat dilihat pada Gambar 3.6.

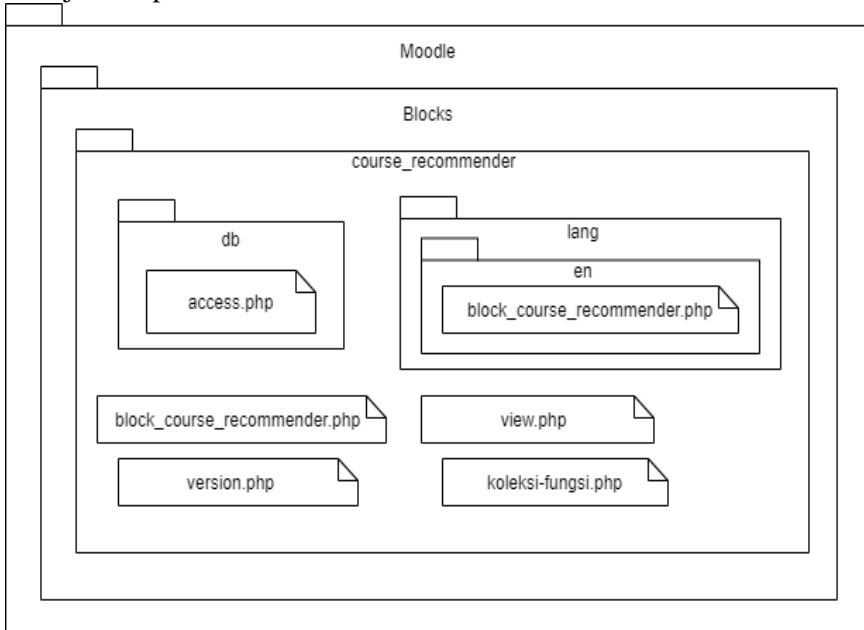
3.2. Perancangan

Pada subbab perancangan akan dijelaskan mengenai arsitektur sistem yang digunakan, perancangan diagram kelas, perancangan basis data, dan perancangan antarmuka.

3.2.1. Perancangan Arsitektur Sistem

Dari beberapa tipe *plugin* yang terdapat pada Moodle, dalam Tugas Akhir ini akan dibangun *plugin* bertipe *blocks* untuk penilaian rekomendasi *course*. *Plugin* yang akan dibangun

bernama ‘*course_recommender*’. Arsitektur dari *plugin* ini ditunjukkan pada Gambar 3.7.



Gambar 3.7 Ilustrasi Arsitektur Sistem Web

Karena *plugin* yang dibangun bertipe *blocks*, maka *plugin* harus dibangun di dalam direktori *blocks* pada Moodle, yang diberi nama direktori *blocks_courserecommender*. Pada direktori *blocks_courserecommender* terdapat direktori *db* untuk menyimpan struktur *database* pada *plugin*. Berkas *db/access.php* bertujuan untuk mendefinisikan apa saja yang dapat dilakukan user berdasarkan *role* nya.

Kemudian terdapat direktori *lang/en/* *blocks_courserecommender.php* untuk mendefinisikan string dalam bahasa Inggris pada *plugin* *course_recommender*.

Kemudian terdapat berkas *blocks_courserecommender.php* untuk menyimpan *class* yang digunakan, dan agar dapat ditampilkan pada layar. Berkas *view.php* untuk menampilkan

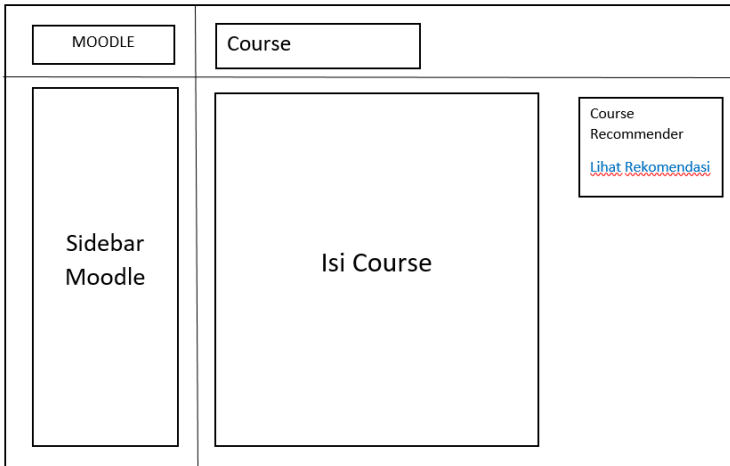
perhitungan yang dilakukan guna mendapatkan rekomendasi *course*. Berkas *settings.php* untuk konfigurasi administrasi pada *plugin course_recommender*. Berkas *version.php* mendefinisikan versi *plugin* yang digunakan.

3.2.2. Perancangan Antarmuka Pengguna

Pada bagian ini dijelaskan mengenai rancangan tampilan antarmuka pengguna dari sistem. Antarmuka pengguna yang berhubungan langsung dengan aktor, harus memiliki tampilan yang menarik bagi penggunanya. Aplikasi ini menggunakan antarmuka yang indah dilihat. Pengguna dapat mengakses menggunakan semua *web* browser dan hanya menggunakan keyboard dan mouse. Berikut akan dijelaskan masing rancangan antarmuka.

3.2.2.1. Rancangan Antarmuka Menambahkan Fitur Rekomendasi Course

Halaman menambahkan fitur rekomendasi *course* akan digunakan pada kasus penggunaan menambahkan fitur rekomendasi *course*. Halaman ini menampilkan *plugin* pada *course* yang akan digunakan untuk melihat rekomendasi *course*. Rancangan dan penjelasan terkait antarmuka menambahkan fitur rekomendasi *course* dapat dilihat pada Gambar 3.8 dan Tabel 3.6.



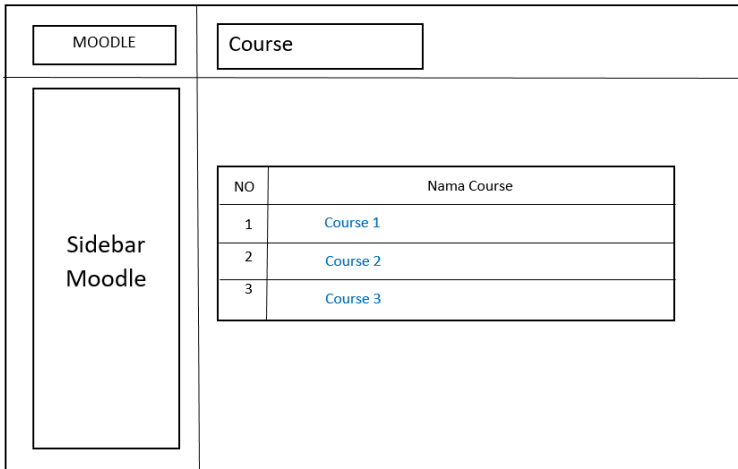
Gambar 3.8 Rancangan Antarmuka Menambahkan Fitur Rekomendasi Course

Tabel 3.6 Penjelasan Antarmuka Menambahkan Fitur Rekomendasi Course

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan/ Keluaran
1	<i>linkLihatRekomendasi</i>	<i>Text</i>	Menampilkan rekomendasi course	<i>linkClicked</i>

3.2.2.2. Rancangan Antarmuka Melihat Rekomendasi Course

Halaman melihat rekomendasi *course* akan digunakan pada kasus penggunaan melihat rekomendasi *course*. Halaman ini berisi daftar rekomendasi *course* yang didapatkan dari perhitungan oleh plugin. Setelah siswa memilih “lihat rekomendasi”, sistem akan menampilkan halaman dimana siswa dapat memilih *course* yang ditampilkan sebagai rekomendasi. Rancangan dan penjelasan terkait antarmuka melihat pengajuan petani mitra dapat dilihat pada **Error! Reference source not found.** dan Tabel 3.7.



Gambar 3.9 Rancangan Halaman Antarmuka Melihat Rekomendasi *Course*

Tabel 3.7 Penjelasan Antarmuka Melihat Rekomendasi *Course*

No	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Jenis Masukan/ Keluaran
1	<i>tableNamaCourse</i>	<i>Tabel</i>	Menampilakn hasil perhitungan rekomendasi	-
2	<i>linckCourse</i>	<i>Text</i>	Mengantarkan halaman menuju halaman course yang direkomendasikan	<i>Link Clicked</i>

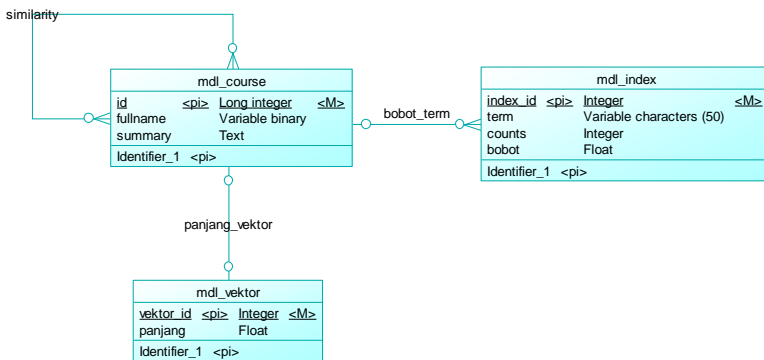
3.2.3. Perancangan Data

Dalam pembuatan Tugas Akhir ini, diperlukan analisis kebutuhan berupa perancangan basis data. Basis data yang digunakan adalah phpMyAdmin.

Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (CDM) pada Gambar 3.10 dan *Physical Data Model* (PDM) pada Gambar 3.11.

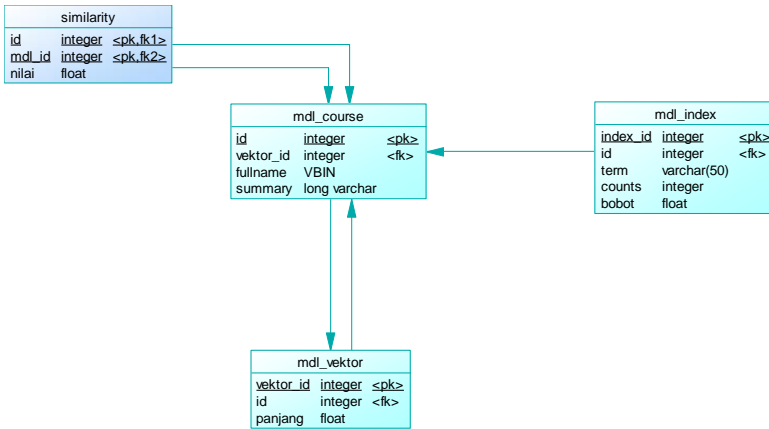
3.2.3.1. Perancangan Basis Data

Dalam membuat suatu aplikasi berupa sistem informasi, diperlukan sebuah analisis berupa perancangan basis data. MySQL dipilih menjadi aplikasi basis data karena penggunaannya yang mudah.



Gambar 3.10 CDM Sistem

Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM). Secara umum, struktur CDM dan PDM dapat dilihat pada **Error! Reference source not found.** dan **Error! Reference source not found.**



Gambar 3.11 PDM Sistem

3.2.3.1.1. Tabel mdl_course

Tabel mdlnr_course adalah tabel bawaan dari Moodle yang menyimpan *course* sebagai mata pelajaran. Deskripsi rinci tabel users dapat dilihat pada Tabel 3.8.

Tabel 3.8 Deskripsi Rinci Tabel mdl_course

Id Field	Deskripsi	Tipe & length	Keterangan
id	Variabel untuk ID <i>course</i> .	Integer	Primary Key
fullname	Variabel untuk nama <i>course</i> .	Varchar(255)	
summary	Variabel untuk deskripsi <i>course</i> .	Text	

3.2.3.1.2. *Tabel mdl_index*

Tabel *mdl_index* digunakan untuk menyimpan setiap term yang ada pada setiap dokumen *course*, dan menyimpan bobot tiap term tersebut. Deskripsi rinci tabel *mdl_index* dapat dilihat pada Tabel 3.9.

Tabel 3.9 Deskripsi Rinci Tabel Jenis *mdl_index*

Id Field	Deskripsi	Tipe & length	Keterangan
index_id	Variabel untuk ID <i>mdl_index</i>	Integer	Primary Key
term	Variabel untuk menyimpan setiap term dalam deskripsi	Varchar(255)	
docid	Variabel untuk menyimpan ID <i>course</i>	Integer	
counts	Variabel untuk menyimpan jumlah frekuensi kemunculan term	Integer	
bobot	Variabel untuk menyimpan bobot tiap term	Float	

3.2.3.1.3. *Tabel mdl_vektor*

Tabel *mdl_vektor* digunakan untuk menyimpan panjang vector tiap *course*. Deskripsi rinci tabel *mdl_vektor* dapat dilihat pada Tabel 3.10.

Tabel 3.10 Deskripsi Rinci Tabel *mdl_vektor*

Id Field	Deskripsi	Tipe & length	Keterangan
vector_id	Variabel untuk ID <i>mdl_vektor</i>	Integer	Primary Key
docid	Variabel ID <i>course</i>	Integer	Foreign Key
panjang	Variabel untuk menyimpan panjang vektor	Float	

3.2.3.1.4. *Tabel mdl_similarity*

Tabel *mdl_similarity* digunakan untuk menyimpan nilai dari nilai cosine similarity dan *course* yang dihitung. Deskripsi rinci tabel *mdl_similarity* dapat dilihat pada Tabel 3.811.

Tabel 3.11 Deskripsi Rinci Tabel *mdl_similarity*

Id Field	Deskripsi	Tipe & length	Keterangan
docid1	Variabel untuk ID <i>course</i> .	Integer	Primary Key
docid2	Variabel untuk ID <i>course</i> .	Integer	Foreign Key
nilai	Variabel untuk menyimpan nilai kemiripan menggunakan <i>Cosine Similarity</i>	Float	

[Halaman ini sengaja dikosongkan]

BAB IV IMPLEMENTASI

Bab ini membahas implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya. Sebelum penjelasan implementasi akan ditunjukkan terlebih dahulu lingkungan untuk melakukan implementasi.

Pada bagian implementasi ini juga akan dijelaskan mengenai fungsi-fungsi yang digunakan dalam program Tugas Akhir ini dan disertai dengan kode semu masing-masing fungsi utama.

4.1. Lingkungan Implementasi

Lingkungan implementasi sistem yang digunakan untuk mengembangkan Tugas Akhir memiliki spesifikasi perangkat keras dan perangkat lunak seperti yang ditampilkan pada Tabel 4.1

Tabel 4.1 Lingkungan Implementasi

Perangkat	Spesifikasi
Perangkat keras	Prosesor: Intel® Core™ i7-7700HQ CPU @ 2.80GHz (4 CPUs) , ~2.81GHz Memori: 16000 MB
Perangkat lunak	Sistem Operasi: Microsoft Windows 10 Pro Education 64-bit Perangkat Pengembang: Laragon, Visual Studio Code, HeidiSQL, Postman Perangkat Perancang Diagram: Sybase Power Designer 16 Perangkat Database: HeidiSQL

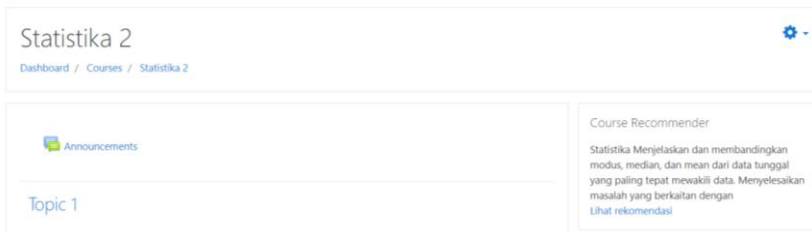
4.2. Implementasi Antarmuka Sistem

Berikut ini akan dijelaskan mengenai implementasi antarmuka sistem yang sudah direalisasikan.

4.2.1. Halaman Menambahkan Fitur Rekomendasi Course

Halaman ini merupakan halaman yang digunakan untuk melakukan kasus penggunaan UC001 yaitu menambahkan fitur rekomendasi *course*. Di dalam halaman ini terdapat *plugin blocks* yang telah ditambahkan pada layar *course*. Realisasi dari halaman ini dapat ditunjukkan pada Gambar 4.1 dan kode sumber fungsi menambahkan fitur rekomendasi *course* dapat dilihat pada Error! Reference source not found..

Kode sumber implementasi Menambahkan Fitur Rekomendasi Course dapat dilihat pada subbab 4.3.



Gambar 4.1 Implementasi Antarmuka Menambahkan Fitur Rekomendasi Course

4.2.2. *Halaman Melihat Hasil Rekomendasi Course*

Halaman antarmuka hasil rekomendasi *course* pada Gambar 4.2, merupakan halaman yang akan ditampilkan pada *plugin* ketika pengguna menekan tombol “lihat rekomendasi” dari tampilan awal *course*. Halaman antarmuka ini mengacu pada kasus penggunaan nomor UC-002.



Rekomendasi Course	Value
Pengumpulan dan Penyajian Data	0.103622
Skala	0.0830987
Statistika	0.0689592

Gambar 4.2 Implementasi Melihat Hasil Rekomendasi Course

4.3. *Implementasi Kasus Penggunaan*

Pada subbab ini akan dijelaskan mengenai implementasi pengambilan data dengan *electronic nose*, proses klasifikasi data sampel mulai dari proses latihan hingga proses klasifikasi kadar boraks yang terkandung pada data sampel.

4.3.1. *Implementasi Menambahkan Fitur Rekomendasi Course*

Kode sumber dari implementasi kasus penggunaan memonitor tingkat pembelajaran bermakna ditunjukkan pada Kode Sumber 4.1. Untuk penjelasan kode sumber ditunjukkan pada Tabel 4.2.

```
1. public function get_content() {
2.     global $USER, $COURSE, $CFG, $OUTPUT, $DB;
```

```

3.     require_once($CFG->libdir . '/filelib.php');
4.
5.     if ($this->content !== null) {
6.         return $this->content;
7.     }
8.
9.     if (empty($this->instance)) {
10.        return '';
11.    }
12.
13.    $this->content = new stdClass();
14.    $options = new stdClass();
15.    $options->noclean = true
16.    $options->overflowdiv = true;
17.    $context = context_course::instance($this->page-
>course->id);
18.    $this->page->course->summary =
file_rewrite_pluginfile_urls($this->page->course-
>summary, 'pluginfile.php', $context->id, 'course',
'summary', NULL);
19.    $this->content->text = format_text($this->page-
>course->summary, $this->page->course->summaryformat,
$options);
20.    $url = new
moodle_url('/blocks/course_recommender/view.php',
array('blockid' => $this->instance->id, 'courseid' =>
$COURSE->id));
21.    $this->content->footer = html_writer::link($url,
get_string('lihatrekom', 'block_course_recommender'));
22.    return $this->content;
23. }

```

Kode Sumber 4.1.Fungsi Menambahkan Fitur Rekomendasi *Course*

Tabel 4.2.Implementasi Fungsi Menambahkan Fitur Rekomendasi *Course*

No. Baris	Kegunaan
11-27	Menggenerate content moodle kedalam plugin blocks
28-31	Memanggil <i>view</i> untuk menampilkan plugin blocks

4.3.2. Implementasi Melihat Hasil Rekomendasi Course

Kode sumber dari implementasi kasus penggunaan Melihat Hasil Rekomendasi Course ditunjukkan pada Kode Sumber 4.2 Untuk penjelasan kode sumber ditunjukkan pada Tabel 4.3.

```

1. <?php
2.
3. require_once('.././config.php');
4. require_once('koleksi-fungsi.php');
5.
6. global $DB, $OUTPUT, $PAGE, $CFG;
7.
8. // Check for all required variables.
9. $courseid = required_param('courseid', PARAM_INT);
10.
11.
12. if (!$course = $DB->get_record('course', array('id' =>
    $courseid))) {
13.     print_error('invalidcourse',
        'block_course_recommender', $courseid);
14. }
15.
16. require_login($course);
17. $PAGE->set_url('/blocks/course_recommender/view.php',
    array('id' => $courseid));
18. $PAGE->set_pagelayout('standard');
19. $PAGE->set_heading(get_string('openingplugin',
    'block_course_recommender'));
20.
21.
22. //MELAKUKAN PREPROSESING PADA DESKRIPSI//
23.
24. //truncate table mdl_index
25. $sqltruncate = "TRUNCATE TABLE {index}";
26. $DB->execute($sqltruncate);
27. $sql = "SELECT mc.id, mc.fullname, mc.summary   from
    mdl_course mc";
28. $percobaan = $DB->get_records_sql($sql);
29. foreach ($percobaan as $value){

```

```

30.   $mcId = $value->id;
31.   $judul = $value->fullname;
32.   $summ = $value->summary;
33.
34.   //preproses
35.   $berita = preproses($summ);
36.
37.   //simpan ke inverted index (tbindex)
38.   $aberita = explode(" ", trim($berita));
39.
40.   $count='0';
41.
42. //MELAKUKAN PREPROSESING//
43.
44.   foreach ($aberita as $j => $value) {
45.     //hanya jika Term tidak null atau nil, tidak kosong
46.     if ($aberita[$j] != "") {
47.       //berapa baris hasil yang
dikembalikan query tersebut?
48.       $Srescount = "SELECT counts, id FROM
mdl_index WHERE term = '$aberita[$j]' AND docid = $mcId";
49.       $rescount = $DB-
>get_records_sql($Srescount);
50.       $crescount = count($rescount);
51.       //jika sudah ada DocId dan Term
tersebut , naikkan Count (+1)
52.       if ($crescount > 0) {
53.         $sql1 = $DB-
>get_records_sql($Srescount);
54.         foreach ($sql1 as $rowcount){
55.           $id = $rowcount->id;
56.           $countnya = $rowcount-
>counts;
57.           $countnya++;
58.           $DB->execute("UPDATE
{index} SET counts = $countnya WHERE term =
'{$aberita[$j]}' AND docid = $mcId");
59.         }
60.
61.       }
62.     } else {

```

```

63.             $sqldex = "INSERT INTO {index}
    (term, docid, counts, bobot) VALUES (?, ?, ?, ?)";
64.             $paramsdex = array($aberita[$j],
    $mcId, '1', '0');
65.             $DB->execute($sqldex, $paramsdex);
66.         }
67.     } //end if
68. }
69.}
70.//--END OF PREPROCESSING--//
71.
72.//START MENGHITUNG BOBOT//
73.//menghitung total dokumen melalui docid
74.$Sresn = "SELECT DISTINCT docid FROM mdl_index";
75.$sqln = $DB->get_records_sql($Sresn);
76.$n = count($sqln);
77.
78.//ambil setiap record dalam tabel tbindex
79.//hitung bobot untuk setiap Term dalam setiap docid
80.$sqlBobot = "SELECT * FROM mdl_index ORDER BY id";
81.$resBobot = $DB->get_records_sql($sqlBobot);
82.
83.foreach($resBobot as $rowBobot) {
84.    //$w = tf * log (n/N)
85.    $term = $rowBobot->term;
86.    $tf = $rowBobot->counts;
87.    $id = $rowBobot->id;
88.    $SresNTerm = "SELECT * FROM mdl_index WHERE term =
    '$term'";
89.    $resNTerm = $DB->get_records_sql($SresNTerm);
90.    $NTerm=count($resNTerm);
91.    $w = $tf * log($n/$NTerm);
92.    //update bobot dari term tersebut
93.    $DB->execute("UPDATE {index} SET bobot = $w WHERE id
    = $id");
94.}
95.//END MENGHITUNG BOBOT
96.
97.
98.// START MENGHITUNG PANJANG VEKTOR//
99.//truncate table mdl_vektor
100.    $sqltruncatev = "TRUNCATE TABLE {vektor}";

```

```

101.  $DB->execute($sqltruncatev);
102.  $SresDocId = "SELECT DISTINCT docid FROM mdl_index";
103.  $resDocId = $DB->get_records_sql($SresDocId);
104.
105.  foreach ($resDocId as $dodo){
106.      $docid = $dodo->docid;
107.
108.      //term utk unique output
109.      $SresVektor = "SELECT term, bobot FROM
mdl_index WHERE docid = $docid";
110.      $resVektor = $DB-
>get_records_sql($SresVektor);
111.
112.      //jumlahkan semua bobot kuadrat
113.      $panjangVektor = 0;
114.      foreach($resVektor as $rowVektor) {
115.          $b=$rowVektor->bobot;
116.          // $table->data[] = array($docid,$b);
117.          $panjangVektor = $panjangVektor +
($rowVektor->bobot * $rowVektor->bobot);
118.      }
119.
120.      //hitung akarnya
121.      $panjangVektor = sqrt($panjangVektor);
122.
123.
124.      // //masukkan ke dalam tbvektor
125.      $sqlv = "INSERT INTO {vektor} (docid,
panjang) VALUES (?, ?)";
126.      $params = array($docid, $panjangVektor);
127.      $DB->execute($sqlv, $params);
128.
129.  } //end foreach $rowDocId
130.  //--END MENGHITUNG PANJANG VEKTOR--// (clear)
131.
132.
133.
134.  //START MENGHITUNG COSINE SIMILARITY//
135.  //truncate similarity
136.  $sqltruncatesim = "TRUNCATE TABLE {similarity}";
137.  $DB->execute($sqltruncatesim);
138.  // ambil panjang vektor courseid

```



```

139. $vektorb = "SELECT panjang FROM `mdl_vektor` WHERE
    docid=$courseid";
140. $Sresvektorb= $DB->get_records_sql($vektorb);
141. foreach ($Sresvektorb as $resvektorb) {
142.     $panjangdocidb = $resvektorb->panjang;
143. }
144.
145. //get the boobot for each term where docid = id course
146. $sqlb= "SELECT * FROM `mdl_index` WHERE
    docid=$courseid";
147. $resBot= $DB->get_records_sql($sqlb);
148. $totaltermb=count($resBot);
149.
150. $queryb=array();
151. foreach($resBot as $valueb ) {
152.     $idb = $valueb->id;
153.     $docidb = $valueb->docid;
154.     $termb = $valueb->term;
155.     $bobotb = $valueb->bobot;
156.
157.     $queryb[]=$termb;
158.     $bobotqueryb[]=$bobotb;
159. }
160. //ambil setiap term dari DocId, bandingkan dengan
    term course
161. //ambil docid kecuali courseid
162. $sqldocidx = "SELECT * FROM mdl_vektor WHERE docid
    <> $courseid ORDER BY docid";
163. $resdocidx = $DB->get_records_sql($sqldocidx);
164. foreach ($resdocidx as $rowDocId) {
165.     $dotproduct = 0;
166.     $docidx = $rowDocId->docid;
167.     $panjangdocidx = $rowDocId->panjang;
168.     $SresTerm = "SELECT * FROM `mdl_index` WHERE
    docid=$docidx";
169.     $resTerm = $DB->get_records_sql($SresTerm);
170.     foreach ($resTerm as $rowTerm) {
171.         for ($i=0; $i<count($queryb); $i++) {
172.             //jika term sama
173.             if ($rowTerm->term ==
    $queryb[$i]) {

```

```

174.                                     $dotproduct           =
    $dotproduct + $rowTerm->bobot * $bobotqueryb[$i];
175.                                     } //end if
176.                                 } //end for $i
177.                             } //end while ($rowTerm)
178.                             if ($dotproduct > 0) {
179.                                 $sim = $dotproduct / ($panjangdocidb
    * $panjangdocidx);
180.                                 //simpan kemiripan > 0 ke dalam
    mdl_similarity
181.                                 $sqlv = "INSERT INTO {similarity}
    (docid1, docid2, nilai) VALUES (?, ?, ?)";
182.                                 $params = array($courseid, $docidx,
    $sim);
183.                                 $DB->execute($sqlv, $params);
184.                             }
185.
186.     } //end while $resdocidx
187.     //--END MENGHIUTNG COSINE SIMILARITY--//
188.
189.     //MENAMPILKAN REKOMENDASI
190.     $sqlfinal="SELECT mc.id, mc.fullname, ms.nilai FROM
    mdl_similarity ms, mdl_course mc WHERE ms.docid2=mc.id
    ORDER BY nilai DESC LIMIT 3";
191.     $sfinal = $DB->get_records_sql($sqlfinal);
192.     $table = new html_table();
193.     $table->head = array('Rekomendasi Course', 'Value');
194.     foreach ($sfinal as $final) {
195.         $hasilid=$final->id;
196.         $hasilcourse=$final->fullname;
197.         $nilai=$final->nilai;
198.         $linkcourse='<a
                                class
    href="http://localhost/server/moodle/course/view.php?id
    ='.$hasilid.'">'.$hasilcourse.'</a>';
199.         $table->data[] = array($linkcourse,$nilai);
200.     }
201.
202.
203.     echo $OUTPUT->header();
204.     // $simplehtml->display();
205.     if(!empty($table->data)) {
206.         echo html_writer::table($table);

```

```

207.     }
208.
209.     echo $OUTPUT->footer();
210.
211.     ?>

```

Kode Sumber 4.2. Fungsi Melihat Hasil Rekomendasi *Course*

Tabel 4.3 Implementasi Fungsi Melihat Hasil Rekomendasi *Course*

No. Baris	Kegunaan
22-70	Melakukan preprocessing terhadap deskripsi <i>course</i> Moodle
72-95	Melakukan perhitungan bobot pada setiap kata dalam dokumen deskripsi <i>course</i>
98-130	Melakukan perhitungan panjang vector tiap <i>course</i>
134-187	Melakukan perhitungan cosine similarity
189-200	Menampilkan rekomendasi

[Halaman ini sengaja dikosongkan]

BAB V PENGUJIAN DAN EVALUASI

Bab ini membahas tentang pengujian dan evaluasi pada perangkat lunak yang dibangun untuk Tugas Akhir ini. Pengujian dilakukan pada kasus penggunaan dari sistem perangkat lunak.

5.1 *Lingkungan Pengujian*

Pada proses pengujian perangkat lunak, dibutuhkan suatu lingkungan pengujian yang sesuai dengan standar kebutuhan. Lingkungan pengujian dalam Tugas Akhir ini dilakukan pada setiap kasus penggunaan. Spesifikasi masing-masing lingkungan pengujian dijabarkan pada Tabel 5.1.

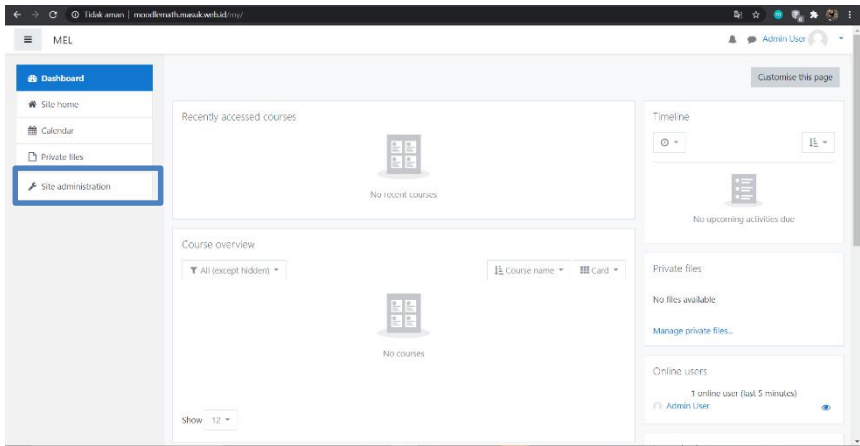
Tabel 5.1 Lingkungan Pengujian Fungsionalitas Perangkat Lunak

Perangkat	Spesifikasi
Perangkat Keras	Prosesor: Intel® Core™ i5-A442U CPU @ 2.00GHz Memori: 4096MB RAM
Perangkat Lunak	Sistem Operasi: Windows 10 Home 64-bit Perangkat Pengembang: Moodle 3.7, PHP 7.1++ Basis Data: MySQL Plugin: Course Recommender Version: 2017051504 Release: 1.0 (Build: 2020041800)

5.2 Penambahan Plugin Pemberi Rekomendasi Course Pada Moodle

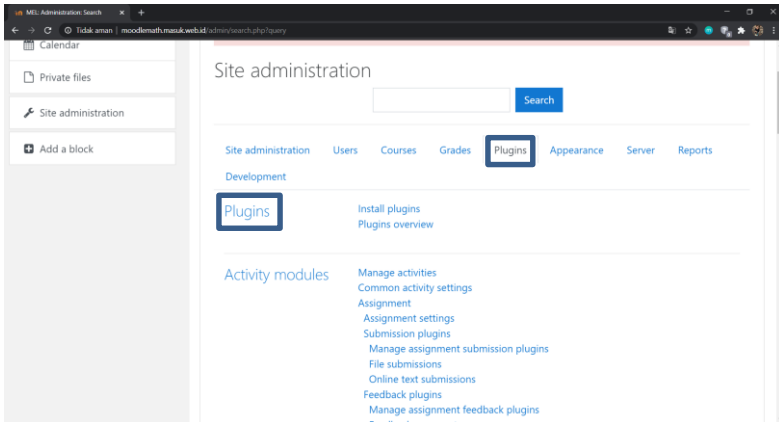
Penambahan *Plugin* pemberi rekomendasi konten belajar pada Moodle hanya dapat dilakukan oleh Admin. Berikut merupakan tahapan-tahapan untuk menambah *plugin* pada Moodle:

1. Admin memilih “Site Administration” pada halaman Dashboard Moodle yang ditunjukkan pada Gambar 5.1.



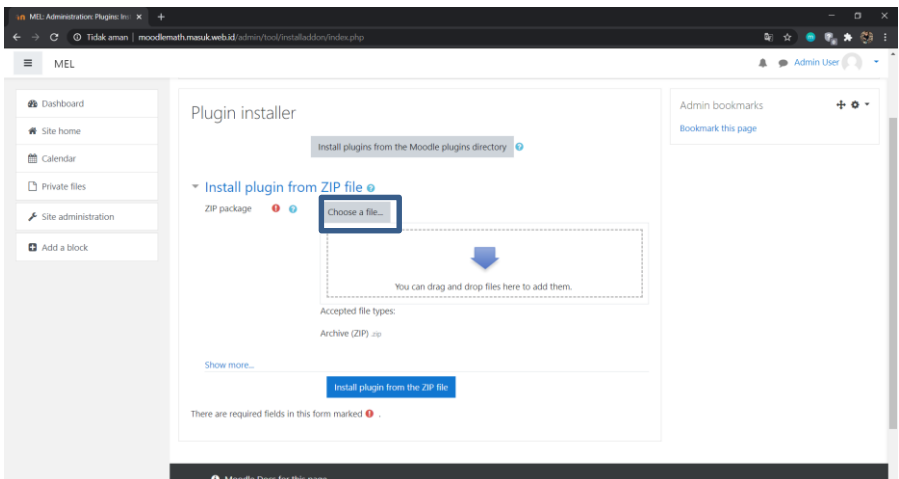
Gambar 5.1 Halaman Dashboard Moodle

2. Admin memilih “Plugins” kemudian memilih “Install Plugins” pada halaman Site Administration yang ditunjukkan pada Gambar 5.2.



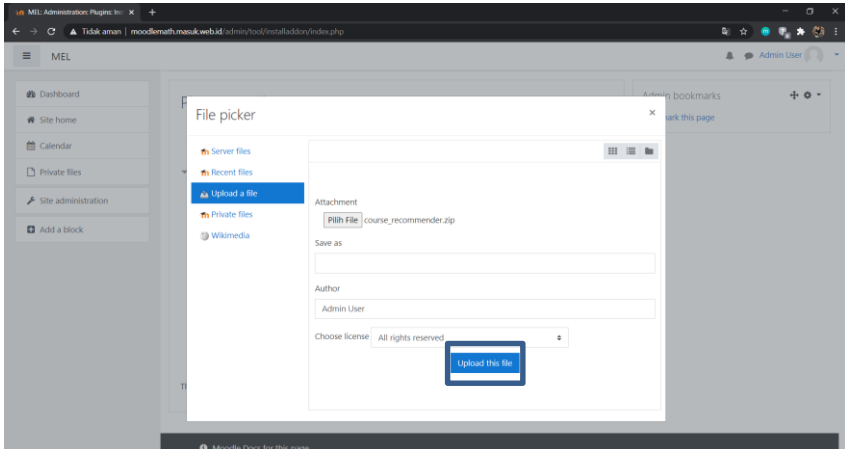
Gambar 5.2 Halaman *Site Administration*

- Admin memilih “Choose a file...” pada halaman Plugin Installer yang ditunjukkan pada Gambar 5.3.



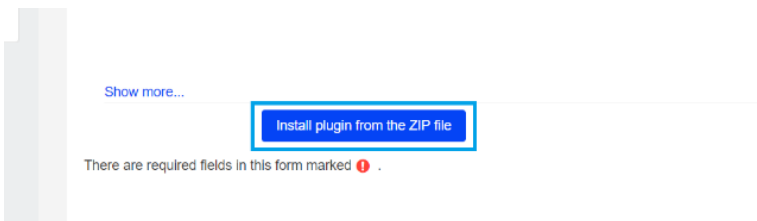
Gambar 5.3 Halaman *Plugin Installer*

- Admin memilih berkas *plugin* yang akan diinstal dengan format zip. Setelah itu admin memilih “Upload this file” yang ditunjukkan pada Gambar 5.4.



Gambar 5.4 Halaman *Upload File Plugin*

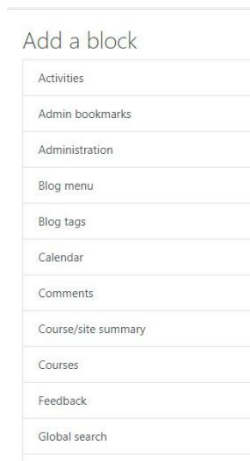
- Admin memilih “Install plugin from the ZIP file” yang ditunjukkan pada Gambar 5.5. *Plugin* berhasil diinstal pada Moodle.



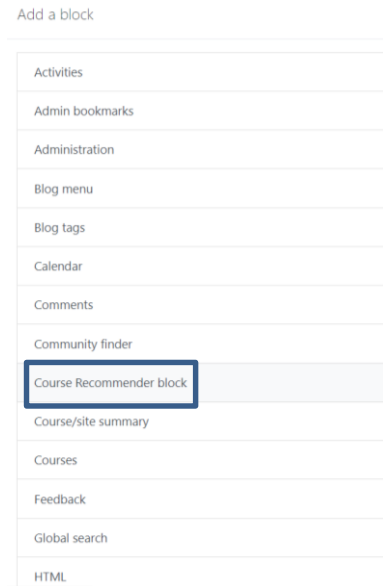
Gambar 5.5 Halaman *Plugin Installer*

5.3 Pengujian Moodle Tanpa Plugin dan Moodle dengan Plugin

Pengujian Moodle tanpa *plugin* dan Moodle dengan *plugin* ini bertujuan untuk menemukan perbedaan antara Moodle tanpa *plugin* dan Moodle dengan *plugin*. Berikut ini perbedaan dari Moodle tanpa *plugin* dan Moodle dengan *plugin course_recommender* yang ditunjukkan pada Gambar 5.6 dan Gambar 5.7.



Gambar 5.6 Moodle Tanpa Plugin Course Recommender



Gambar 5.7 Moodle dengan *Plugin Course Recommender*

Dari Gambar 5.6 dan Gambar 5.7 terdapat perbedaan pada sub menu, dimana pada Gambar 5.6 tidak terdapat fitur “Course Recommender”, sedangkan pada Gambar 5.7 terdapat fitur “Course Recommender” yang didalamnya terdapat halaman rekomendasi *course*.

Pada Tugas Akhir ini telah dilakukan pengujian terhadap responden dalam hal ini adalah wali murid yang mendampingi siswa dalam menggunakan Moodle dengan tambahan *plugin Course Recommender*. Bentuk formulir kuesioner ditunjukkan pada Tabel 8.1. Dari pertanyaan yang telah diajukan dapat disimpulkan pada Tabel 5.2.

Tabel 5.2 Hasil Pengujian kepada Wali Murid

No	Kesimpulan
1.	Wali murid siswa dapat dengan mudah melihat hasil rekomendasi <i>course</i> .
2.	Wali murid mendapatkan rekomendasi yang sesuai dengan kemiripan antar <i>course</i> , namun beberapa <i>course</i> kurang menghasilkan hasil yang maksimal.
3.	Plugin dapat digunakan sebagai bahan penunjang dalam membantu siswa belajar.

Berdasarkan kesimpulan hasil pengujian terhadap wali murid siswa, terdapat beberapa manfaat menggunakan Moodle dengan *plugin* sebagai berikut:

1. Wali murid terbantu dalam membantu siswa menentukan bab pelajaran untuk dipelajari.
 Dalam proses pembelajaran, siswa sering dibantu oleh orang tuanya sebagai walinya. Terkadang orang tua tidak mengetahui bab mana yang berkaitan dengan materi yang baru saja diselesaikan oleh siswa. Dengan adanya fitur rekomendasi, orang tua merasa dipermudah dalam memilih konten ketika mendampingi anak belajar.

2. Sebagai bahan penunjang agar siswa dapat belajar mandiri
Sebagai penunjang, sistem dapat memberikan kesempatan kepada siswa untuk belajar mandiri dengan memberikan rekomendasi konten belajar.

5.4 Pengujian Fungsionalitas

Pengujian akan dilakukan menggunakan data materi dari bab pelajaran matematika, yang dalam studi kasus ini akan digunakan sebagai *course* untuk pengujian, dengan menguji fungsionalitas dari aplikasi yang ada. Data diambil dari mata pelajaran matematika kelas 4, 5, dan 6 SD. Materi akan dihitung rekomendasinya berdasarkan kemiripan dari kontennya, tanpa memerhatikan tingkatan level.

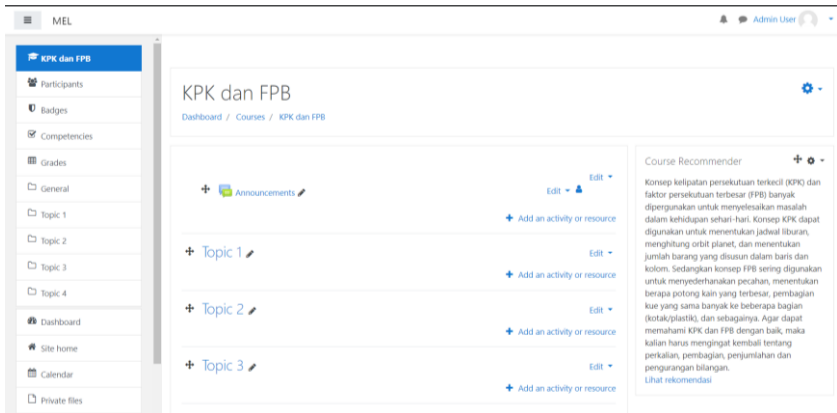
5.4.1 Pengujian Menambahkan Fitur Rekomendasi Course

Pengujian menambahkan fitur rekomendasi *course* merupakan pengujian terhadap kemampuan sistem (*plugin*) untuk menambahkan *plugin course recommender* dalam sebuah *course* yang sudah dipilih. Pengujian ini dilakukan ketika pengajar menekan tombol “Add a block” pada bagian menu *setting* dari *course* pada *e-learning*, kemudian pengajar memilih menu “Course Recommender Block”. Rincian skenario pengujian pada kasus penggunaan ini ditunjukkan pada Tabel 5.3. Hasil yang diujikan 4 sample *course* (bab pelajaran) yaitu Bilangan Bulat, KPK dan FPB, Pecahan, Pengumpulan dan Penyajian Data ditunjukkan pada Gambar 5.8, Gambar 5.9, Gambar 5.9, dan Gambar 5.9.

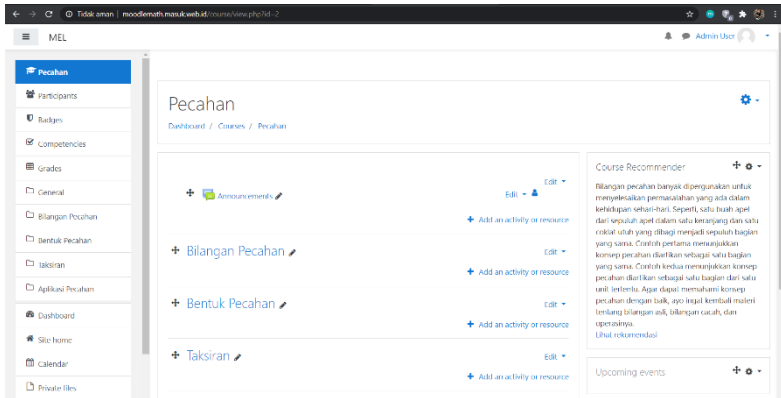
Tabel 5.3 Pengujian Menambahkan Fitur Rekomendasi Course

No. Pengujian	UC-001
Referensi Kasus Penggunaan	UC-001

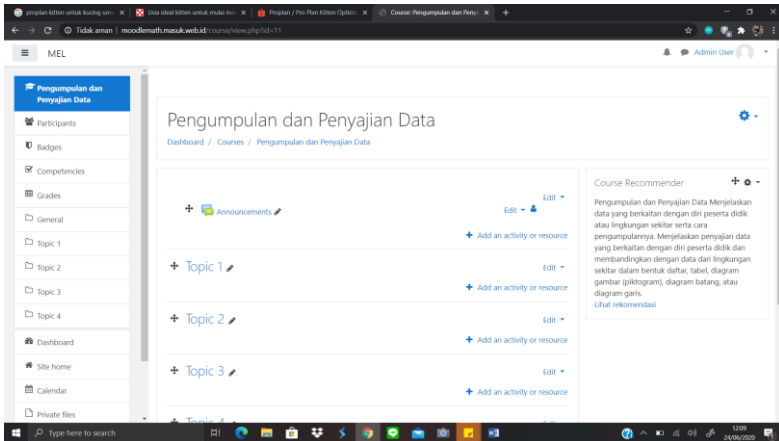
Nama	Pengujian menambahkan fitur rekomendasi <i>course</i>
Tujuan Pengujian	Menguji fungsionalitas untuk menambahkan fitur pemberian rekomendasi <i>course</i> pada mata pelajaran yang telah dibuat dalam Moodle
Kondisi Awal	Pengajar berada pada halaman <i>course</i> (mata pelajaran) yang dipilih.
Data Uji	-
Langkah Pengujian	<ol style="list-style-type: none"> 1. Pengajar memilih pengaturan <i>course</i>. 2. Pengajar menyalakan mode editing dengan memilih "Turn editing on" pada pengaturan <i>course</i> 3. Pada sisi kanan halaman <i>course</i>, pengajar memilih "Add a block" 4. Pengajar memilih "Course Recommender Block"
Hasil yang diharapkan	Sistem dapat menambahkan plugin course recommender block pada mata pelajaran yang telah dipilih
Hasil yang didapat	Plugin course recommender ditambahkan pada <i>course</i> .
Hasil Pengujian	Berhasil
Kondisi Akhir	Plugin course recommender berhasil ditambahkan.



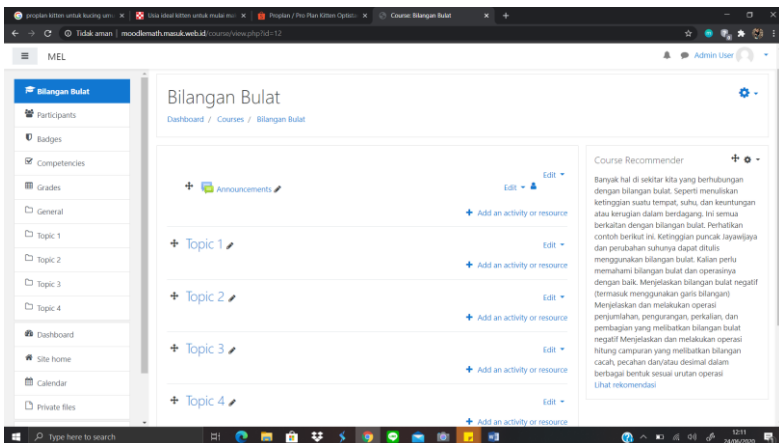
Gambar 5.8 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran KPK dan FPB



Gambar 5.9 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran Pecahan



Gambar 5.10 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran Pengumpulan dan Penyajian Data



Gambar 5.11 Hasil Pengujian Penambahan Fitur Rekomendasi Course pada Bab Pelajaran Bilangan Bulat

5.4.2 Pengujian Melihat Hasil Rekomendasi Course

Pengujian melihat hasil rekomendasi *course* merupakan pengujian terhadap kemampuan sistem (*plugin*) untuk menampilkan data hasil rekomendasi. Pengujian ini dilakukan ketika siswa memilih “lihat rekomendasi” pada plugin. Rincian skenario pengujian pada kasus penggunaan ini ditunjukkan pada

Tabel 5.4. Hasil yang diujikan pada 4 sample course (bab pelajaran) yaitu Bilangan Bulat, KPK dan FPB, Pecahan, Pengumpulan dan Penyajian Data ditunjukkan pada Gambar 5.12, Gambar 5.13, Gambar 5.12 dan Gambar 5.13.

Tabel 5.4 Pengujian Melihat Hasil Rekomendasi Course

No. Pengujian	UC-002
Referensi Kasus Penggunaan	UC-002
Nama	Pengujian melihat hasil rekomendasi <i>course</i>
Tujuan Pengujian	Menguji fungsionalitas untuk melihat hasil rekomendasi <i>course</i>
Kondisi Awal	Plugin <i>course recommender</i> telah terpasang
Data Uji	-
Langkah Pengujian	Siswa memilih “lihat rekomendasi” pada plugin yang telah terpasang
Hasil yang diharapkan	Sistem dapat menampilkan rekomendasi <i>course</i> yang memiliki kemiripan dengan <i>course</i> yang sedang diambil.
Hasil yang didapat	Data rekomendasi <i>course</i> yang memiliki kemiripan dengan <i>course</i> yang sedang diambil ditampilkan.
Hasil Pengujian	Berhasil
Kondisi Akhir	Data rekomendasi <i>course</i> berhasil ditampilkan

The screenshot shows a Moodle course page titled 'Rekomendasi Course' for the course 'KPK dan FPB'. The page displays a table with the following data:

Rekomendasi Course	Value
Pecahan	0.134025
Bilangan Bulat	0.0767703
Statistika	0.0482057

The left sidebar shows the course structure with 'KPK dan FPB' selected. The bottom status bar indicates the user is logged in as 'Admin User'.

Gambar 5.12 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran KPK dan FPB

The screenshot shows a Moodle course page titled 'Rekomendasi Course' for the course 'Pecahan'. The page displays a table with the following data:

Rekomendasi Course	Value
KPK dan FPB	0.134025
Bilangan Bulat	0.0965887
Aproksimasi	0.0231301

The left sidebar shows the course structure with 'Pecahan' selected. The bottom status bar indicates the user is logged in as 'Admin User'.

Gambar 5.13 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran Pecahan

The screenshot shows the Moodle interface for a course titled 'Rekomendasi Course' under the category 'Pengumpulan dan Penyajian Data'. The left sidebar contains navigation options like 'Participants', 'Badges', 'Competencies', 'Grades', and 'General'. The main content area displays a table with course recommendations and their values.

Rekomendasi Course	Value
Statistika	0.188673
Statistika 2	0.103622
Skala	0.0357472

At the bottom of the page, there is a notification: 'Moodle Docs for this page: You are logged in as Admin User (Log out) Pengumpulan dan Penyajian Data Data retention summary Barge all caches'.

Gambar 5.14 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran Pengumpulan dan Penyajian Data

The screenshot shows the Moodle interface for a course titled 'Rekomendasi Course' under the category 'Bilangan Bulat'. The left sidebar contains navigation options like 'Participants', 'Badges', 'Competencies', 'Grades', and 'General'. The main content area displays a table with course recommendations and their values.

Rekomendasi Course	Value
Pecahan	0.0965887
KPK dan FPB	0.0767703
Bangun Datar	0.0430364

At the bottom of the page, there is a notification: 'Moodle Docs for this page: You are logged in as Admin User (Log out) Bilangan Bulat Data retention summary Barge all caches'.

Gambar 5.15 Hasil Melihat Hasil Rekomendasi Course pada Bab Pelajaran Bilangan Bulat

5.5 Data Uji

Berdasarkan pengujian yang telah dilakukan, dihasilkan data uji yang berisi input beserta output yang keluar dari hasil pengujian. Data tersebut ditunjukkan pada Tabel 5.5.

Tabel 5.5 Data Uji

Materi	Rekomendasi	Nilai Cosine
Pecahan	KPK dan FPB	0,134025
	Bilangan Bulat	0,0965887
	Aproksimasi	0,0231301
KPK dan FPB	Pecahan	0,134025
	Bilangan Bulat	0,0767703
	Statistika	0,0482057
Aproksimasi	Statistika	0,0829588
	Bangun Ruang	0,0499827
	Pecahan	0,0231301
Bangun Datar	Bangun Ruang 2	0,0718691
	Bilangan Bulat	0,0430364
	Statistika	0,0417124
Statistika	Pengumpulan dan Penyajian Data	0,188673
	Aproksimasi	0,0829588
	Statistika 2	0,0678414
Skala	Statistika 2	0,0830987
	KPK dan FPB	0,0426424
	Pengumpulan dan Penyajian Data	0,0357472
Bangun Ruang	Bangun Ruang 2	0,249093
	Aproksimasi	0,0499827
	Bangun Datar	0,035777

Pengumpulan dan Penyajian Data	Statistika	0,188673
	Statistika 2	0,103622
	Skala	0,0357472
Bilangan Bulat	Pecahan	0,0965887
	KPK dan FPB	0,0767703
	Bangun Datar	0,0430364
Bangun Ruang 2	Bangun Ruang	0,249093
	Bangun Datar	0,0718691
	Aproksimasi	0,0210362
Statistika 2	Pengumpulan dan Penyajian Data	0,103622
	Skala	0,0830987
	Statistika	0,0678414

Berdasarkan hasil pengujian, sistem menampilkan 3 materi teratas yang memiliki nilai *cosine similarity* tertinggi. Sehingga hasil yang didapat sesuai dengan kemiripan deskripsi dari materi. Untuk pengujian lebih lanjut, dilakukan pengujian silang dengan pelajaran Sejarah, PKN, dan Bahasa Inggris. Hasil pengujian ditunjukkan pada Tabel 5.6.

Tabel 5.6 Hasil Rekomendasi Pengujian Silang

Materi	Rekomendasi	Nilai Cosine
Sejarah	Bilangan Bulat	0,0285031
	PKN	0,0257139
	Pecahan	0,0186188
PKN	Sejarah	0,0257139
	Pecahan	0,0161088
	Bahasa Inggris	0,0149266

Bahasa Inggris	Statistika	0,0358549
	KPK dan FPB	0,020089
	Pecahan	0,0180446

Berdasarkan hasil pengujian silang, nilai *cosine similarity* yang dihasilkan sangat kecil dibandingkan dengan data uji yang dilakukan dengan pengujian pertama. Hal ini dikarenakan korelasi antar materi yang sangat kecil sehingga menghasilkan nilai kemiripan yang kecil pula. Sehingga, meskipun ditambahkan materi yang tidak berhubungan dengan matematika, sistem tidak akan menampilkan materi tersebut sebagai rekomendasi, hal tersebut dikarenakan sistem hanya menampilkan 3 materi teratas dengan nilai *cosine similarity* tertinggi sebagai rekomendasi.

5.6 Evaluasi Pengujian

Berdasarkan pengujian yang telah dilakukan, semua pengujian fungsionalitas memberikan hasil yang sesuai dengan skenario yang direncanakan. Rangkuman mengenai hasil pengujian fungsionalitas yang diujicobakan pada 4 sample *course* (bab pelajaran) yaitu Bilangan Bulat, KPK dan FPB, Pecahan, Pengumpulan dan Penyajian Data yang diambil dari buku BSE pelajaran matematika untuk siswa kelas 4, 5 dan 6 ditunjukkan pada Tabel 5.7.

Tabel 5.7 Evaluasi Pengujian Fungsionalitas

No	Fungsionalitas	Terpenuhi
1.	Menambahkan Fitur Rekomendasi <i>Course</i>	✓
2.	Melihat Hasil Rekomendasi <i>Course</i>	✓

Berdasarkan data pada Tabel 5.7, seluruh skenario pengujian berhasil dilakukan. Sehingga dapat disimpulkan bahwa fungsionalitas dari sistem (*plugin*) bekerja sesuai dengan yang diharapkan.

[Halaman ini sengaja dikosongkan]

BAB VI KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan yang diperoleh selama pengerjaan Tugas Akhir dan saran mengenai pengembangan yang dapat dilakukan terhadap Tugas Akhir ini di masa yang akan datang.

6.1 Kesimpulan

Kesimpulan dari Tugas Akhir ini adalah tujuan dari *plugin Moodle* pemberi rekomendasi konten belajar untuk mendukung pembelajaran mandiri telah berhasil dibangun dengan solusi dari beberapa rumusan masalah sebagai berikut:

1. Implementasi modul pengukuran kedekatan antar dokumen pada *plugin Course Recommender Block* berhasil dilakukan. Implementasi ini melakukan perhitungan frekuensi kemunculan kata, bobot TF-IDF, panjang vector, dan cosine similarity. Perhitungan dimasukkan ke dalam arsitektur plugin menggunakan bahasa pemrograman PHP.
2. Pemberian rekomendasi konten belajar dapat menunjang pembelajaran siswa secara daring dan memudahkan siswa dalam memilih konten belajar.
3. *Plugin* rekomendasi *course* berhasil ditambahkan pada Moodle versi 3.7+. Penambahan ini dilakukan oleh Administator dengan menginstalasi *plugin* tersebut.
4. Berdasarkan pengujian silang yang dilakukan, plugin tidak menampilkan *course* yang tidak memiliki korelasi dengan *course* yang diambil. Hal tersebut dikarenakan nilai *cosine similarity* yang dihasilkan terlalu kecil dan sistem hanya menampilkan 3 nilai *cosine similarity* tertinggi sebagai rekomendasi.
5. Hasil evaluasi *precision*, *recall*, dan *accuracy* berdasarkan data uji ditunjukkan pada table 6.1.

Tabel 6.1 Hasil Evaluasi Sistem pada data uji

No	Evaluasi	Nilai
1.	<i>precision</i>	0,8
2.	<i>recall</i>	0,888889
3.	<i>accuracy</i>	0,727273

6. Kelemahan dari sistem terdapat pada performa dan skalabilitas. Hal tersebut dikarenakan sistem melakukan perhitungan ulang setiap kali plugin dijalankan. Sehingga, untuk data yang sangat besar, sistem akan membutuhkan waktu yang cukup lama untuk melakukan proses perhitungan.

6.2 Saran

Berikut adalah saran untuk pengembangan *Plugin Course Recommender* di masa yang akan datang. Saran ini didasarkan pada hasil perancangan, implementasi dan pengujian yang telah dilakukan.

1. Melibatkan pakar dan pengajar.
2. Memperluas batasan kategori *course*, tidak hanya pada topik matematika.
3. Memperluas batasan konten pada plugin, bukan hanya menghitung dari kemiripan konten *course* saja, tetapi juga memperhatikan rekomendasi tingkat level (*learning path*).
4. Meningkatkan performa dan skalabilitas dari plugin.

DAFTAR PUSTAKA

- [1] N. Prasetyo, "Recommendation System Dengan Python : Definisi (Part 1)," 11 July 2019. [Online]. Available: <https://medium.com/data-folks-indonesia/recommendation-system-dengan-python-definisi-part-1-71154dc3f700>. [Diakses 29 May 2020].
- [2] E. Grimaldi, "How to build a content-based movie recommender system with Natural Language Processing," 2 October 2018. [Online]. Available: <https://towardsdatascience.com/how-to-build-from-scratch-a-content-based-movie-recommender-with-natural-language-processing-25ad400eb243>. [Diakses 15 March 2020].
- [3] M. Riadi, "Pengertian, Karakteristik dan Manfaat E-Learning," 17 June 2014. [Online]. Available: <https://www.kajianpustaka.com/2014/06/pengertian-karakteristik-dan-manfaat-elearning.html>. [Diakses 19 November 2019].
- [4] "Wikipedia," [Online]. Available: <https://id.wikipedia.org/wiki/Moodle>. [Diakses 20 November 2019].
- [5] P. S. Adi, "Sistem rekomendasi nilai mata kuliah menggunakan metode content-based filtering," 2010.
- [6] W. R. T., "Penerapan Algoritma Cosine Similarity dan pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi," *Jurnal Teknik Elektro Vol. 9 No. I*, vol. II, p. 20, 2019.
- [7] "IdCloudHost," [Online]. Available: <https://idcloudhost.com/kamus-hosting/php/>. [Diakses 20 05 2020].

[Halaman ini sengaja dikosongkan]

LAMPIRAN

Tabel 8.1 Kuesioner Pengujian Terhadap Pengajar

No	Pertanyaan	Setuju	Tidak Setuju
1.	Kemudahan melihat rekomendasi konten belajar berdasarkan kemiripan deskripsi <i>course</i>		
2.	Data rekomendasi yang ditampilkan oleh sistem sesuai dengan kebutuhan siswa		
5.	Sistem dapat digunakan sebagai bahan penunjang pembelajaran		

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Penulis lahir di Denpasar, 16 Juni 1998. Penulis telah menempuh pendidikan dasar di SDI Al-Falah Kediri, kemudian untuk pendidikan menengah pertama di MTsN Kediri 2 dan di jenjang menengah atas di SMAN 2 Kediri. Lalu penulis melanjutkan pendidikan sarjana S1 di Jurusan Informatika, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya. Lalu penulis melanjutkan pendidikan sarjana S1 di Jurusan Informatika, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya.

Selama kuliah, penulis aktif berorganisasi menjadi Sekretaris Schematics 2016 dan 2017, staf Himpunan Mahasiswa Teknik Computer-Informatika (HMTC) ITS 2017/2018 sebagai Staff Departemen Pengembangan Profesi, HMTC ITS 2018/2019 sebagai Staff Ahli Departemen Pengembangan Profesi.

Penulis dalam menyelesaikan pendidikan S1 mengambil rumpun mata kuliah (RMK) Rekayasa Perangkat Lunak serta memiliki ketertarikan di bidang Sistem dan Manajemen Basis Data, Pemrograman Web, serta pemrograman perangkat bergerak berbasis Android. Untuk komunikasi, penulis dapat dihubungi melalui surel: finayunita98@gmail.com.

