

**TUGAS AKHIR –** **IF184802**

**PEMBUATAN PLUGIN MOODLE PENGUKURAN ASPEK KOOPERATIF UNTUK MENDUKUNG PEMBELAJARAN BERMAKNA**

**FIRDA RHEINALIA NUGROHO**

**NRP 05111540000011**

Dosen Pembimbing

Dr. Siti Rochimah

Daniel Oranova S., S.Kom., M.Sc., P.D.Eng.

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2019



***(Halaman ini sengaja dikosongkan)***

**TUGAS AKHIR – IF184802**

**PEMBUATAN PLUGIN MOODLE PENGUKURAN ASPEK KOOPERATIF UNTUK MENDUKUNG PEMBELAJARAN BERMAKNA**

**FIRDA RHEINALIA NUGROHO**

**NRP 05111540000011**

**Dosen Pembimbing**

**Dr. Siti Rochimah**

**Daniel Oranova S., S.Kom., M.Sc., P.D.Eng.**

**DEPARTEMEN INFORMATIKA**

**Fakultas Teknologi Informasi dan Komunikasi**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2019**

****

**UNDERGRADUATE THESIS – IF184802**

**DEVELOPMENT OF COOPERATIVE ASPECT MOODLE PLUGIN MEASUREMENT TO SUPPORT MEANINGFUL LEARNING**

**FIRDA RHEINALIA NUGROHO**

**NRP 05111540000011**

**Supervisors**

**Dr. Siti Rochimah**

**Daniel Oranova S., S.Kom., M.Sc., P.D.Eng.**

**DEPARTMENT OF INFORMATICS**

**Faculty of Information Technology and Communication**

**Sepuluh Nopember Institute of Technology**

**Surabaya 2019**

***(Halaman ini sengaja dikosongkan)***

**LEMBAR PENGESAHAN**

**PEMBUATAN PLUGIN MOODLE PENGUKURAN**

**ASPEK KOOPERATIF UNTUK MENDUKUNG PEMBELAJARAN BERMAKNA**

**TUGAS AKHIR**

Diajukan Untuk Memenuhi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

pada

Bidang Studi Rekayasa Perangkat Lunak

Program Studi S-1 Departemen Informatika

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Oleh

**FIRDA RHEINALIA NUGROHO**

**NRP: 05111540000011**

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Dr. Ir. Siti Rochimah, M.T. ..........................

(NIP 196810021994032001) (Pembimbing 1)

Daniel Oranova S, S.Kom., M.Sc.,PD.Eng ....................

(NIP 197411232006041001) (Pembimbing 2)

**SURABAYA**

**JANUARI, 2019**

***(Halaman ini sengaja dikosongkan)***

**PEMBUATAN PLUGIN MOODLE PENGUKURAN**

**ASPEK KOOPERATIF UNTUK MENDUKUNG PEMBELAJARAN BERMAKNA**

**Nama Mahasiswa : Firda Rheinalia Nugroho**

**NRP : 05111540000011**

**Departemen : Informatika FTIK-ITS**

**Dosen Pembimbing 1 : Dr. Siti Rochimah**

**Dosen Pembimbing 2 : Daniel Oranova S., S.Kom., M.Sc., P.D.Eng.**

# Abstrak

*Sistem e-learning telah digunakan secara luas di dunia, termasuk di Indonesia. Pengukuran efektifitas e-learning dapat dilakukan melalui observasi keterlibatan pelajar dalam e-learning. Di lain pihak, tujuan utama pendidikan di setiap level adalah harus melibatkan pelajar di dalam meaningful learning. Keterlibatan pelajar di dalam meaningful learning terjadi ketika pelajar dapat menciptakan makna. Meaningful learning merupakan suatu proses mengkaitkan informasi baru pada konsep-konsep relevan yang terdapat dalam struktur kognitif seseorang. Struktur kognitif ialah fakta-fakta, konsep-konsep, dan generalisasi-generalisasi yang telah dipelajari dan diingat pelajar. Meaningful learning terjadi ketika pelajar itu aktif, konstruktif, intensional, kooperatif dan bekerja pada tugas autentik.*

*Oleh karena itu, penambahan plugin untuk penilaian pembelajaran bermakna pada e-learning diperlukan. Dalam tugas akhir ini akan berfokus pada pembuatan plugin untuk aspek kooperatif. Plugin tersebut akan dibangun menggunakan Moodle yang merupakan salah satu aplikasi CMS (Course Management System) berbasis web. Plugin ini dibuat untuk membantu para pengajar dalam melakukan penilaian pada tiap pelajar berdasarkan aktifitas pelajar pada e-learning sehingga penilaian menjadi lebih objektif.*

*Tujuan dari pengerjaan Tugas Akhir ini adalah membuat plugin Moodle untuk mengukur aspek kooperatif yang mendukung pembelajaran bermakna dan mengukur keterlibatan pelajar dengan pengukuran aspek kooperatif yang mendukung pembelajaran bermakna dalam e-learning.*

**Kata kunci: e-learning, plugin, moodle, pembelajaran bermakna, kooperatif.**

**DEVELOPMENT OF COOPERATIVE ASPECT**

**MOODLE PLUGIN MEASUREMENT TO SUPPORT MEANINGFUL LEARNING**

**Student’s Name : Firda Rheinalia Nugroho**

**Student’s ID : 05111540000011**

**Department : Informatics FTIK-ITS**

**First Advisor :** **Dr. Siti Rochimah**

**Second Advisor :** **Daniel Oranova S., S.Kom., M.Sc., P.D.Eng.**

# Abstract

*The e-learning system has been used widely in the world, including in Indonesia. Measurement of e-learning effectiveness can be done through observation of student involvement in e-learning. On the other hand, the main purpose of education at each level is to involve students in meaningful learning. Student involvement in meaningful learning occurs when students can create meaning. Meaningful learning is a process of linking new information to relevant concepts contained in one's cognitive structure. Cognitive structures are facts, concepts, and generalizations that students have learned and remembered. Meaningful learning occurs when students are active, constructive, intentional, cooperative and work on authentic tasks.*

*Therefore, the addition of plugins for the assessment of meaningful learning on e-learning is needed. In this final project will focus on making plugins for cooperative aspects. The plugin will be built using Moodle which is one of the web-based CMS (Course Management System) applications. This plugin is created to help instructors in assessing each student based on student activities in e-learning so that the assessment becomes more objective.*

*The purpose of this Final Project is to create a Moodle plugin to measure cooperative aspects that support meaningful learning and measure student involvement with cooperative aspects of measurement that support meaningful learning in e-learning.*

**Keyword: e-learning, plugin, moodle, meaningful learning, cooperative**

# KATA PENGANTAR

Alhamdulillahirobbil ‘alamiin, Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul :

***“*PEMBUATAN PLUGIN MOODLE PENGUKURAN ASPEK KOOPERATIF UNTUK MENDUKUNG PEMBELAJARAN BERMAKNA”**

Harapan dari penulis semoga apa yang tertulis di dalam buku Tugas Akhir ini dapat bermanfaat bagi pengembangan ilmu pengetahuan saat ini, serta dapat memberikan kontribusi yang nyata.

Dalam pelaksanaan dan pembuatan Tugas Akhir ini tentunya sangat banyak bantuan yang penulis terima dari berbagai pihak, tanpa mengurangi rasa hormat penulis ingin mengucapkan terima kasih sebesar-besarnya kepada:

1. Allah SWT atas segala karunia dan rahmat-Nya yang telah diberikan sampai saat ini.
2. Ayah penulis Bapak Nugroho, Ibu penulis ibu Luluk Rosida, Kakak Fachreza Reynaldi Nugroho, Adik Frania Rheivalia Nugroho, dan keluarga yang selalu memberikan dukungan penuh untuk menyelesaikan Tugas Akhir ini.
3. Ibu Dr. Siti Rochimah dan Bapak Daniel Oranova S., S.Kom., M.Sc., P.D.Eng. selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk memberikan petunjuk selama proses pengerjaan tugas akhir ini.
4. Bapak, Ibu dosen Departemen Informatika ITS yang telah banyak memberikan ilmu dan bimbingan yang tak ternilai harganya bagi penulis.
5. Seluruh staf dan karyawan FTIK ITS yang banyak memberikan kelancaran administrasi akademik kepada penulis.
6. Astrid, Lia, Mutia, Tata yang telah bersama-sama berjuang menyelesaikan Tugas Akhir ini.
7. Teman-teman Administrator Laboratorium Rekayasa Perangkat Lunak (RPL) yaitu Irsa, Astrid, Zahra, Nila, Alvin, Djohan, Adit, Mail, Hafid, Andika, dan Adis yang telah membantu dan memberi dukungan selama pengerjaan Tugas Akhir Ini.
8. Faiq, Azizah, Nindy, Rozana, Syavira serta seluruh teman-teman angkatan 2015 Departemen Informatika ITS yang telah menemani perjuangan selama kurang lebih 3,5 tahun ini serta memberikan saran, masukan, dan dukungan terhadap pengerjaan Tugas Akhir ini.
9. Teman-teman Kabinet Semangat Berpadu BEM FTIK 2017/2018 yang telah memberikan kesempatan untuk mendapatkan banyak pengalaman selama masa perkuliahan.
10. Serta pihak-pihak lain yang tidak dapat disebutkan disini yang telah banyak membantu penulis dalam penyusunan Tugas Akhir ini.

Penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Januari 2019

Firda Rheinalia Nugroho

# DAFTAR ISI

[LEMBAR PENGESAHAN vi](#_Toc534976228)

[Abstrak viii](#_Toc534976229)

[Abstract x](#_Toc534976230)

[KATA PENGANTAR xii](#_Toc534976231)

[DAFTAR ISI xiv](#_Toc534976232)

[DAFTAR GAMBAR xviii](#_Toc534976233)

[DAFTAR TABEL xx](#_Toc534976234)

[DAFTAR KODE SUMBER xxii](#_Toc534976235)

[BAB I PENDAHULUAN 1](#_Toc534976236)

[1.1 Latar Belakang 1](#_Toc534976237)

[1.2 Rumusan Masalah 2](#_Toc534976238)

[1.3 Batasan Masalah 2](#_Toc534976239)

[1.4 Tujuan 3](#_Toc534976240)

[1.5 Manfaat 3](#_Toc534976241)

[1.6 Metodologi 3](#_Toc534976242)

[1.6.1 Penyusunan Proposal Tugas Akhir 3](#_Toc534976243)

[1.6.2 Studi Literatur 4](#_Toc534976244)

[1.6.3 Analisis dan Desain Perangkat Lunak 4](#_Toc534976245)

[1.6.4 Implementasi Perangkat Lunak 4](#_Toc534976246)

[1.6.5 Pengujian dan Evaluasi 4](#_Toc534976247)

[1.6.6 Penyusunan Buku 5](#_Toc534976248)

[1.7 Sistematika Penulisan Laporan 5](#_Toc534976249)

[BAB II DASAR TEORI 7](#_Toc534976250)

[2.1 Pembelajaran Bermakna 7](#_Toc534976251)

[2.2 Aspek Kooperatif 8](#_Toc534976252)

[2.3 *E-learning* 12](#_Toc534976253)

[2.3.1 Filosofi Moodle 12](#_Toc534976254)

[2.3.2 Aktivitas Moodle 14](#_Toc534976255)

[2.3.3 Arsitektur Moodle 15](#_Toc534976256)

[2.3.4 *Plugin* Moodle 16](#_Toc534976257)

[2.3.5 *Plugin* *Report* 17](#_Toc534976258)

[2.3.6 *Scheduled Task* 17](#_Toc534976259)

[2.4 Keserupaan Semantik 18](#_Toc534976260)

[2.5 NLP Tools 19](#_Toc534976261)

[2.6 Yandex Translator 19](#_Toc534976262)

[BAB III ANALISIS DAN PERANCANGAN SISTEM 21](#_Toc534976263)

[3.1 Analisis Sistem 21](#_Toc534976264)

[3.1.1 Analisis Permasalahan 21](#_Toc534976265)

[3.1.2 Deskripsi Umum Sistem 22](#_Toc534976266)

[3.1.2.1 Perhitungan Data Log Tindakan Pelajar 23](#_Toc534976267)

[3.1.2.2 Data Konten 24](#_Toc534976268)

[3.1.2.3 Data Konteks 25](#_Toc534976269)

[3.1.2.4 Perhitungan Frekuensi Tindakan Pelajar pada Aktivitas Moodle 31](#_Toc534976270)

[3.1.2.5 Perhitungan Bobot Tindakan Aktivitas Moodle........... 32](#_Toc534976271)

[3.1.2.6 Perhitungan Nilai Aktivitas Pelajar pada Moodle..... 34](#_Toc534976272)

[3.1.2.7 Perhitungan Aspek Kooperatif 34](#_Toc534976273)

[3.1.2.8 Penentuan Tingkat Keterlibatan Pelajar pada Aspek Kooperatif 35](#_Toc534976274)

[3.1.3 Spesifikasi Kebutuhan Perangkat Lunak 36](#_Toc534976275)

[3.1.3.1 Kebutuhan Fungsional 37](#_Toc534976276)

[3.1.3.2 Kebutuhan Non-Fungsional 37](#_Toc534976277)

[3.1.3.3 Aktor 38](#_Toc534976278)

[3.1.3.4 Kasus Penggunaan 38](#_Toc534976279)

[3.1.3.4.1 Memonitor Tingkat *Meaningful Learning* (UC-001).... 39](#_Toc534976280)

[3.1.3.4.2 Melihat Detail Nilai Aspek (UC-002) 41](#_Toc534976293)

[3.1.3.4.3 Melihat Detail Nilai setiap Pelajar (UC-003)........... 43](#_Toc534976307)

[3.2 Perancangan 45](#_Toc534976308)

[3.2.1 Perancangan Arsitektur Sistem 46](#_Toc534976309)

[3.2.2 Perancangan Diagram Kelas 48](#_Toc534976310)

[3.2.3 Perancangan Basis Data 48](#_Toc534976311)

[3.2.3.1 Tabel mdlnr\_course 48](#_Toc534976312)

[3.2.3.2 Tabel mdlnr\_user 53](#_Toc534976313)

[3.2.3.3 Tabel mdlnr\_logstore\_standard\_log 53](#_Toc534976314)

[3.2.3.4 Tabel mdlnr\_aktivitas 53](#_Toc534976315)

[3.2.3.5 Tabel mdlnr\_aksi 54](#_Toc534976316)

[3.2.3.6 Tabel mdlnr\_aktivitas\_aksi 54](#_Toc534976317)

[3.2.3.7 Tabel mdlnr\_aspek 55](#_Toc534976318)

[3.2.3.8 Tabel mdlnr\_bobot 55](#_Toc534976319)

[3.2.3.9 Tabel mdlnr\_frekuensi 56](#_Toc534976320)

[3.2.3.10 Tabel mdlnr\_log\_aksi 57](#_Toc534976321)

[3.2.3.11 Tabel mdlnr\_nilai\_akhir 57](#_Toc534976322)

[3.2.3.12 Tabel mdlnr\_keyword\_kognitif 58](#_Toc534976323)

[3.2.4 Perancangan Antarmuka Pengguna 53](#_Toc534976324)

[3.2.4.1 Rancangan Halaman Antarmuka Memonitor Tingkat Meaningful Learning Pelajar 59](#_Toc534976325)

[3.2.4.2 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Kooperatif 61](#_Toc534976326)

[3.2.4.3 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar 62](#_Toc534976327)

[BAB IV IMPLEMENTASI 65](#_Toc534976328)

[4.1 Lingkungan Implementasi 65](#_Toc534976333)

[4.2 Implementasi Tampilan Antarmuka 65](#_Toc534976334)

[4.2.1 Implementasi Halaman Monitor Tingkat *Meaningful Learning* 65](#_Toc534976335)

[4.2.2 Implementasi Halaman Melihat Detail Nilai Aspek Kooperatif 66](#_Toc534976336)

[4.2.3 Implementasi Halaman Melihat Detail Nilai Setiap Pelajar 67](#_Toc534976337)

[4.3 Implementasi Kasus Penggunaan 68](#_Toc534976338)

[4.3.1 Implementasi Kasus Penggunaan Monitor Tingkat *Meaningful Learning* 69](#_Toc534976339)

[4.3.2 Implementasi Kasus Penggunaan Melihat Detail Nilai Aspek Kooperatif 72](#_Toc534976340)

[4.3.3 Implementasi Kasus Penggunaan Melihat Detail Nilai setiap Pelajar 76](#_Toc534976341)

[4.4 Implementasi Perhitungan 82](#_Toc534976342)

[4.4.1 Implementasi Perhitungan Frekuensi Tindakan Pelajar pada Aktivitas Moodle 83](#_Toc534976343)

[4.4.1.1 Implementasi Perhitungan Frekuensi pada Data Konten...... 83](#_Toc534976344)

[4.4.1.2 Implementasi Perhitungan Frekuensi pada Data Konteks... 104](#_Toc534976345)

[4.4.2 Implementasi Perhitungan Bobot Tindakan Aktivitas Moodle 119](#_Toc534976346)

[4.4.3 Implementasi Perhitungan Aktivitas Pelajar pada Moodle 121](#_Toc534976347)

[4.4.4 Implementasi Perhitungan Aspek Kooperatif 122](#_Toc534976348)

[4.4.5 Implementasi Perhitungan Tingkat Keterlibatan Pelajar 124](#_Toc534976349)

[BAB V PENGUJIAN DAN EVALUASI 127](#_Toc534976350)

[5.1 Lingkungan Pengujian 127](#_Toc534976351)

[5.2 Pengujian Fungsionalitas 127](#_Toc534976352)

[5.2.1 Pengujian Monitor Tingkat Meaningful Learning 127](#_Toc534976353)

[5.2.2 Pengujian Melihat Detail Nilai Aspek 129](#_Toc534976354)

[5.2.3 Pengujian Melihat Detail Nilai setiap Pelajar 130](#_Toc534976355)

[5.3 Evaluasi Pengujian Fungsionalitas 134](#_Toc534976356)

[BAB VI KESIMPULAN DAN SARAN 137](#_Toc534976357)

[6.1 Kesimpulan 137](#_Toc534976359)

[6.2 Saran 138](#_Toc534976360)

[DAFTAR PUSTAKA 139](#_Toc534976361)

[BIODATA PENULIS 141](#_Toc534976362)

# DAFTAR GAMBAR

[Gambar 2.1 Aspek *Meaningful Learning* [7] 7](#_Toc534976518)

[Gambar 2.2 Direktori Minimal Pembuatan Plugin dengan Tipe Report 17](#_Toc534976519)

[Gambar 3.1 Deskripsi Umum Sistem 22](#_Toc534976520)

[Gambar 3.2 Rancangan Model Pengukuran Aspek Kooperatif Pelajar dalam E-learning 23](#_Toc534976521)

[Gambar 3.3 Pendeteksian Relevansi Pesan 28](#_Toc534976522)

[Gambar 3.4 Diagram Kasus Penggunaan 39](#_Toc534976523)

[Gambar 3.5 Diagram Aktivitas UC-001 41](#_Toc534976524)

[Gambar 3.6 Diagram Sekuens UC-001 41](#_Toc534976525)

[Gambar 3.7 Diagram Aktivitas UC-002 43](#_Toc534976526)

[Gambar 3.8 Diagram Sekuens UC-002 43](#_Toc534976527)

[Gambar 3.9 Diagram Aktivitas UC-003 45](#_Toc534976528)

[Gambar 3.10 Diagram Sekuens UC-003 45](#_Toc534976529)

[Gambar 3.11 Arsitektur *Plugin* Report Moodle 46](#_Toc534976530)

[Gambar 3.12 Class Diagram 49](#_Toc534976531)

[Gambar 3.13 Conceptual Data Model 50](#_Toc534976532)

[Gambar 3.14 Physical Data Model 51](#_Toc534976533)

[Gambar 3.15 Rancangan Halaman Antarmuka Memonitor Tingkat Meaningful Learning Pelajar 60](#_Toc534976534)

[Gambar 3.16 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Kooperatif 61](#_Toc534976535)

[Gambar 3.17 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar 63](#_Toc534976536)

[Gambar 3.18 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar 63](#_Toc534976537)

[Gambar 3.19 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar 64](#_Toc534976538)

[Gambar 4.1 Implementasi Monitor Tingkat Meaningful Learning 66](#_Toc534976539)

[Gambar 4.2 Implementasi Halaman Melihat Detail Nilai Aspek Kooperatif 66](#_Toc534976540)

[Gambar 4.3 Implementasi Halaman Detail Nilai setiap Pelajar 67](#_Toc534976541)

[Gambar 4.4 Implementasi Halaman Detail Nilai setiap Pelajar 68](#_Toc534976542)

[Gambar 4.5 Implementasi Halaman Detail Nilai setiap Pelajar 68](#_Toc534976543)

[Gambar 5.1 Pengujian Halaman Monitoring Tingkat *Meaningful Learning* Kelas PPL D 129](#_Toc534976544)

[Gambar 5.2 Pengujian Halaman Monitoring Tingkat Meaningful Learning Kelas PPL E 129](#_Toc534976545)

[Gambar 5.3 Pengujian Halaman Detail Nilai Aspek Kooperatif Kelas PPL D 130](#_Toc534976546)

[Gambar 5.4 Pengujian Halaman Detail Nilai Aspek Kooperatif Kelas PPL E 130](#_Toc534976547)

[Gambar 5.5 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL D 132](#_Toc534976548)

[Gambar 5.6 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL D 132](#_Toc534976549)

[Gambar 5.7 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL D 133](#_Toc534976550)

[Gambar 5.8 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL E 133](#_Toc534976551)

[Gambar 5.9 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL E 134](#_Toc534976552)

[Gambar 5.10 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL E 134](#_Toc534976553)

# 

# DAFTAR TABEL

[Tabel 2.1. Hasil Perhitungan Keserupaan Semantik antara Aspek Kooperatif Meaningful Learning dan Tindakan Aktivitas Moodle dari pelajar [5] 9](#_Toc534976554)

[Tabel 2.2. Daftar Bobot Tindakan dalam Perhitungan Nilai Aspek Kooperatif [5] 10](#_Toc534976555)

[Tabel 2.3. Contoh Nilai f(t) dan Nilai Aspek Kooperatif untuk Pelajar [5] 11](#_Toc534976556)

[Tabel 2.4. Tipe Plugin pada Moodle 16](#_Toc534976557)

[Tabel 3.1 Contoh Data Mentah Log Tindakan Pelajar dalam *E-Learning* 24](#_Toc534976558)

[Tabel 3.2 Data Konten yang Terlibat 25](#_Toc534976559)

[Tabel 3.3 Data Konteks yang Terlibat 26](#_Toc534976560)

[Tabel 3.4 Aktivitas Moodle pada *Plugin Report* 26](#_Toc534976561)

[Tabel 3.5 Tindakan setiap Aktivitas Moodle pada *Plugin Report* 26](#_Toc534976562)

[Tabel 3.6 Relasi Aktivitas dan Tindakan Aktivitas Moodle pada *Plugin Report* 27](#_Toc534976563)

[Tabel 3.7. Frekuensi Tindakan Pelajar terhadap Aktivitas Moodle pada *Plugin Report* 32](#_Toc534976564)

[Tabel 3.8 Ilustrasi Aktivitas yang digunakan pada Moodle 32](#_Toc534976565)

[Tabel 3.9 Ilustrasi Aktivitas yang tidak digunakan pada Moodle 33](#_Toc534976566)

[Tabel 3.10 Bobot setiap Tindakan Aktivitas Moodle pada *Plugin Report* 33](#_Toc534976567)

[Tabel 3.11 Nilai Aktivitas Pelajar pada Moodle pada Plugin Report 34](#_Toc534976568)

[Tabel 3.12 Nilai Aspek Kooperatif Pelajar pada Moodle 35](#_Toc534976569)

[Tabel 3.13 Kebutuhan Fungsional 37](#_Toc534976570)

[Tabel 3.14 Kualitas Perangkat Lunak 38](#_Toc534976571)

[Tabel 3.15 Kasus Penggunaan 39](#_Toc534976572)

[Tabel 3.16 Spesifikasi Kasus Penggunaan UC-001 39](#_Toc534976573)

[Tabel 3.17 Spesifikasi Kasus Penggunaan UC-002 42](#_Toc534976574)

[Tabel 3.18 Spesifikasi Kasus Penggunaan UC-0003 44](#_Toc534976575)

[Tabel 3.19 Tabel mdlnr\_aktivitas 54](#_Toc534976576)

[Tabel 3.20 Tabel mdlnr\_aksi 54](#_Toc534976577)

[Tabel 3.21 Tabel mdlnr\_aktivitas\_aksi 55](#_Toc534976578)

[Tabel 3.22 Tabel mdlnr\_aspek 55](#_Toc534976579)

[Tabel 3.23 Tabel mdlnr\_bobot 56](#_Toc534976580)

[Tabel 3.24 Tabel mdlnr\_frekuensi 56](#_Toc534976581)

[Tabel 3.25 Tabel mdlnr\_log\_aksi 57](#_Toc534976582)

[Tabel 3.26 Tabel mdlnr\_nilai\_akhir 58](#_Toc534976583)

[Tabel 3.27 Tabel mdlnr\_keyword\_kognitif 58](#_Toc534976584)

[Tabel 3.28 Atribut Antarmuka Memonitor Tingkat Meaningful Learning Pelajar 60](#_Toc534976585)

[Tabel 3.29 Atribut Antarmuka Melihat Detail Nilai Aspek Kooperatif 61](#_Toc534976586)

[Tabel 3.30 Atribut Antarmuka Aktivitas Mahasiswa 62](#_Toc534976587)

[Tabel 4.2 Penjelasan Kode Sumber 4.1 71](#_Toc534976588)

[Tabel 4.3 Penjelasan Kode Sumber 4.2 75](#_Toc534976589)

[Tabel 4.4 Penjelasan Kode Sumber 4.3 81](#_Toc534976590)

[Tabel 4.5 Penjelasan Kode Sumber 4.4 103](#_Toc534976591)

[Tabel 4.6 Penjelasan Kode Sumber 4.5 118](#_Toc534976592)

[Tabel 4.7 Penjelasan Kode Sumber 4.6 120](#_Toc534976593)

[Tabel 4.8 Penjelasan Kode Sumber 4.7 121](#_Toc534976594)

[Tabel 4.9 Penjelasan Kode Sumber 4.8 123](#_Toc534976595)

[Tabel 4.10 Penjelasan Kode Sumber 4.9 125](#_Toc534976596)

[Tabel 5.2 Pengujian Memonitor Tingkat Meaningful Learning 128](#_Toc534976597)

[Tabel 5.3 Pengujian Melihat Detail Nilai Aspek 129](#_Toc534976598)

[Tabel 5.4 Pengujian Melihat Detail Nilai setiap Pelajar 131](#_Toc534976599)

[Tabel 5.5 Evaluasi Pengujian Fungsionalitas 135](#_Toc534976600)

# DAFTAR KODE SUMBER

[Kode Sumber 4.1 Implementasi Kasus Penggunaan Monitor Tingkat Meaningful Learning 71](#_Toc534976601)

[Kode Sumber 4.2 Implementasi Kasus Penggunaan Melihat Detail Nilai Aspek Kooperatif 75](#_Toc534976602)

[Kode Sumber 4.3 Implementasi Kasus Penggunaan Melihat Detail Nilai setiap Pelajar 81](#_Toc534976603)

[Kode Sumber 4.4 Perhitungan Frekuensi pada Data Konten 103](#_Toc534976604)

[Kode Sumber 4.5 Perhitungan Frekuensi pada Data Konteks 118](#_Toc534976605)

[Kode Sumber 4.6 Perhitungan Bobot Tindakan Aktivitas Moodle 120](#_Toc534976606)

[Kode Sumber 4.7 Perhitungan Aktivitas Pelajar pada Moodle 121](#_Toc534976607)

[Kode Sumber 4.8 Perhitungan Aspek Kooperatif 123](#_Toc534976608)

[Kode Sumber 4.9 Perhitungan Tingkat Keterlibatan Pelajar 125](#_Toc534976609)

***(Halaman ini sengaja dikosongkan)***

# BAB I PENDAHULUAN

## Latar Belakang

Sejalan dengan berkembangnya teknologi, penggunaan *e-learning* dalam pendidikan tinggi menjadi semakin popular dan menjadi komponen penting dalam dunia pendidikan [1]. *E-learning* merupakan model pembelajaran yang dapat menjadi pendukung ruang kelas tradisional dan menawarkan pendidikan dengan lebih sedikit ruang atau pembatasan waktu, pendidikan dimana diskriminasi terhadap usia dan ras hampir tidak ada, dan pencatatan lebih mudah [2].

Sementara itu, pada pembelajaran tradisional seperti pembelajaran di kelas yang membutuhkan tatap muka hanya mengandalkan ingatan pengajar dalam penilaian setiap pelajarnya. Karena beberapa pengajar hanya mengingat orang-orang tertentu dalam suatu kelas, sehingga penilaian menjadi lebih subjektif dan pembelajaran menjadi kurang bermakna.

Ausubel seorang ahli psikologi pendidikan menyatakan bahwa bahan pelajaran yang dipelajari harus “bermakna” (*meaningful*) [3]. *Meaningful learning* (pembelajaran bermakna) merupakan suatu proses mengkaitkan informasi baru pada konsep-konsep relevan yang terdapat dalam struktur kognitif seseorang. Struktur kognitif ialah fakta-fakta, konsep-konsep, dan generalisasi-generalisasi yang telah dipelajari dan diingat pelajar. Pembelajaran bermakna terjadi ketika pelajar itu aktif, konstruktif, intensional, kooperatif dan bekerja pada tugas autentik [4].

Dalam penelitian Tenriawaru telah ditemukan rumus untuk mengukur pembelajaran bermakna [5]. Namun perhitungan yang dilakukan dalam penelitian tersebut masih manual. Rumus yang terdapat pada penelitian tersebut juga masih memiliki batasan yaitu asumsi bahwa setiap pelajar pasti menggunakan semua aktivitas yang terdapat dalam Moodle.

Oleh karena itu, penambahan *plugin* untuk penilaian pembelajaran bermakna pada *e-learning* diperlukan. Dalam tugas akhir ini akan berfokus pada pembuatan *plugin* untuk aspek kooperatif. *Plugin* tersebut akan dibangun menggunakan Moodle yang merupakan salah satu aplikasi CMS (*Course Management System*) berbasis *web*. *Plugin* ini dibuat untuk membantu para pengajar dalam melakukan penilaian pada tiap pelajar berdasarkan aktifitas pelajar pada *e-learning* sehingga penilaian menjadi lebih objektif.

## Rumusan Masalah

Berdasarkan hal-hal yang telah diuraikan, maka hal-hal yang menjadi permasalahan dalam tugas akhir ini adalah:

* 1. Bagaimana cara merubah data *log* dari setiap aktivitas yang terdapat pada Moodle ke dalam bentuk nilai kuantitatif sesuai dengan parameter yang dibutuhkan?
  2. Bagaimana cara merumuskan perhitungan aspek kooperatif pada *plugin* jika tidak semua aktivitas yang terdapat pada Moodle dipakai oleh pengajar?
  3. Bagaimana menentukan level apa saja yang digunakan untuk merepresentasikan tingkat keterlibatan pelajar pada aspek kooperatif dalam pembelajaran bermakna?
  4. Bagaimana cara mengintegrasikan pengukuran aspek kooperatif dengan aspek lain dalam pembelajaran bermakna?

## Batasan Masalah

Permasalahan pada tugas akhir ini memiliki beberapa batasan, diantaranya sebagai berikut:

1. *Plugin* dibangun pada Moodle versi 3.3*.*8.
2. *Input* berupa pesan pada aktivitas *assignment, chat, forum, glossary* dan *wiki* adalah berupa teks.
3. Bahasa pengantar yang digunakan dalam Moodle adalah Bahasa Inggris dan Bahasa Indonesia.
4. *Plugin* yang dibangun untuk menghitung pembelajaran bermakna melakukan *update* secara otomatis seminggu sekali, yaitu pada hari Minggu.

## Tujuan

Tujuan dari pengerjaan tugas akhir ini adalah membuat *plugin* Moodle untuk mengukur aspek kooperatif yang mendukung pembelajaran bermakna.

## Manfaat

Hasil dari pengerjaan tugas akhir ini dapat memudahkan pengajar yang menggunakan sistem *e-learning* Moodle mengukur keterlibatan pelajar pada aspek kooperatif dalam pembelajaran bermakna. Selain itu, pengajar juga dapat menggunakan *e-learning* sebagai bahan evaluasi untuk pengajar supaya dapat mendukung ataupun memotivasi pelajar yang kurang terlibat dalam pembelajaran bermakna pada *e-learning*.

## Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

### Penyusunan Proposal Tugas Akhir

Proposal tugas akhir ini terdiri dari deskripsi pendahuluan yang menjabarkan latar belakang dan rumusan masalah yang mendasari dibangunnya aplikasi ini, batasan masalah dalam pembangunan aplikasi ini, serta tujuan dan manfaat yang diharapkan dapat dicapai dengan dibangunnya aplikasi ini. Selain itu, pada proposal tugas akhir ini juga terdapat tinjauan pustaka yang menjelaskan teori-teori yang menjadi dasar pembuatan tugas akhir ini.

### Studi Literatur

Pada tahap ini, akan dicari studi literatur yang relevan untuk dijadikan referensi dalam pengerjaan tugas akhir. Studi literatur ini didapatkan dari buku, internet, dan materi kuliah yang berhubungan dengan metode yang akan digunakan.

### Analisis dan Desain Perangkat Lunak

Tahap ini meliputi perumusan kebutuhan fungsional, kebutuhan non-fungsional, kasus penggunaan, diagram aktivitas, diagram kelas, diagram sekuens, rancangan antarmuka pengguna untuk akun admin, dosen, dan mahasiswa, serta pembuatan rancangan basis data.

### Implementasi Perangkat Lunak

Aplikasi ini diimplementasikan dengan menggunakan kakas bantu:

1. Moodle versi 3.3.8.

2. Bahasa pemrograman PHP.

3. *Database* MySQL.

4. Sublime Text sebagai *text editor* dalam pengerjaan *plugin*

### Pengujian dan Evaluasi

Pengujian akan dilakukan dengan data 2 mata kuliah (*course*), dengan menguji fungsionalitas dari aplikasi yang ada. Data diambil dari dua kelas paralel pada matakuliah bidang Rekayasa Perangkat Lunak semester ganjil tahun ajaran 2018/2019 untuk strata 1 dengan mengisikan aktivitas-aktivitas yang dibutuhkan dalam pembuatan *plugin*.

### Penyusunan Buku

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan

a. Latar Belakang

b. Rumusan Masalah

c. Batasan Masalah

d. Tujuan

e. Manfaat

f. Metodologi Pembuatan Tugas Akhir

g. Sistematika Penulisan Laporan Tugas Akhir

2. Tinjauan Pustaka

3. Analisis dan Perancangan Sistem

4. Implementasi

5. Pengujian dan Evaluasi

6. Kesimpulan dan Saran

7. Daftar Pustaka

## Sistematika Penulisan Laporan

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini:

Bab I Pendahuluan

Bab yang berisi mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu metodologi yang digunakan dan sistematika penulisan laporan akhir juga merupakan bagian dari bab ini.

Bab II Tinjauan Pustaka

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

Bab III Analisis dan Perancangan Sistem

Bab ini berisi tentang analisis permasalahan, deskripsi umum sistem, spesifikasi kebutuhan perangkat lunak, lingkungan perancangan, perancangan arsitektur sitem, diagram kelas, dan struktur data.

Bab IV Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa kode sumber yang digunakan untuk proses implementasi.

Bab V Pengujian dan Evaluasi

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

Bab VI Kesimpulan dan Saran

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

# BAB II DASAR TEORI

Pada bab ini akan dibahas mengenai teori-teori yang menjadi dasar dari pembuatan tugas akhir ini. Pembahasan bertujuan untuk memberikan gambaran secara umum terhadap sistem yang dibuat dan berguna sebagai penunjang dalam pengembangan tugas akhir.

## Pembelajaran Bermakna

Ausubel merupakan orang yang pertama kali memperkenalkan konsep pembelajaran bermakna [6]. Ausubel membagi pembelajaran bermakna ke dalam tiga buah aspek, yaitu hubungan, masuk akal, dan relevan. Setelah itu, beberapa ahli membahas konsep pembelajaran bermakna secara meluas [6]. Salah satunya adalah Jonassen yang membagi pembelajaran bermakna ke dalam lima aspek, yakni aktif, autentik, konstruktif, kooperatif, dan intensional [4].



Gambar 2.1 Aspek *Meaningful Learning* [7]

Kelima aspek pembelajaran bermakna saling terhubung, interaktif, dan tergantung, seperti yang terlihat pada Gambar 2.1. Oleh karena itu, pembelajaran harus melibatkan dan mendukung kombinasi dari aktif, konstruktif, intensional, autentik, dan kooperatif [4].

Setiap aspek pembelajaran bermakna memiliki ciri tertentu. Sebuah aspek memberikan gambaran dari sifat dari pelajar. Aspek aktif mengilustrasikan pelajar sebagai seorang yang bersifat manipulatif dan observatif, pelajar konstruktif memiliki sifat artikulatif dan reflektif, pelajar intensional memiliki sifat berorientasi tujuan dan regulatori, pelajar autentik bersifat kompleks dan konstektual, sedangkan pelajar kooperatif cenderung bersifat kolaboratif dan konversational.

## Aspek Kooperatif

Seperti disebutkan di atas, bahwa pembelajaran bermakna terbagi ke dalam lima aspek, yakni aktif, autentik, konstruktif, kooperatif, dan intensional. Dalam hal ini yang dibahas yaitu penilaian aspek kooperatif.

Manusia secara alami bekerjasama dalam komunitas pembelajaran, saling mengeksploitasi keahlian dan menguasai pengetahuan dalam usaha untuk menyelesaikan masalah atapun melaksanakan tugas. Percakapan antar pelajar sangat dibutuhkan dalam sebuah kolaborasi sehingga dalam sebuah kelompok sosial nantinya para pelajar dapat menegosiasikan pemahaman bersama tentang tugas dan metode yang akan mereka gunakan untuk mencapainya. Jadi, pada dasarnya orang secara alami mencari pendapat dan ide-ide dari orang lain untuk menyelesaikan masalah atau tugasnya.

Ketika pelajar terlibat dalam komunitas pembinaan pengetahuan di kelas maupun di luar kelas, mereka telah belajar bahwa ada lebih dari satu cara untuk melihat dunia dan ada beberapa solusi untuk masalah-masalah kehidupan. Kesimpulannya, percakapan harus didorong karena hal ini merupakan cara yang lebih alami untuk membuat makna.

Berdasarkan penelitian yang dilakukan oleh Tenriawaru diambil satu contoh data tindakan dari salah satu pelajar dan didapatkan hasil perhitungan keserupaan semantik antara aspek kooperatif dengan tindakan aktivitas Moodle dari pelajar seperti pada Tabel 2.1 [5]. Perhitungan tersebut didapatkan melalui hasil perhitungan semantik dari fakta kunci pada setiap tindakan pelajar dengan fakta kunci dari aspek kooperatif [5].

Tabel 2.1. Hasil Perhitungan Keserupaan Semantik antara Aspek Kooperatif Meaningful Learning dan Tindakan Aktivitas Moodle dari pelajar [5]

|  |  |  |
| --- | --- | --- |
| Aktivitas Moodle | Tindakan | Aspek Kooperatif |
| Assignment | View | 0,850 |
| Submit | 0,655 |
| Blog | Create | 0,668 |
| View | 0,727 |
| Chat | Chat | 0,494 |
| Readlog | 0,752 |
| Course | View | 0,753 |
| Forum | Viewdiscussion | 0,802 |
| Startdiscussion | 0,678 |
| Replypost | 0,708 |
| Deleteownpost | 0,667 |
| allowforcesubscribe | 0,464 |
| Glossary | Write | 0,642 |
| View | 0,722 |
| Quiz | Attempt | 0,662 |
| reviewmyattempts | 0,724 |
| View | 0,827 |
| Wiki | viewpage | 0,758 |
| createpage | 0,442 |
| Editpage | 0,578 |
| participate | 0,578 |
| viewcomment | 0,765 |

Berdasarkan penelitian Tenriawaru didapatkan perhitungan nilai aspek kooperatif pelajar dimulai dengan perhitungan bobot keserupaan semantik berdasarkan persamaan 2.1 dan skor keserupaan semantik pada Tabel 2.1 [5].

(2.1)

Bobot untuk tindakan *assignment\_view* diperoleh dari skor tindakan *view* pada aktivitas *assigment* terhadap aspek kooperatif dibagi dengan jumlah skor seluruh tindakan yang terlibat terhadap aspek kooperatif seperti yang terdapat pada persamaan 2.2.

(2.2)

Sehingga diperoleh bobot tindakan *assignment\_view* untuk aspek kooperatif yaitu 0,05699 sesuai dengan perhitungan yang terdapat pada persamaan 2.3.

= 0,05699 (2.3)

Dengan cara yang sama diperoleh bobot untuk tindakan lainnya. Hasil perhitungan bobot untuk semua tindakan yang terlibat ditunjukkan pada Tabel 2.2.

Tabel 2.2. Daftar Bobot Tindakan dalam Perhitungan Nilai Aspek Kooperatif [5]

|  |  |  |
| --- | --- | --- |
| Aktivitas Moodle | Tindakan | Bobot |
| Assignment | View | 0,0478 |
| Submit | 0,0451 |
| Blog | Create | 0,0416 |
| View | 0,0488 |
| Chat | Chat | 0,0465 |
| Readlog | 0,0438 |
| Course | View | 0,0532 |
| Forum | Viewdiscussion | 0,0482 |
| Startdiscussion | 0,0448 |
| Replypost | 0,0490 |
| Deleteownpost | 0,0474 |
| allowforcesubscribe | 0,0376 |
| Glossary | Write | 0,0396 |
| View | 0,0474 |
| Quiz | Attempt | 0,0423 |
| reviewmyattempts | 0,0474 |
| View | 0,0465 |
| Wiki | viewpage | 0,0488 |
| createpage | 0,0359 |
| Editpage | 0,0447 |
| participate | 0,0447 |
| viewcomment | 0,0488 |
| Total | | 1 |

Tabel 2.3. Contoh Nilai f(t) dan Nilai Aspek Kooperatif untuk Pelajar [5]

|  |  |  |  |
| --- | --- | --- | --- |
| Aktivitas Moodle | Tindakan (t) |  | \* Bobot |
| Assignment | View | 1 | 0,0478 |
| Submit | 1 | 0,0451 |
| Blog | Create | 1 | 0,0416 |
| View | 1 | 0,0488 |
| Chat | Chat | 1 | 0,0465 |
| Readlog | 1 | 0,0438 |
| Course | View | 1 | 0,0532 |
| Forum | Viewdiscussion | 1 | 0,0482 |
| Startdiscussion | 1 | 0,0448 |
| Replypost | 1 | 0,0490 |
| Deleteownpost | 0 | 0,0000 |
| allowforcesubscribe | 1 | 0,0376 |
| Glossary | Write | 1 | 0,0396 |
| View | 1 | 0,0474 |
| Quiz | Attempt | 1 | 0,0423 |
| reviewmyattempts | 1 | 0,0474 |
| View | 1 | 0,0465 |
| Wiki | viewpage | 1 | 0,0488 |
| createpage | 0,875 | 0,0314 |
| Editpage | 1 | 0,0447 |
| participate | 1 | 0,0447 |
| viewcomment | 1 | 0,0488 |
|  | | | 0,9481 |

Selanjutnya dihitung nilai fungsi untuk setiap tindakan t. Berdasarkan nilai dan bobot ( dihitung nilai aspek kooperatif dengan menggunakan persamaan 2.4.

(2.4)

Contoh hasil perhitungan nilai fungsi untuk setiap tindakan pelajar dan nilai aspek kooperatif pelajar tersebut ditunjukkan pada Tabel 2.3. Berdasarkan Tabel 2.3 diperoleh nilai aspek kooperatif dari salah satu pelajar adalah 0,9481. Perhitungan nilai aspek kooperatif tersebut akan dilakukan pada modul aspek kooperatif dalam plugin di Moodle.

## *E-learning*

*E-learning* merupakan metode pembelajaran yang mempunyai banyak kelebihan, yaitu tidak dibatasi oleh waktu, jarak, maupun ruang sehingga penggunaannya di dalam pendidikan tinggi semakin popular. Moodle adalah salah satu *e-learning platform* atau *Course Management System* (CMS) yang dapat digunakan secara gratis dan dapat dimodifikasi sesuai kebutuhan karena *source code* nya tersedia (*open source*). Berikut akan dibahas mengenai filosofi Moodle, aktivitas, arsitektur, *plugin,* serta *scheduled task* yang terdapat pada Moodle.

### Filosofi Moodle

Moodle dirancang untuk mendukung sebuah gaya belajar yang interaktif yang disebut dengan pedagogi konstruksionis sosial. Filosofi konstruksionis sosial percaya bahwa cara terbaik untuk belajar adalah dari sudut pandang pelajar tersebut sendiri. Pada Moodle, pengajar dapat berkomunikasi ataupun memahami kebutuhan tiap pelajar secara perorangan. Pengajar akan menjadi moderator dalam diskusi serta aktivitas yang berguna untuk mencapai tujuan belajar. Moodle menyediakan beberapa fasilitas untuk pengajar dalam rangka menyusun materi pembelajarannya, yaitu:

a.Pengajar dapat menambah lima jenis materi pembelajarannya yang bersifat statis. Materi pembelajaran ini dapat dilihat dan dibaca oleh pelajar tetapi mereka tidak bisa melakukan tindakan lainnya dengan:

1) Sebuah halaman teks.

2) Sebuah halaman *web*.

3) Sebuah *link* untuk apapun di *web*.

4) Sebuah *view* ke salah satu direktori *course*.

5) Sebuah label yang menampilkan teks atau gambar.

b.Pengajar dapat menambahkan enam jenis materi pelajaran interaktif. Para pelajar dapat berinteraksi dengan materi pelajaran ini dengan menjawab pertanyaan, memasukkan teks, atau meng-*upload* *file* (berkas):

1) *Assignment* (tugas)

2) *Choice*

3) *Journal*

4) *Lesson*

5) *Quiz* (kuis)

6) *Survey*

c.Pengajar juga dapat menggunakan lima jenis kegiatan dimana pelajar berinteraksi satu sama lain. Ini digunakan untuk membuat materi pelajaran sosial:

1) *Chat* (obrolan)

2) *Forum*

3) *Glossary* (glossarium)

4) *Wiki*

5) *Workshop*

### Aktivitas Moodle

Moodle memiliki banyak aktivitas, baik yang bersifat *asynchronous* maupun *synchronous* yang dapat digunakan oleh pengajar dan pelajar. Berikut deskripsi singkat dari setiap aktivitas Moodle [7], [8]:

1. *Assignment*

Ruang dimana pengajar dapat menentukan model tugas yang akan diberikan kepada pelajar, mengumpulkan tugas pelajar, meninjau dan memberikan umpan balik/*feedback* termasuk nilai.

1. *Blog*

Bentuk jurnal online yang diselenggarakan sebagai rangkaian kronologis *posting* yang dibuat oleh pengguna *blog* (yaitu pelajar atau pengajar). Moodle memungkinkan pengguna untuk mendaftar *blog* eksternalnya, sehingga secara otomatis disertakan dalam *blog* Moodle mereka.

1. *Chat*

Fasilitas komunikasi *synchronous* sederhana yang memungkinkan para pengguna berkomunikasi secara *real-time* melalui *web* yang dapat berguna untuk mendapatkan pemahaman yang berbeda satu sama lain dan topik yang dibahas.

1. *Choice*

Ruang dimana seorang pengajar dapat memberi pertanyaan sekaligus menyediakan berberapa pilihan jawaban.

1. *Courses*

Ruang yang memungkinkan pelajar dapat melihat materi pembelajaran yang disiapkan oleh pengajar.

1. *Feedback*

Sebuah ruang untuk menyimpan atau mengumpulkan umpan balik. Hasil dari umpan balik ini dapat disimpan oleh pengajar sebagai rahasia ataupun tidak.

1. *Forum*

Fasilitas komunikasi *synchronous* di Moodle. *Forum* merupakan fasilitas utama untuk diskusi *online*. Pengajar dan pelajar dapat berkomunikasi satu sama lainnya kapan saja, dari mana saja dengan internet.

1. *Glossary*

Ruang memungkinkan peserta untuk menyimpan, membuat dan memelihara daftar definisi, seperti kamus.

1. *Lesson*

Memberikan konten dengan cara yang fleksibel.

1. *Quiz*

Ruang dimana pengajar dapat merancang dan mengatur bentuk tes atau ujian.

1. *Survey*

Ruang untuk mengumpulkan data dari pelajar yang dapat membantu pengajar mengetahui tentang kelas mereka dan merefleksi apa yang telah diajarkan.

1. *Wiki*

Ruang yang berfungsi untuk mengumpulkan halaman *web* yang mengijinkan setiap pengguna dapat menambah atau mengeditnya. Artinya, *wiki* adalah sebuah koleksi dokumen *web* yang ditulis bersama-sama.

1. *Workshop*

Kegiatan penilaian sejawat dengan banyak pilihan. Pelajar menyerahkan pekerjaan mereka melalui alat teks *online* dan lampiran.

### Arsitektur Moodle

Moodle terstruktur sebagai *application core* yang dikelilingi oleh berbagai macam *plugin*. Moodle dirancang agar mudah untuk dikembangkan dan dapat disesuaikan tanpa memodifikasi *application core*, karena hal itu akan menjadi masalah saat melakukan *upgrade* Moodle ke versi yang lebih baru. Sehingga untuk menyesuaikan atau memperluas instalasi Moodle, dilakukan melalui arsitektur *plugin*.

Secara fisik, *plugin* Moodle merupakan *folder* skrip PHP (HTML, CSS, JavaScript, dll.). Inti Moodle atau Moodle core berkomunikasi dengan *plugin* dengan mencari titik masuk tertentu, sering didefinisikan dalam berkas lib.php di dalam *plugin*. Moodle core menyediakan semua infrastruktur yang diperlukan untuk membangun CMS. Dengan mengimplementasikan konsep yang harus digunakan oleh semua *plugin* yang berbeda yaitu seperti *Course and Activities, Users, Course Enrolment, User Functionality*, dan sebagainya.

### *Plugin* Moodle

Terdapat berbagai macam tipe *plugin* pada Moodle sesuai dengan versi Moodle yang digunakan. Setiap tipe *plugin* memiliki fungsi tersendiri. Terdapat beberapa tipe *plugin* beserta deskripsinya seperti pada Tabel 2.4:

Tabel 2.4. Tipe Plugin pada Moodle

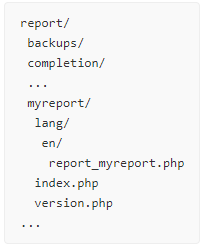
|  |  |  |  |
| --- | --- | --- | --- |
| No | Tipe Plugin | Deskripsi | Versi Moodle |
| 1 | *Activity Modules* | Jenis plugin penting dalam Moodle karena mereka menyediakan aktivitas dalam course. Misal: Forum, Quiz, Assignment | 1.0+ |
| 2 | *Database Fields* | Untuk menambahkan berbagai jenis data ke Activity Modules | 1.6+ |
| 3 | *Report* | Menyediakan tampilan data yang berguna di situs Moodle untuk admin dan guru | 2.2+ |

Sedangkan untuk struktur *database* Moodle didefinisikan dalam berkas install.xml di dalam folder db di setiap *plugin*. Misalnya berkas mod / forum / db / install.xml berisi definisi *database* untuk modul forum. Berkas lib / db / install.xml mendefinisikan tabel yang digunakan oleh inti Moodle. Berkas install.xml berisi komentar yang harus menjelaskan tujuan setiap tabel dan kolom.

### *Plugin* *Report*

*Report* adalah folder berisi kode di bawah direktori *report*. Untuk membuat *plugin report* kita harus membuat folder di sana yang disebut *meaningfullearning*. Satu-satunya kode yang harus dimiliki adalah berkas index.php. Berkas ini biasanya akan menampilkan form HTML sederhana untuk mengontrol laporan, dan, kode untuk menampilkan laporan. Jika kita menginginkan berkas Bahasa (misal: en), maka masukkan berkas tersebut pada folder /report/meaningfullearning yang akan disebut lang/en/report\_meaningfullearning.php.

Selain itu juga dapat menambahkan kode PHP lain yang dibutuhkan, lalu tautkan pada berkas index.php. Jadi, tata letak kode minimal adalah:



Gambar 2.2 Direktori Minimal Pembuatan Plugin dengan Tipe Report

### *Scheduled Task*

*Scheduled Task* merupakan sekumpulan perintah yang dilakukan pada waktu tertentu yang telah ditentukan pada Moodle. Jadwal waktu dapat diubah sesuai dengan kebutuhan masing-masing. Admin memiliki kewenangan untuk mengubah jadwal pada *scheduled task* [9]*.*

## Keserupaan Semantik

Tugas akhir ini menggunakan metode yang diusulkan oleh Simpson dan Dao dalam menghitung keserupaan semantik antara dua buah kalimat [10]. Langkah-langkah untuk menghitung keserupaan semantik antara dua buah kalimat yaitu[10]:

1. Tokenisasi

Proses pemisahan setiap kata dari kalimat dan menghapus kata-kata yang berpotensi menjadi *stop word*. *Stop word* adalah kata-kata yang sering muncul serta tidak memiliki makna. Proses pemisahan menggunakan *parsing* berdasarkan pada spasi pemisah pada masing-masing kata.

1. *Stemming*

Proses pembuangan akhiran atau imbuhan sehingga kata tersebut kembali menjadi kata dasar. Adapun proses *stemming* dapat digambarkan sebagai berikut:

Bentuk awal → kata dipecah menjadi bentuk *morphesme* yang mungkin → mendapatkan bentuk tengah → bentuk pemisahan.

Sebagai contoh proses *stemming* pada kata *laboratories*

*laboratories → laboratory* + s → *laboratory*.

1. Untuk menyelesaikan permasalahan perhitungan ini menggunakan metode *cosine similarity*. *Cosine similarity* mengukur keserupaan antara dua dokumen vektor. Vektor A mewakili dokumen K1 dan vektor B mewakili dokumen K2. Perhitungan keserupaan semantik antara dokumen K1 dan K2 dengan *cosine similarity* dilakukan dengan menggunakan persamaan 2.5

(2.5)

## NLP Tools

NLP Tools merupakan sebuah *library* yang bertujuan untuk pemrosesan bahasa alami. Dalam tugas akhir ini NLP Tools yang digunakan yaitu NLP Tools dalam bahasa PHP yang dapat digunakan untuk tokenisasi, *stemming,* penghitungan *cosine similarity,* dan lain-lain [11].

## Yandex Translator

Yandex Translator merupakan sebuah *library* yang menyediakan akses ke layanan terjemahan *online* serta mendukung lebih dari 90 bahasa dan dapat memprediksi maupun menerjemahkan kata-kata yang terpisah ataupun teks dalam bentuk kalimat. Pada tugas akhir ini yandex translator digunakan untuk menerjemahkan dari bahasa Indonesia ke bahasa Inggris [12], [13]

***(Halaman ini sengaja dikosongkan)***

# BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas analisis kebutuhan dan rancangan yang akan digunakan untuk membangun perangkat lunak yang diajukan sebagai tugas akhir.

## Analisis Sistem

Pada subbab berikut akan dijelaskan analisis plugin pengukuran aspek kooperatif pada Moodle. Analisis yang dilakukan meliputi analisis permasalahan, deskripsi umum sistem, dan kebutuhan fungsional sistem.

### Analisis Permasalahan

Permasalahan utama yang diangkat pada pembuatan tugas akhir ini adalah bagaimana membuat plugin untuk menghitung penilaian aspek kooperatif pada *e-learning* berbasis Moodle.

Di lingkungan Institut Teknologi Sepuluh Nopember (ITS) terutama di Departemen Informatika, e-learning seringkali digunakan oleh pengajar untuk memasukkan materi kuliah serta melakukan pengumpulan tugas yang bisa diakses oleh pelajar. Selain itu *e-learning* juga memiliki aktivitas berupa *forum, wiki, glossary*, dan *chat* yang sebagian telah digunakan oleh pelajar. Namun, saat ini dari sistem *e-learning* masih belum terdapat *plugin* untuk menghitung penilaian dari aktivitas *e-learning* yang telah disediakan.

Untuk mengatasi permasalahan tersebut, penulis akan membuat *plugin* untuk menghitung penilaian aspek kooperatif pada Moodle secara otomatis agar pengajar dapat melihat nilai dari tiap aktivitas pada *e-learning* untuk mata kuliah yang diampu.

Dalam pembuatannya, *plugin* ini dikembangan pada *plugin* *report* dimana *plugin* ini menyediakan tampilan data yang berguna di situs Moodle untuk admin dan pengajar. Sedangkan untuk struktur *database plugin* didefinisikan dalam berkas install.xml pada folder db di *plugin report*. Setelah itu dilakukan pengukuran penilaian aspek kooperatif pada setiap aktivitas moodle sesuai dengan data konten dan konteks yang telah disediakan. Pengajar dapat melihat nilai keterlibatan pelajar terhadap aspek kooperatif pada setiap mata kuliah jika telah terdapat aktivitas pada *e-learning* dan pelajar yang melakukan aktivitas terkait pada *e-learning*.

### Deskripsi Umum Sistem

Pada tugas akhir ini akan dibuat sebuah *plugin* pengukuran aspek kooperatif pada *e-learning* yang digunakan oleh pengajar untuk melihat nilai pelajar dari tiap aktivitas pada mata kuliah yang diampu. Informasi yang dapat diakses oleh pengajar di antaranya nilai setiap aktivitas *e-learning*, nilai aspek kooperatif pelajar dan kemajuan studi berdasarkan tren waktu. Untuk menjalankan plugin ini, perangkat harus terhubung dengan internet.

Pada Gambar 3.1 dijelaskan bahwa informasi yang diakses oleh pengguna yang dalam hal ini pengajar pada plugin report merupakan hasil dari pengolahan data yang diambil dari database.



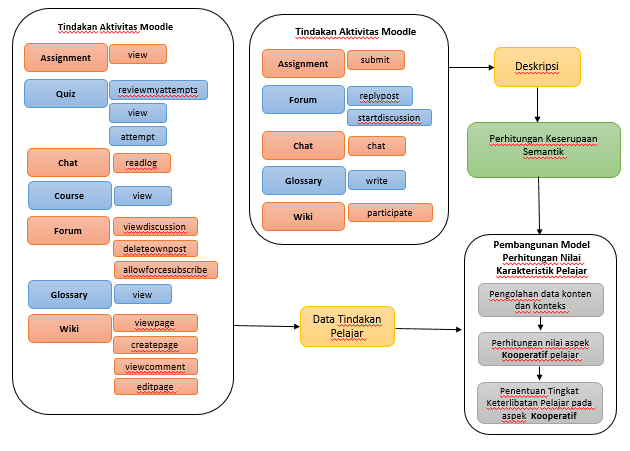
Moodle Server

Plugin Report



Gambar 3.1 Deskripsi Umum Sistem

Sedangkan rancangan pengukuran aspek kooperatif pelajar dalam *e-learning* dimulai dengan menghitung perhitungan keserupaan semantik dan data tindakan pelajar dari aktivitas pada Moodle. Kemudian dilakukan pengolahan data konten dan data konteks, sampai menghasilkan nilai aspek kooperatif. Rancangan model pengukuran aspek kooperatif digambarkan pada Gambar 3.2.



Gambar 3.2 Rancangan Model Pengukuran Aspek Kooperatif Pelajar dalam E-learning

#### Perhitungan Data Log Tindakan Pelajar

Sebelum melakukan proses perhitungan nilai untuk aspek kooperatif, terlebih dahulu dilakukan proses pengolahan data log pelajar, hal ini dilakukan karena di dalam Moodle belum menyediakan data yang sesuai kebutuhan penelitian ini. Moodle mencatat data log pelajar berdasarkan beberapa atribut seperti *id, component, user full name, action*.

Terdapat dua tipe data analitik yang terlibat, yaitu data konten berupa frekuensi dari tindakan pelajar pada aktivitas Moodle dan data konteks berupa hasil analisis konten dari beberapa tindakan tertentu. Sebagai contoh pada aktivitas forum, penelitian ini melibatkan data konten berupa data frekuensi tindakan-tindakan yang ada pada forum dan data konteks berupa hasil analisis konten dari pesan yang ditulis pelajar berupa informasi relevansi isi pesan dengan tema materi pembelajaran dan level kognitif pesan. Selain pada aktivitas forum, data konten dan data konteks diperoleh juga dari aktivitas *assignment*, *chat*, *blog*, *glossary*, dan *wiki*.

#### Data Konten

Data konten diperoleh dari data setiap tindakan pelajar yang tersimpan dalam data mentah yang diperoleh dari tabel mdlnr\_logstore\_standard\_log. Contoh data mentah yang diperoleh dari tabel mdlnr\_logstore\_standard\_log ditunjukkan pada Tabel 3.1. Pada tabel tersebut terdapat atribut id sebagai *primary key* dari tabel, userid yang terhubung pada data di mdlnr\_user, courseid yang terhubung pada tabel mdlnr\_course, component berisi aktivitas yang terdapat pada Moodle, dan action berisi tindakan yang dilakukan pada aktivitas tersebut.

Setiap tindakan yang dilakukan oleh pelajar akan dihitung frekuensinya dan disimpan di dalam atribut yang sesuai. Contoh, setiap tindakan *view* pada aktivitas *quiz* yang dilakukan oleh seorang pelajar akan dihitung dan disimpan pada atribut viewquiz. Tabel 3.2 menampilkan data konten yang dilibatkan dalam tugas akhir ini.

Tabel 3.1 Contoh Data Mentah Log Tindakan Pelajar dalam *E-Learning*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **userid** | **courseid** | **component** | **action** |
| 5340535 | 31818 | 2 | mod\_chat | sent |
| 4278382 | 17507 | 2 | mod\_forum | created |
| 4302145 | 31824 | 2 | mod\_forum | deleted |
| 4539523 | 31821 | 2 | mod\_quiz | viewed |
| 4539531 | 31821 | 2 | mod\_quiz | submitted |
| 4783080 | 14415 | 2 | mod\_glossary | created |
| 5222086 | 31823 | 2 | mod\_wiki | viewed |
| 5222183 | 31818 | 2 | mod\_wiki | created |
| 5341870 | 17136 | 2 | mod\_chat | viewed |
| 5436513 | 17136 | 2 | mod\_wiki | viewed |
| 5443992 | 31821 | 2 | mod\_forum | created |
| 5449190 | 31836 | 2 | mod\_wiki | updated |
| 5479518 | 17568 | 2 | mod\_forum | viewed |

Tabel 3.2 Data Konten yang Terlibat

|  |  |  |
| --- | --- | --- |
| **Aktivitas Moodle** | **Tindakan dalam aktivitas** | **Deskripsi** |
| *Assignment* | *View* | Frekuensi *view* |
| *Blog* | *View* | Frekuensi *view* |
| *Chat* | *readlog* | Frekuensi *readlog* |
| *Course* | *View* | Frekuensi *view* |
| *Forum* | *viewdiscussion* | Frekuensi *viewdiscussion* |
| *deleteownpost* | Frekuensi *deleteownpost* |
| *allowforcesubscribe* | Frekuensi *allowforcesubscribe* |
| *Glossary* | *View* | Frekuensi *comment* |
| *Quiz* | *attempt* | Frekuensi *attempt* |
| *reviewmyattempts* | Frekuensi *reviewmyattempts* |
| *view* | Frekuensi *view* |
| *Wiki* | *viewpage* | Frekuensi *viewpage* |
| *createpage* | Frekuensi *createpage* |
| *editpage* | Frekuensi *editpage* |
| *viewcomment* | Frekuensi *viewcomment* |

#### Data Konteks

Data konteks diperoleh dari hasil analisis konten aktivitas *assignment, blog, chat, forum, glossary*, dan *wiki* pada tindakan *submit, create, chat, startdiscussion, replypost, write*, dan *participate*. Proses pengambilan data konteks dimulai dengan mengecek relevansi konten dengan materi mata kuliah. Konten yang tidak relevan diabaikan atau dianggap nilai atributnya nol. Tabel 3.3 menampilkan data konteks yang dilibatkan dalam tugas akhir ini. Aktivitas pada Moodle akan tersimpan pada tabel mdlnr\_aktivitas pada *plugin report* seperti ditunjukkan pada Tabel 3.4. Sedangkan tindakan setiap aktivitas pada Moodle akan tersimpan pada tabel mdlnr\_aksi pada *plugin report* seperti ditunjukkan pada Tabel 3.5. Relasi antara aktivitas dan tindakan setiap aktivitas pada Moodle akan tersimpan pada tabel mdlnr\_aktivitas\_aksi pada plugin report seperti ditunjukkan pada Tabel 3.6

Tabel 3.3 Data Konteks yang Terlibat

|  |  |  |
| --- | --- | --- |
| **Aktivitas Moodle** | **Tindakan dalam aktivitas** | **Deskripsi** |
| *Assignment* | *Submit* | Frekuensi tugas yang relevan |
| *Blog* | *Create* | Frekuensi *blog* yang relevan |
| *Chat* | *Chat* | Frekuensi *chat* yang relevan |
| *Forum* | *startdiscussion* | Frekuensi pesan yang relevan |
| *replypost* | Frekuensi pesan yang relevan |
| *Glossary* | *write* | Frekuensi *glossary* yang relevan |
| *Wiki* | *participate* | Frekuensi *wiki* yang relevan |

Tabel 3.4 Aktivitas Moodle pada *Plugin Report*

|  |  |
| --- | --- |
| **id** | **nama\_aktivitas** |
| 1 | assignment |
| 2 | blog |
| 3 | chat |
| 4 | course |
| 5 | forum |
| 6 | glossary |
| 7 | quiz |
| 8 | wiki |

Tabel 3.5 Tindakan setiap Aktivitas Moodle pada *Plugin Report*

|  |  |
| --- | --- |
| **id** | **nama\_aksi** |
| 1 | view |
| 2 | submit |
| 3 | create |
| 4 | chat |
| 5 | readlog |
| 6 | viewdiscussion |
| 7 | startdiscussion |
| 8 | replypost |
| 9 | deleteownpost |
| 10 | allowforcesubscribe |
| 11 | write |
| 12 | attempt |
| 13 | reviewmyattempts |
| 14 | viewpage |
| 15 | createpage |
| 16 | editpage |
| 17 | participate |
| 18 | viewcomment |

Tabel 3.6 Relasi Aktivitas dan Tindakan Aktivitas Moodle pada *Plugin Report*

|  |  |  |
| --- | --- | --- |
| **id** | **id\_aksi** | **id\_aktivitas** |
| 1 | 1 | 1 |
| 2 | 2 | 1 |
| 3 | 3 | 2 |
| 4 | 1 | 2 |
| 5 | 4 | 3 |
| 6 | 5 | 3 |
| 7 | 1 | 4 |
| 8 | 6 | 5 |
| 9 | 7 | 5 |
| 10 | 8 | 5 |
| 11 | 9 | 5 |
| 12 | 10 | 5 |
| 13 | 11 | 6 |
| 14 | 1 | 6 |
| 15 | 12 | 7 |
| 16 | 13 | 7 |
| 17 | 1 | 7 |
| 18 | 14 | 8 |
| 19 | 15 | 8 |
| 20 | 16 | 8 |
| 21 | 17 | 8 |
| 22 | 18 | 8 |

1. Penentuan relevansi pesan pada forum



Gambar 3.3 Pendeteksian Relevansi Pesan

Tahap ini terdiri dari beberapa proses antara lain proses menghitung keserupaan semantik antara pesan dengan deskripsi forum dan antara pesan dengan *parent*-nya, menentukan *threshold*, dan melakukan pengecekkan apakah pesan relevan atau tidak berdasarkan keserupaan semantiknya. Desain kerangka kerja ditunjukkan pada Gambar 3.3. Deskripsi setiap forum disimpan pada tabel mdlnr\_forum kolom ‘intro’. Sedangkan pesan dan *parent* disimpan pada tabel mdlnr\_forum\_discussions kolom ‘message’ dan ‘parent’.

1. Perhitungan keserupaan semantik

Proses perhitungan keserupaan semantik dilakukan dengan menghitung keserupaan semantik baik antara pesan dan deskripsi forum maupun antara pesan dan *parent*-nya dengan menggunakan metode keserupaan semantik [10].

1. Penentuan *threshold*

Tahap ini adalah tahap untuk menentukan *threshold*. *Threshold* merupakan nilai parameter batas yang ditentukan untuk mengetahui apakah pesan relevan dengan mata kuliah atau tidak. *Threshold* ditentukan berdasarkan nilai keserupaan semantik baik antara pesan dan deskripsi forum maupun antara pesan dan *parent*-nya. Pada penelitian Tenriawaru, telah dilakukan serangkaian pengujian untuk menentukan *threshold* pada forum [5]. *Threshold* terbaik untuk menentukan relevansi pesan pada forum yaitu 0,6.

1. Penentuan pesan yang relevan dan tidak relevan

Tahap ini adalah tahap untuk mengecek dan menentukan apakah pesan relevan atau tidak relevan dengan forum ataupun *parent* berdasarkan *threshold*. Jika nilai keserupaan ≥ *threshold* maka pesan disimpulkan relevan dengan forum ataupun *parent*. Jika nilai keserupaan < *threshold* maka disimpulkan pesan tidak relevan. Setelah diketahui relevansi pesan terhadap forum maupun *parent*, akan dicek relevansi pesan dengan *course*.

1. Penentuan relevansi *assignment* dan *chat*

Penentuan relevansi kontenpada aktivitas *assignment* dilakukan berdasarkan nilai tugas (*assignment*) yang di *submit* oleh pelajar. Nilai tugas setiap pelajar tersimpan pada tabel mdlnr\_assign\_grades kolom ‘*grade’*. Jika nilai tugas yang diperoleh pelajar ≥ *passing grade*, maka diasumsikan bahwa tugas yang disubmit relevan. Pada penelitian Tenriawaru telah dilakukan serangkaian pengujian untuk menentukan *passing grade* pada nilai tugas hingga didapatkan *passing grade* yang harus dilalui oleh pelajar yaitu 41 [5].

Penentuan relevansi kontenpada aktivitas *chat* dilakukan berdasarkan nilai keserupaan semantik antara konten *chat* dan deskripsi mata kuliah*.* Jika nilai keserupaan semantik ≥ *threshold* maka disimpulkan bahwa konten *chat* relevan dengan mata kuliah. *Threshold* untuk aktivitas chat ditentukan berdasarkan relevansi untuk text pendek yang telah dihasilkan pada penelitian Tenriawaru yaitu 0,581 [5].

1. Penentuan relevansi *blog, glossary*, dan *wiki*

Penentuan relevansi kontenpada aktivitas *blog, glossary*, dan *wiki* dilakukan berdasarkan nilai keserupaan semantik antara konten *blog, glossary*, dan *wiki* terhadap deskripsi mata kuliah*.* Jika nilai keserupaan semantik antara antara konten *blog* dan deskripsi mata kuliah ≥ *threshold* maka disimpulkan bahwa konten *blog* relevan dengan mata kuliah. Demikian halnya untuk *glossary* dan *wiki*.

Konten *glossary* pelajar tersimpan pada tabel mdlnr\_glossary\_entries kolom ‘*definition’* dan deskripsi mata kuliah pada *glossary* tersimpan pada tabel mdlnr\_glossary kolom ‘*intro’*. Sedangkan untuk konten *wiki* pelajar tersimpan pada tabel mdlnr\_wiki\_pages kolom ‘cachedcontent’ dan deskripsi mata kuliah pada *wiki* tersimpan pada tabel mdlnr\_wiki kolom ‘*intro’*.

Pada penelitian Tenriawaru telah dilakukan serangkaian pengujian untuk menentukan *threshold* pada *blog, glossary* dan *wiki*. *Threshold* terbaik untuk menentukan relevansi konten pada *blog* yaitu 0,86, *glossary* yaitu 0,66 dan untuk *wiki* yaitu 0,82 [5].

#### Perhitungan Frekuensi Tindakan Pelajar pada Aktivitas Moodle

Nilai frekuensi atau setiap pelajar untuk setiap tindakan *t* diperoleh dengan cara yang berbeda. Berikut persamaan-persamaan yang digunakan untuk menghitung nilai :

1. Untuk *Assignment\_view* dan *Quiz\_view*, diperoleh dengan menggunakan persamaan 3.1.

(3.1)

1. Untuk *Assignment\_submit* dan *Quiz\_attempt*, diperoleh dengan menggunakan persamaan 3.2.

(3.2)

*s* adalahfrekuensi *Assignment\_submit* atau *Quiz\_attempt* dan *n* adalah banyaknya *assignment* atau *quiz* yang diberikan.

1. Untuk tindakan lainnya, diperoleh dengan menggunakan persamaan 3.3.

*f* (*t*) *=*(3.3)

Dalam persamaan ini, nilai dihitung dengan menggunakan persamaan 3.4.

=  (3.4)

Hasil frekuensi setiap tindakan aktivitas Moodle untuk setiap pelajar dan mata kuliah tertentu akan tersimpan pada kolom ‘frekuensi’ tabel mdlnr\_frekuensi pada *plugin report* seperti ditunjukkan pada Tabel 3.7. Pada tabel tersebut terdapat atribut id sebagai *primary key,* id\_aktivitas\_aksi diambil dari tabel mdlnr\_aktivitas\_aksi, id\_user diambil dari tabel mdlnr\_user, id\_course diambil dari tabel mdlnr\_course, frekuensi yang merupakan hasil dari perhitungan frekuensi tindakan pelajar pada aktivitas Moodle.

Tabel 3.7. Frekuensi Tindakan Pelajar terhadap Aktivitas Moodle pada *Plugin Report*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **id\_aktivitas\_aksi** | **id\_user** | **id\_course** | **frekuensi** |
| 5340535 | 1 | 31818 | 2 | 0 |
| 4278382 | 1 | 17507 | 2 | 1 |
| 4302145 | 1 | 31824 | 2 | 1 |
| 4539523 | 1 | 31821 | 2 | 0 |
| 4539531 | 1 | 31821 | 2 | 0,2 |
| 4783080 | 1 | 14415 | 2 | 1 |
| 5222086 | 1 | 31823 | 2 | 0 |

#### Perhitungan Bobot Tindakan Aktivitas Moodle

Berdasarkan penelitian dari Tenriawaru, didapatkan daftar bobot tindakan dalam perhitungan nilai aspek kooperatif seperti ditunjukkan pada Tabel 2.2 [5]. Namun bobot tersebut berlaku jika semua aktivitas pada Moodle digunakan. Pengecekan aktivitas yang digunakan dan tidak digunakan pada Moodle yaitu pada tabel mdlnr\_logstore\_standard\_log. Dapat diilustrasikan seperti pada Tabel 3.8, jika semua aktivitas pada Moodle digunakan, bobot tidak berubah.

Tabel 3.8 Ilustrasi Aktivitas yang digunakan pada Moodle

|  |  |  |
| --- | --- | --- |
| **Aktivitas Moodle** | **Keterangan** | **Bobot** |
| A | Digunakan | 0,1 |
| B | Digunakan | 0,2 |
| C | Digunakan | 0,3 |

Namun jika terdapat aktivitas yang tidak digunakan seperti aktivitas B pada Tabel 3.9, maka bobot aktivitas B menjadi 0. Dan bobot awal aktivitas B disebar pada bobot aktivitas Moodle yang lain, sehingga bobot aktivitas Moodle yang lain akan mengalami perubahan. Perhitungan persebaran bobot untuk setiap aktivitas menggunakan Persamaan 3.5.

(3.5)

Tabel 3.9 Ilustrasi Aktivitas yang tidak digunakan pada Moodle

|  |  |  |
| --- | --- | --- |
| **Aktivitas Moodle** | **Keterangan** | **Bobot** |
| A | Digunakan | 0,15 |
| B | Tidak Digunakan | 0 |
| C | Digunakan | 0,45 |

Bobot yang telah dihitung dengan persebaran bobot pada setiap tindakan aktivitas Moodle tersimpan pada kolom ‘bobot\_baru’ tabel mdlnr\_bobot pada *plugin report* seperti ditunjukkan pada Tabel 3.10. Pada tabel tersebut terdapat atribut id sebagai *primary key*, id\_aktivitas\_aksi yang diambil dari tabel mdlnr\_aktivitas\_aksi, id\_aspek yang diambil dari tabel mdlnr\_aspek, bobot yang berisi bobot ketika semua aktivitas pada Moodle digunakan, bobot\_baru yang berisi hasil dari persebaran bobot ketika terdapat aktivitas yang tidak digunakan pada Moodle.

Tabel 3.10 Bobot setiap Tindakan Aktivitas Moodle pada *Plugin Report*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **id\_aktivitas\_aksi** | **id\_aspek** | **bobot** | **bobot\_baru** |
| 1 | 1 | 5 | 0,047800 | 0,047800 |
| 2 | 2 | 5 | 0,045100 | 0,045100 |
| 3 | 3 | 5 | 0,041600 | 0,041600 |
| 4 | 4 | 5 | 0,048800 | 0,048800 |
| 5 | 5 | 5 | 0,046500 | 0,046500 |

#### Perhitungan Nilai Aktivitas Pelajar pada Moodle

Perhitungan nilai aktivitas setiap pelajar pada mata kuliah tertentu membutuhkan kolom ‘frekuensi’ pada tabel mdlnr\_frekuensi, dengan menyesuaikan kolom ‘id\_course’ dan ‘mdl\_user’, serta kolom ‘bobot\_baru’ pada tabel mdlnr\_bobot. Sehingga nilai aktivitas pelajar dapat dirumuskan seperti pada persamaan 3.6.

(3.6)

Dimana merupakan frekuensi untuk setiap tindakan dan merupakan bobot untuk setiap tindakan Nilai aktivitas pelajar pada Moodle untuk setiap mata kuliah dan aspek tertentu tersimpan pada kolom ‘nilai’ tabel mdlnr\_log\_aksi pada *plugin report* seperti ditunjukkan pada Tabel 3.11. Dimana aspek kooperatif memiliki id\_aspek=5. Pada tabel tersebut terdapat atribut id sebagai *primary key,* id\_frekuensi diambil dari tabel mdlnr\_frekuensi, id\_bobot diambil dari tabel mdlnr\_bobot, id\_aspek diambil dari tabel mdlnr\_aspek, nilai yang berisi hasil perhitungan nilai aktivitas pelajar pada Moodle

Tabel 3.11 Nilai Aktivitas Pelajar pada Moodle pada Plugin Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **id\_frekuensi** | **id\_bobot** | **id\_aspek** | **nilai** |
| 1 | 35 | 6 | 1 | 0,052455 |
| 2 | 35 | 28 | 2 | 0,060988 |
| 3 | 35 | 50 | 3 | 0,065761 |
| 4 | 35 | 72 | 4 | 0,065419 |
| 5 | 35 | 94 | 5 | 0,056220 |

#### Perhitungan Aspek Kooperatif

Perhitungan aspek kooperatif pada setiap tindakan adalah seperti ditunjukkan pada Persamaan 3.7 [5]

) (3.7)

Dimana adalah frekuensi untuk setiap tindakan dan adalah bobot untuk setiap tindakan . Perhitungan telah tersimpan pada kolom ‘nilai’ tabel mdlnr\_log\_aksi. Sehingga nilai aspek kooperatif pelajar didapatkan dengan menjumlahkan kolom ‘nilai’ pada tabel mdlnr\_log\_aksi dengan id\_aspek=5 dan id\_frekuensi yang sesuai dengan id\_course dan id\_user.

Nilai aspek kooperatif pelajar pada Moodle untuk setiap mata kuliah tertentu tersimpan pada kolom ‘nilai\_akhir’ tabel mdlnr\_nilai\_akhir pada *plugin report* seperti ditunjukkan pada Tabel 3.12. Pada tabel tersebut terdapat atribut id sebagai *primari key,* id\_course diambil dari tabel mdlnr\_course, id\_user diambil dari tabel mdlnr\_user, id\_aspek diambil dari tabel mdlnr\_aspek, nilai\_akhir yang berisi nilai dari aspek pada pembelajaran bermakna, level yang berisi tingkat keterlibatan pelajar pada pembelajaran bermakna, dan week yang berisi waktu penginputan data dalam hitungan minggu.

Tabel 3.12 Nilai Aspek Kooperatif Pelajar pada Moodle

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **id** | **id\_course** | **id\_user** | **id\_aspek** | **nilai\_akhir** | **level** | **week** |
| 1 | 2 | 5 | 1 | 0,524550 | tinggi | 1 |
| 2 | 2 | 5 | 2 | 0,609880 | tinggi | 1 |
| 3 | 2 | 5 | 3 | 0,657610 | tinggi | 1 |
| 4 | 2 | 5 | 4 | 0,654190 | tinggi | 1 |
| 5 | 2 | 5 | 5 | 0,562200 | tinggi | 1 |

#### Penentuan Tingkat Keterlibatan Pelajar pada Aspek Kooperatif

Penentuan tingkat keterlibatan pelajar pada aspek kooperatifdilakukan dengan membagi nilai aspek kooperatifpelajar dalam tiga level, yakni rendah, sedang, atau tinggi. Pembagian level dilakukan berdasarkan persamaan 3.8

Level = (3.8)

Dimana nilai , , , dan secara berturut-turut dihitung menggunakan persamaan 3.9, persamaan 3.10, persamaan 3.11, dan persamaan 3.12

(3.9)

(3.10)

(3.11)

(3.12)

Nilai adalah jumlah pelajar, adalah nilai karakteristik *k* pelajar ke-*i*,adalah rerata nilai karakteristik *k* pelajar dan adalah standar deviasi untuk seluruh . Level pelajar pada Moodle untuk setiap mata kuliah tertentu tersimpan pada kolom ‘level’ tabel mdlnr\_nilai\_akhir pada *plugin report* seperti ditunjukkan pada Tabel 3.12.

### Spesifikasi Kebutuhan Perangkat Lunak

Sesuai dengan cakupan perangkat lunak yang telah dijelaskan pada bagian deskripsi umum sistem, dibutuhkan adanya spesifikasi perangkat lunak agar dapat memberikan solusi dari permasalahan yang diberikan dan dapat bekerja dengan baik dalam mengakomodasi kebutuhan. Diharapkan dengan adanya spesifikasi ini dapat menyesuaikan kebutuhan-kebutuhan pengguna. Spesifikasi kebutuhan perangkat lunak adalah penjelasan mengenai kebutuhan sistem yang diinginkan pelanggan atau klien dalam bentuk tulisan. Spesifikasi kebutuhan perangkat lunak pada tugas akhir ini terdiri dari kebutuhan fungsional, kebutuhan non-fungsional, aktor, dan kasus penggunaan.

#### Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan layanan yang harus dimiliki oleh perangkat lunak, reaksi dari perangkat lunak terhadap suatu masukan, hasil yang dilakukan perangkat lunak pada situasi khusus. Kebutuhan fungsional dari perangkat lunak dijelaskan pada Tabel 3.13.

Tabel 3.13 Kebutuhan Fungsional

|  |  |  |
| --- | --- | --- |
| No | Kebutuhan Fungsional | Deskripsi |
| 1 | Memonitor tingkat meaningful learning pelajar | Memonitor tingkat meaningful learning (tinggi, sedang, rendah) dari setiap pelajar yang terdaftar pada mata kuliah yang dipilih |
| 2 | Melihat detail nilai aspek. | Melihat detail nilai dari aspek aktif, konstruktif, intensional, autentik, dan kooperatif yang terdiri dari beberapa aktivitas (assignment, chat, course, forum, glossary, quiz, wiki). |
| 3 | Melihat detail nilai setiap pelajar | Melihat detail nilai mahasiswa yang terdiri dari aspek aktif, konstruktif, intensional, autentik, kooperatif dari setiap mahasiswa. |

#### Kebutuhan Non-Fungsional

Kebutuhan non fungsional mendefinisikan batasan layanan yang ditawarkan oleh sistem. Kebutuhan non fungsional dari perangkat lunak dijelaskan pada Tabel 3.14.

Tabel 3.14 Kualitas Perangkat Lunak

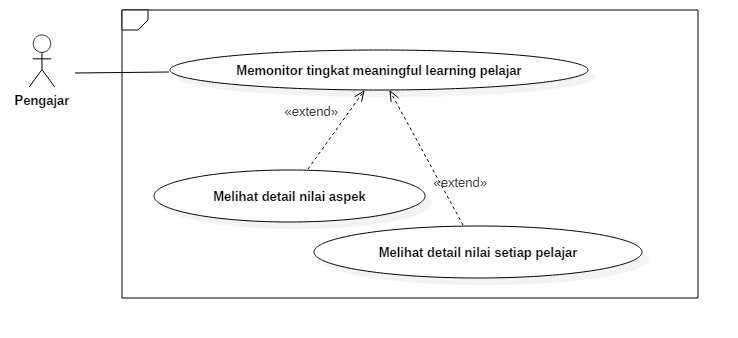
|  |  |  |
| --- | --- | --- |
| No | Parameter | Deskripsi |
| 1 | Bahasa | Bahasa yang digunakan pada antarmuka merupakan bahasa Inggris |
| 2 | Integrity | Hanya user dengan role teacher yang mempunyai wewenang untuk menginstall dan menggunakan plugin report. |
| 3 | Portability | Aplikasi mudah untuk dioperasikan pada browser komputer maupun browser smartphone. |
| 4 | Correctness | Aplikasi menghasilkan keluaran yang benar untuk setiap kemungkinan masukkin oleh user |

#### Aktor

Pengertian pengguna adalah pihak-pihak, baik manusia maupun sistem atau perangkat lain yang terlibat dan berinteraksi secara langsung dengan sistem. Pada perangkat lunak ini terdapat sembilan pengguna yaitu system administrator, manager, course creator, teacher, non-editing teacher, student, guest, authenticated user, dan authenticated user on frontpage. Pada tugas akhir ini, user yang berperan adalah teacher dan student. *Teacher* adalah seorang pengajar yang bisa melihat semua fitur dari meaningful learning berdasarkan mata kuliah yang diampu. Student adalah siswa yang berperan aktif dalam pengisian semua aktivitas moodle yang nantinya digunakan dalam penilaian meaningful learning.

#### Kasus Penggunaan

Berdasarkan analisis spesifikasi kebutuhan fungsional dan analisis aktor dari sistem, dibuat kasus penggunaan sistem. Kasus penggunaan digambarkan dalam tabel penjelasan kasus penggunaan dan diagram kasus penggunaan. Tabel penjelasan kasus penggunaan dapat dilihat pada Tabel 3.15 dan diagram kasus penggunaan dapat dilihat pada Gambar 3.4



Gambar 3.4 Diagram Kasus Penggunaan

Tabel 3.15 Kasus Penggunaan

|  |  |  |
| --- | --- | --- |
| Kode Kasus Penggunaan | Nama | Aktor |
| UC-001 | Memonitor tingkat meaningful learning pelajar | Pengajar |
| UC-002 | Melihat detail nilai aspek | Pengajar |
| UC-003 | Melihat detail nilai setiap pelajar | Pengajar |

##### Memonitor Tingkat *Meaningful Learning* (UC-001)

Kasus penggunaan nomor UC-001 ini diakses ketika pengajar (sebagai pengguna) hendak melihat nilai *meaningful learning* dari mahasiswa yang terdaftar pada mata kuliah pengajar. Nilai *meaningful learning* terdiri dari 3 level (tingkat keterlibatan pelajar) yaitu tinggi, sedang, dan rendah. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.16, Gambar 3.5 dan Gambar 3.6.

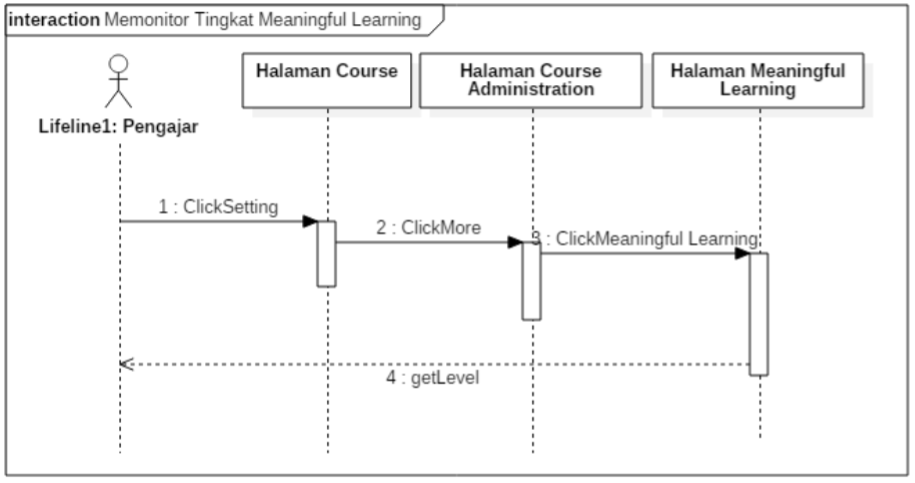
Tabel 3.16 Spesifikasi Kasus Penggunaan UC-001

|  |  |  |
| --- | --- | --- |
| Kode Use Case | UC-001 | |
| Nama Use Case | Memonitor tingkat meaningful learning | |
| Aktor | Pengajar | |
| Deskripsi | Pengajar dapat memonitor tingkat meaningful learning | |
| Relasi | - | |
| Kondisi Awal | Pengajar belum melihat tingkat meaningful learning pelajar | |
| Kondisi Akhir | Pengajar sudah melihat tingkat meaningful learning pelajar | |
| Alur kejadian normal | Pengajar | Sistem |
| 1. Memilih tombol “More” pada pengaturan course |  |
|  | 2. Menampilkan halaman “Course Administration” |
| 3. Memilih menu “Meaningful Learning” pada bagian Reports |  |
|  | 4. Menampilkan halaman *Meaningful Learning* |
| Alur kejadian alternatif | Aktor | Sistem |
| - | - |

A screenshot of a cell phone

Description automatically generated

Gambar 3.5 Diagram Aktivitas UC-001



Gambar 3.6 Diagram Sekuens UC-001



##### Melihat Detail Nilai Aspek (UC-002)

Kasus penggunaan nomor UC-002 ini diakses ketika pengajar hendak melihat detail nilai yang didapatkan dari tiap aspek. Dalam satu aspek, nilai diambil dari beberapa aktivitas pada *course*, seperti *assignment, chat, forum, quiz, course, wiki*, dan *glossary*. Untuk melihat detail nilai aspek, dapat diakses dari halaman monitoring *meaningful learning* pada UC-001. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.17, Gambar 3.7, dan Gambar 3.8

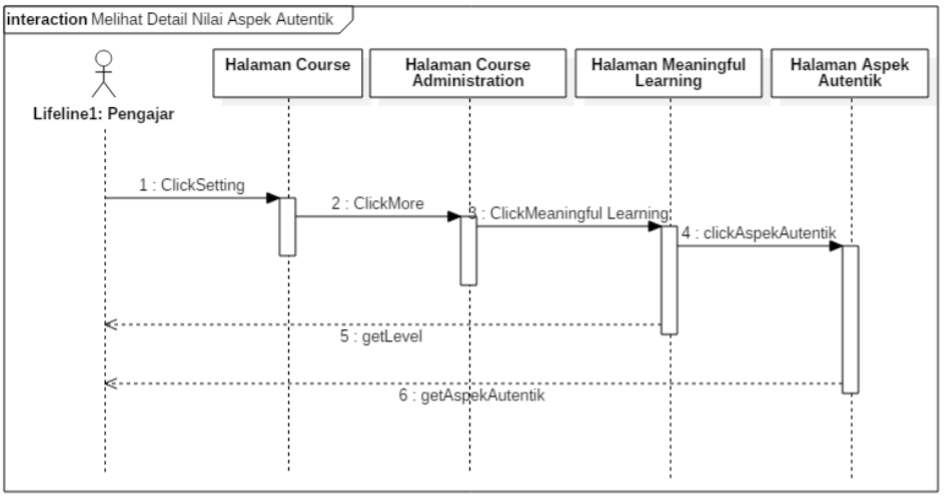
Tabel 3.17 Spesifikasi Kasus Penggunaan UC-002

|  |  |  |
| --- | --- | --- |
| Kode Use Case | UC-002 | |
| Nama Use Case | Melihat detail setiap nilai aspek | |
| Aktor | Pengajar | |
| Deskripsi | Pengajar dapat melihat detail nilai setiap aspek (aktif, konstruktif, intensional, autentik, dan kooperatif) | |
| Relasi | Extend UC-0001 | |
| Kondisi Awal | Pengajar sudah melihat halaman Meaningful Learning | |
| Kondisi Akhir | Pengajar sudah melihat detail nilai setiap aspek | |
| Alur kejadian normal | Aktor | Sistem |
| 1. Memilih menu aspek yang ingin dilihat |  |
|  | 2. Menampilkan detail nilai dari aspek yang dipilih |
| Alur kejadian alternatif | Aktor | Sistem |
| - | - |

A screenshot of a cell phone

Description automatically generated

Gambar 3.7 Diagram Aktivitas UC-002



Gambar 3.8 Diagram Sekuens UC-002



##### Melihat Detail Nilai setiap Pelajar (UC-003)

Kasus penggunaan nomor UC-003 ini diakses ketika pengajar hendak melihat detail nilai yang didapatkan dari tiap pelajar. Dalam satu pelajar, detail nilai ditampilkan berdasarkan aspek yaitu aktif, konstruktif, intensional, autentik, dan kooperatif, selain itu juga ditampilkan berdasarkan aktivitas pada course yang berhasil dilakukan pelajar, misalnya *assignment, chat, forum, quiz, course, wiki*, dan *glossary*. Untuk melihat detail nilai setiap pelajar, dapat diakses dari halaman monitoring *meaningful learning* pada UC-001. Spesifikasi, dan diagram aktivitas kasus penggunaan ini dapat dilihat pada Tabel 3.18, Gambar 3.9, dan Gambar 3.10

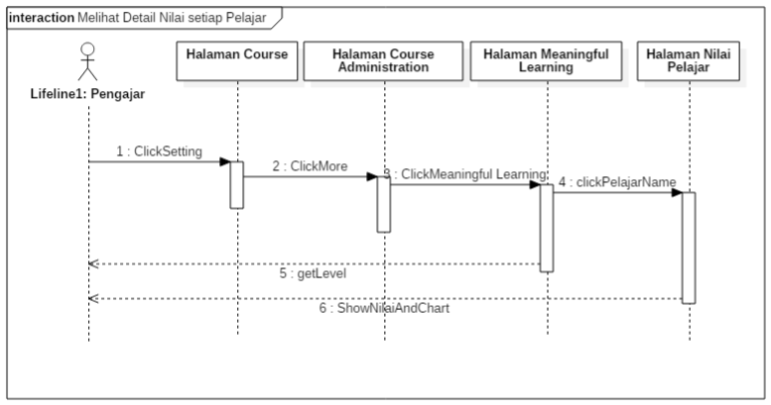
Tabel 3.18 Spesifikasi Kasus Penggunaan UC-0003

|  |  |  |
| --- | --- | --- |
| Kode Use Case | UC-003 | |
| Nama Use Case | Melihat detail nilai setiap pelajar | |
| Aktor | Pengajar | |
| Deskripsi | Pengajar dapat melihat detail nilai setiap pelajar yang terdiri dari nilai aspek dan aktivitas pada course. | |
| Relasi | Extend UC-0001 | |
| Kondisi Awal | Pengajar sudah melihat halaman Meaningful Learning | |
| Kondisi Akhir | Pengajar sudah melihat detail nilai dari pelajar yang dipilih | |
| Alur kejadian normal | Aktor | Sistem |
| 1. Memilih nama pelajar yang ingin dilihat detail nilainya |  |
|  | 2. Menampilkan halaman detail nilai pelajar yang dipilih |
| Alur kejadian alternatif | Aktor | Sistem |
| - | - |

A screenshot of a cell phone

Description automatically generated

Gambar 3.9 Diagram Aktivitas UC-003



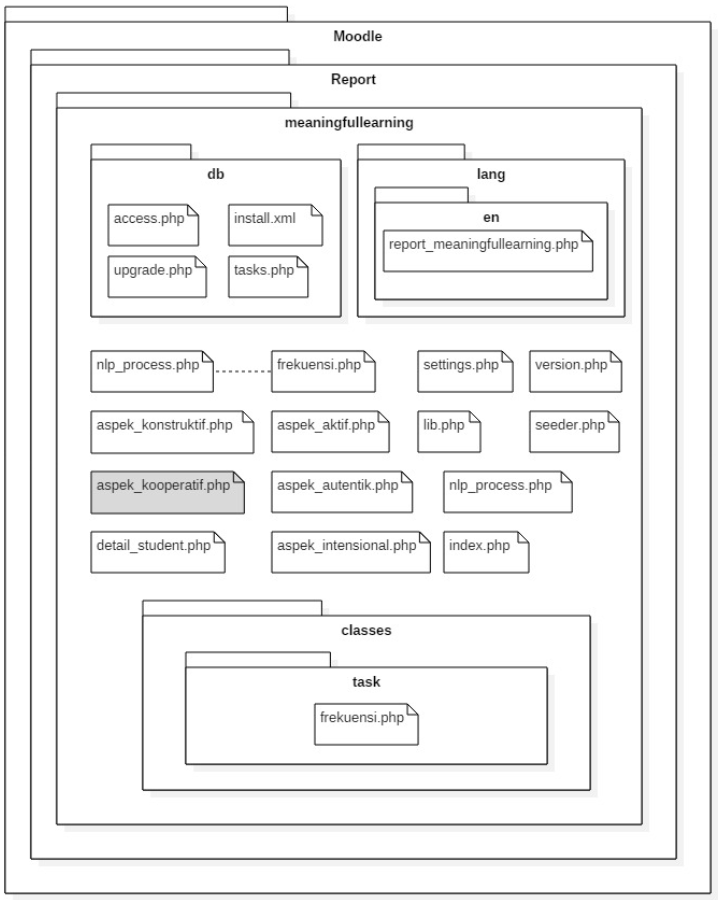
Gambar 3.10 Diagram Sekuens UC-003

## Perancangan

Perancangan dalam subbab ini membahas perancangan dari aplikasi tugas akhir. Subbab ini terdiri dari lingkungan perancangan perangkat lunak, perancangan arsitektur sistem, perancangan diagram kelas, perancangan basis data, dan perancangan antarmuka pengguna.

### Perancangan Arsitektur Sistem

Dari beberapa tipe plugin yang terdapat pada Moodle, dalam tugas akhir ini akan dibangun *plugin* bertipe *report* untuk penilaian aspek kooperatif dalam pembelajaran bermakna karena mampu menyediakan data yang dibutuhkan oleh admin dan pengajar pada Moodle *e-learning*. *Plugin* yang akan dibangun bernama ‘meaningfullearning’. Arsitektur dari *plugin* ini ditunjukkan pada Gambar 3.11.



Gambar 3.11 Arsitektur *Plugin* Report Moodle

Karena *plugin* yang dibangun bertipe *report*, maka *plugin* harus dibangun di dalam direktori *report* pada Moodle, yang diberi nama direktori meaningfullearning. Pada direktori meaningfullearning terdapat direktori db untuk menyimpan struktur *database* pada *plugin*. Berkas db/install.xml bertujuan untuk mendefinisikan kolom-kolom pada tabel yang dibangun pada *plugin*, berkas db/upgrade.php bertujuan untuk memasukkan tabel-tabel yang dibangun ke *database*, berkas db/access.php bertujuan untuk mendefinisikan apa saja yang dapat dilakukan user berdasarkan *role* nya, sedangkan db/tasks.php bertujuan untuk menjadwalkan *task* yang telah didefinisikan pada *plugin*. Direktori lang/en/report\_meaningfullearning.php digunakan untuk mendefinisikan *string* dalam bahasa Inggris pada *plugin* meaningfullearning. Direktori classes/task/frekuensi.php bertujuan untuk mengeksekusi *task* yang terdapat pada *plugin* meaningfullearning.

Kemudian terdapat berkas nlp\_process.php untuk menyimpan *class* yang digunakan untuk *text processing*. Berkas seeder.php untuk menambahkan *record* yang bersifat *default* ke dalam *database plugin* meaningfullearning. Berkas index.php untuk menampilkan seluruh nilai aspek pada *meaningful learning*. Berkas detail\_student.php untuk menampilkan detail nilai aspek *meaningful learning* pada pelajar. Berkas aspek\_aktif.php untuk menampilkan nilai aspek aktif pelajar berdasarkan *course* yang dipilih. Berkas aspek\_autentik.php untuk menampilkan nilai aspek autentik pelajar berdasarkan *course* yang dipilih. Berkas aspek\_konstruktif.php untuk menampilkan nilai aspek konstruktif pelajar berdasarkan *course* yang dipilih. Berkas aspek\_kooperatif.php untuk menampilkan nilai aspek kooperatif pelajar berdasarkan *course* yang dipilih. Berkas aspek\_intensional.php untuk menampilkan nilai aspek intensional pelajar berdasarkan *course* yang dipilih. Berkas frekuensi.php berisi kode perhitungan seluruh aspek *meaningful learning*. Berkas settings.php untuk konfigurasi administrasi pada *plugin* meaningfullearning. Berkas version.php mendefinisikan versi *plugin* yang digunakan. Berkas lib.php mendefinisikan antarmuka antara Moodle *core* dengan *plugin* yang dibangun.

Aspek yang dikerjakan pada tugas akhir ini yaitu aspek kooperatif, dimana kode nilai aspek kooperatif pada *course* yang dipilih terdapat pada berkas aspek\_kooperatif.php . Nilai aspek kooperatif juga terdapat pada index.php yang akan menampilkan seluruh nilai aspek *meaningful learning* pada course yang dipilih.

### Perancangan Diagram Kelas

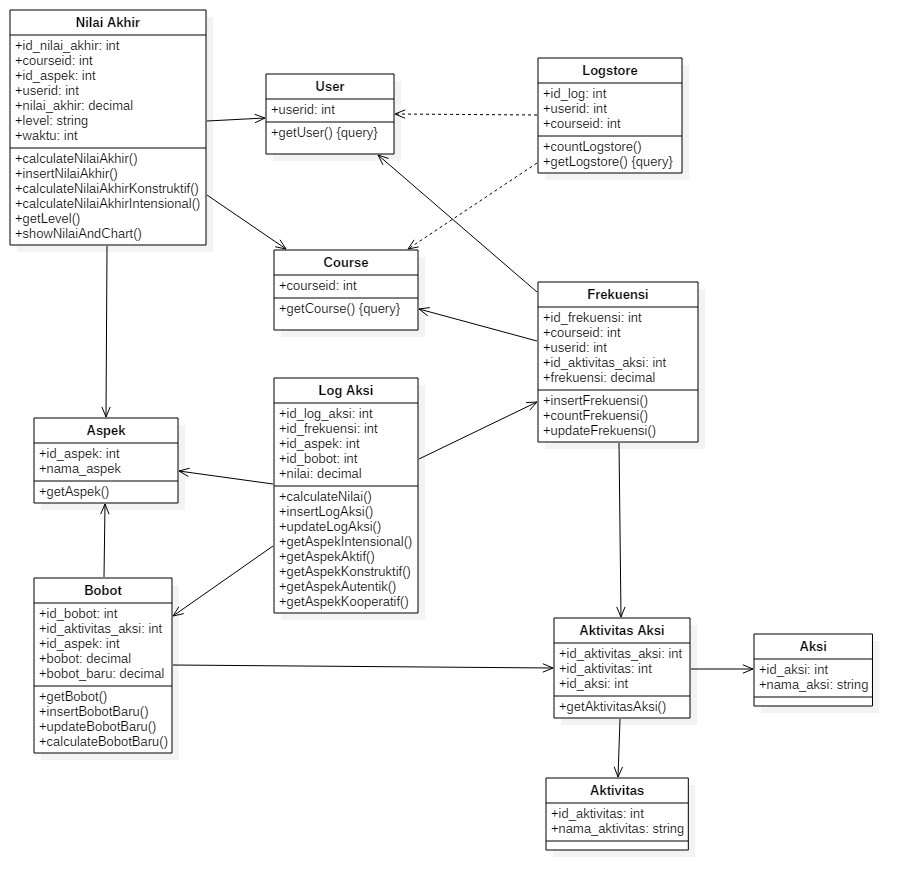
Diagram kelas dapat dilihat pada Gambar 3.12.

### Perancangan Basis Data

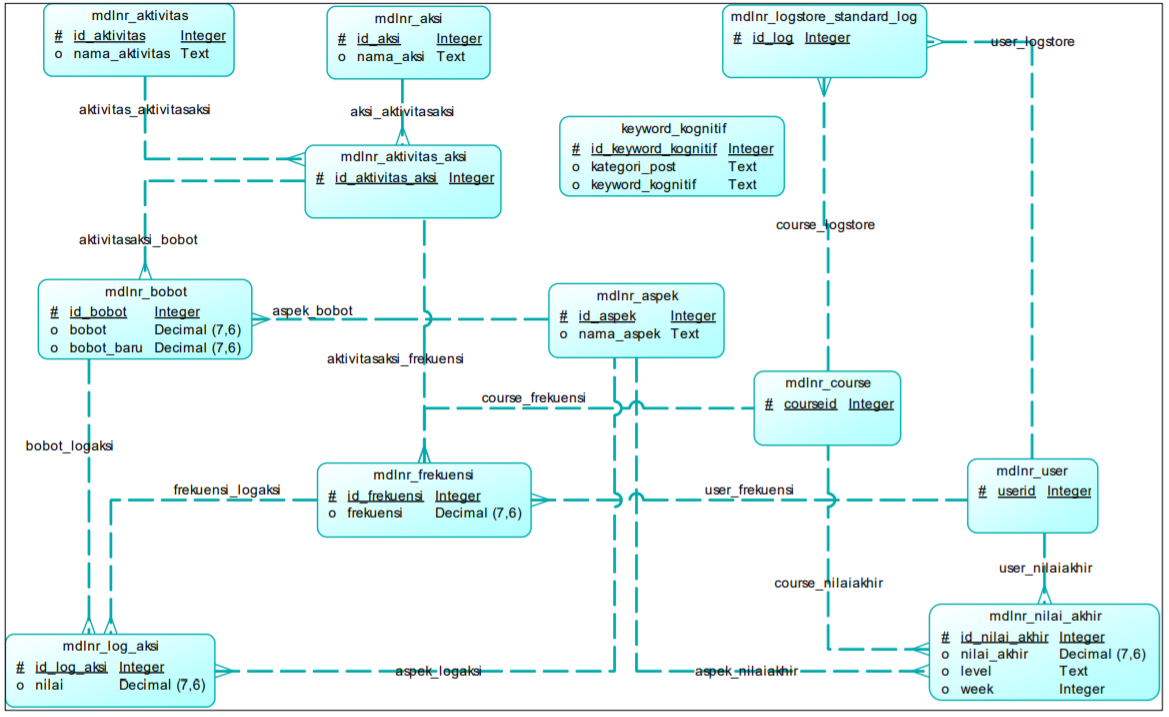
Dalam pembuatan tugas akhir ini, diperlukan analisis kebutuhan berupa perancangan basis data. Basis data yang digunakan adalah phpMyAdmin. Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (CDM) pada Gambar 3.13 dan *Physical Data Model* (PDM) pada Gambar 3.14.

#### Tabel mdlnr\_course

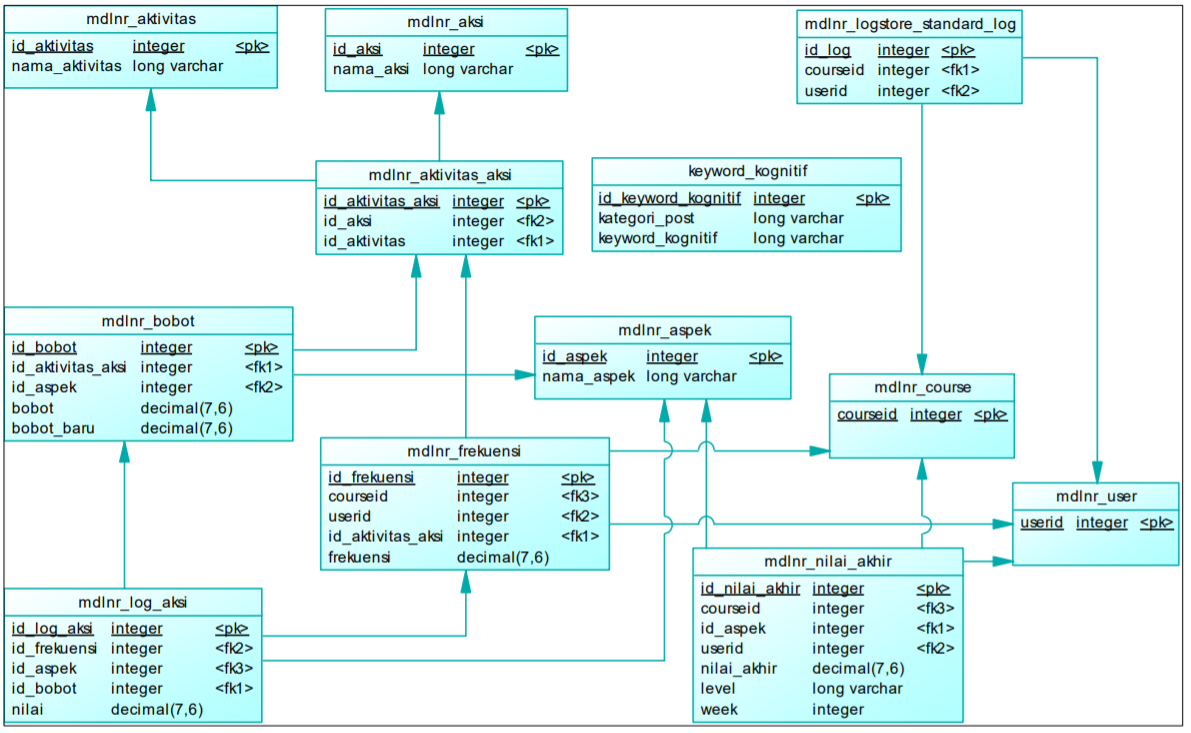
Tabel mdlnr\_course adalah tabel bawaan dari Moodle yang menyimpan course sebagai mata kuliah. Tabel ini memiliki atribut id\_course sebagai primary key, category, sortorder, fullname, shortname, idnumber, summary, summaryformat, format, showgrades, newsitems, startdate, enddate, marker, maxbytes, legacyfiles, showreports, visible, visibleold, groupmode, groupmodeforce, defaultgroupingid, lang, calendartype, theme, timecreated, timemodified, requested, enablecompletion, completionnotify, dan cacherev.



Gambar 3.12 Class Diagram



Gambar 3.13 Conceptual Data Model



Gambar 3.14 Physical Data Model

***(Halaman ini sengaja dikosongkan)***

#### Tabel mdlnr\_user

Tabel mdlnr\_user adalah tabel bawaan dari Moodle yang menyimpan pengguna di dalam sistem *e-learning* diantaranya yaitu admin, pengajar, dan pelajar. Tabel ini memiliki atribut id\_user sebagai primary key, auth, confirmed, policyagreed, deleted, suspended, mnethostid, username, password, idnumber, firstname, lastname, email, emailstop, icq, skype, yahoo, aim, msn, phone1, phone2, institution, department, address, city, country, lang, calendartype, theme, timezone, firstaccess, lastaccess, lastlogin, currentlogin, lastip, secret, picture, url, description, descriptionformat, mailformat, maildigest, maildisplay, autosubscribe, trachforums, timecreated, timemodified, trustbitmask, imagealt, lastnamephonetic, middlename dan altername.

#### Tabel mdlnr\_logstore\_standard\_log

Tabel mdlnr\_logstore\_standard\_log adalah tabel bawaan dari Moodle yang menyimpan berbagai tindakan pelajar pada aktivitas *e-learning* berdasarkan *course* tertentu, seperti melihat forum, melakukan chat, mengumpulkan *assignment* dan berbagai aktivitas lainnya. Tabel ini memiliki atribut id\_log sebagai *primary key*, eventname, component, action, target, objecttable, objectid, crud, edulevel, contexed, contextlevel, contexinstaceid, userid diambil dari table mdlnr\_user, courseid diambil dari table mdlnr\_course, relateduserid, anonymous, other, timecreated, origin, ip, dan relateduserid.

#### Tabel mdlnr\_aktivitas

Tabel mdlnr\_aktivitas berisi aktivitas dalam *e-learning.* Detail atribut tabel mdlnr\_aktivitas dijelaskan pada Tabel 3.19

Tabel 3.19 Tabel mdlnr\_aktivitas

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_aktivitas | Integer | *Primary key* dari tabel mdlnr\_aktivitas. |
| nama\_aktivitas | Text | Aktivitas dalam *e-learning* yang terdiri dari *assignment, chat, course, forum, glossary, quiz,* dan *wiki.* |

#### Tabel mdlnr\_aksi

Tabel mdlnr\_aksi berisi daftar tindakan yang dilakukan oleh pelajar dalam aktivitas yang terdapat pada *e-learning*. Detail atribut tabel mdlnr\_aksi dijelaskan pada Tabel 3.20.

Tabel 3.20 Tabel mdlnr\_aksi

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_aksi | Integer | *Primary key* dari tabel mdlnr\_aksi. |
| nama\_aksi | Text | Tindakan pelajar dalam aktivitas *e-learning* yang terdiri dari *view, submit, create, chat, readlog, viewdiscussion, startdiscussion, replypost, deleteownpost, allowforcesubscribe, write, attemp, reviewmyattemps, viewpage, createpage, editpage, participate,* dan *viewcomment*. |

#### Tabel mdlnr\_aktivitas\_aksi

Tabel mdlnr\_aktivitas\_aksi merupakan table pivot dari tabel mdlnr\_aktivitas dan tabel mdlnr\_aksi. Detail atribut tabel mdlnr\_aktivitas\_aksi dijelaskan pada Tabel 3.21.

Tabel 3.21 Tabel mdlnr\_aktivitas\_aksi

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_aktivitas\_aksi | Integer | *Primary key* dari tabel mdlnr\_aktivitas\_aksi. |
| id\_aktivitas | Integer | *Foreign key* identitas aktivitas dari aktivitas\_aksi |
| id\_aksi | Integer | *Foreign key* identitas aksi dari aktivitas\_aksi |

#### Tabel mdlnr\_aspek

Tabel mdlnr\_aspek berisi aspek yang ada di dalam *meaningful learning.* Detail atribut mdlnr\_aspek dijelaskan pada Tabel 3.22.

Tabel 3.22 Tabel mdlnr\_aspek

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_aspek | Integer | *Primary key* dari tabel mdlnr\_aspek. |
| nama\_aspek | Text | Aspek di dalam *meaningful learning* yang terdiri dari aspek aktif, konstruktif, intensional, autentik, dan kooperatif. |

#### Tabel mdlnr\_bobot

Tabel mdlnr\_bobot berisi bobot dari tindakan pelajar terhadap aktivitas *e-learning* untuk setiap aspek. Detail atribut tabel mdlnr\_bobot dijelaskan pada Tabel 3.23.

Tabel 3.23 Tabel mdlnr\_bobot

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_bobot | Integer | *Primary key* dari tabel mdlnr\_bobot. |
| id\_aktivitas\_aksi | Integer | *Foreign key* identitas aktivitas\_aksi dari bobot. |
| id\_aspek | Integer | *Foreign key* identitas aspek dari bobot. |
| bobot | Decimal (7,6) | Bobot tindakan pelajar terhadap aktivitas e*-learning* pada tiap aspek. |
| bobot\_baru | Decimal (7,6) | Menyimpan bobot jika ada aktivitas yang tidak terpakai pada *course* tertentu. Bobot baru diperoleh dari persamaan 3.5. |

#### Tabel mdlnr\_frekuensi

Tabel mdlnr\_frekuensi digunakan untuk menyimpan *f(t)* atau frekuensi tindakan pelajar terhadap aktivitas *e-learning* pada tiap *course*. Detail atribut tabel mdlnr\_frekuensi dijelaskan pada Tabel 3.24.

Tabel 3.24 Tabel mdlnr\_frekuensi

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_frekuensi | Integer | *Primary key* dari tabel mdlnr\_frekuensi. |
| id\_aktivitas\_aksi | Integer | *Foreign key* identitas aktivitas\_aksi dari frekuensi. |
| id\_user | Integer | *Foreign key* identitas *user* dari frekuensi. |
| id\_course | Integer | *Foreign key* identitas *course* dari frekuensi. |
| frekuensi | Decimal (7,6) | Menyimpan f(t) yang telah dihitung dengan menggunakan persamaan 3.1, persamaan 3.2, dan persamaan 3.3. Nilai fungsi f(t) dianggap NULL (atau bernilai -1) jika ada aktivitas yang tidak terpakai pada *course* tertentu. |

#### Tabel mdlnr\_log\_aksi

Tabel mdlnr\_log\_aksi menyimpan nilai dari tindakan pelajar terhadap aktivitas *e-learning* untuk tiap aspek pada *course* tertentu. Detail atribut tabel mdlnr\_log\_aksi dijelaskan pada Tabel 3.25

Tabel 3.25 Tabel mdlnr\_log\_aksi

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_log\_aksi | Integer | *Primary key* dari tabel mdlnr\_log\_aksi. |
| id\_bobot | Integer | *Foreign key* identitas bobot dari log\_aksi. |
| id\_aspek | Integer | *Foreign key* identitas aspek dari log\_aksi. |
| nilai | Decimal (7,6) | Menyimpan nilai yang telah dihitung dengan persamaan 3.6. |

#### Tabel mdlnr\_nilai\_akhir

Tabel mdlnr\_nilai\_akhir menyimpan total nilai tiap aspek *meaningful learning* pelajar pada *course* tertentu. Detail atribut tabel mdlnr\_nilai\_akhir dijelaskan pada Tabel 3.26.

Tabel 3.26 Tabel mdlnr\_nilai\_akhir

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_nilai\_akhir | Integer | *Primary key* dari tabel mdlnr\_nilai\_akhir. |
| id\_user | Integer | *Foreign key* identitas *user* dari nilai\_akhir. |
| id\_course | Integer | *Foreign key* identitas *course* dari nilai\_akhir. |
| nilai\_akhir | Decimal (7,6) | Total nilai tiap aspek *meaningful learning* pelajar pada *course* tertentu. |
| level | Text | Tingkat keterlibatan pelajar pada masing-masing aspek *meaningful learning* yang diperoleh dengan persamaan 3.8. |
| week | Integer | Menyimpan nilai aspek pelajar berdasarkan tren waktu tiap minggu. |

#### Tabel mdlnr\_keyword\_kognitif

Tabel mdlnr\_keyword\_kognitif berisi kamus *keyword* kognitif tinggi berdasarkan POS *Tagging*. Detail atribut tabel mdlnr\_keyword\_kognitif dijelaskan pada Tabel 3.27

Tabel 3.27 Tabel mdlnr\_keyword\_kognitif

| **Nama Kolom** | **Tipe Data** | **Keterangan** |
| --- | --- | --- |
| id\_keyword\_kognitif | Integer | *Primary key* dari tabel mdlnr\_keyword\_kognitif. |
| kategori\_post | Text | Kategori POS *Tagging* pada kamus *keyword* kognitif tinggi yang terdiri dari good\_RB, good\_MD, good\_JJ, good\_VB, good\_NN dan good\_IN. |
| keyword\_kognitif | Text | Isi kamus *keyword* kognitif tinggi berdasarkan kategori POS *Tagging*. |

### Perancangan Antarmuka Pengguna

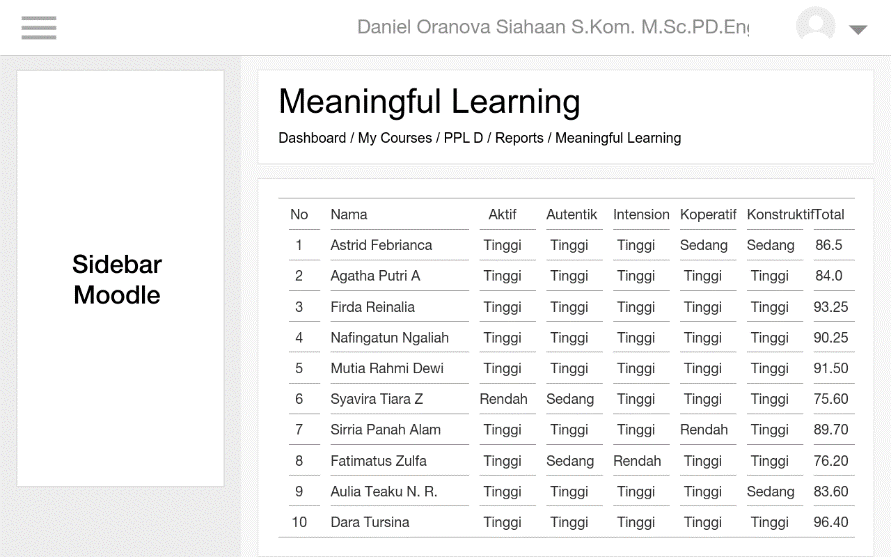
Perancangan antarmuka pengguna merupakan hal yang penting dalam melakukan perancangan perangkat lunak. Antarmuka pengguna yang berhubungan langsung dengan aktor harus mudah digunakan tanpa harus menggunakan petunjuk apapun. Perancangan antarmuka pembangunan *plugin report* untuk menghitung *meaningful learning* pelajar ini hanya memiliki satu hak akses, yaitu pengajar

#### Rancangan Halaman Antarmuka Memonitor Tingkat Meaningful Learning Pelajar

Halaman ini digunakan oleh pengajar untuk memonitor tingkat *meaningful learning* dari pelajar yang terdaftar pada mata kuliah pengajar. Nilai *meaningful learning* terdiri dari 3 level, yaitu tinggi, sedang, dan rendah. Pada halaman ini juga menampilkan total nilai rata-rata dari seluruh aspek untuk setiap pelajar. Rancangan halaman dapat dilihat pada Gambar 3.15. Atribut antarmuka dapat dilihat pada Tabel 3.28

Tabel 3.28 Atribut Antarmuka Memonitor Tingkat Meaningful Learning Pelajar

| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| --- | --- | --- | --- | --- |
| 1 | *listPelajar* | *Table* | Berisi daftar pelajar yang tergabung pada mata kuliah yang dipilih. | *Array* |
| 2 | *nilaiAktif* | *Text view* | Berisi informasi nilai aktif | *String* |
| 3 | *nilaiAutentik* | *Text view* | Berisi informasi nilai autentik | *String* |
| 4 | *nilaiIntensional* | *Text view* | Berisi informasi nilai intensional | *String* |
| 5 | *nilaiKoperatif* | *Text view* | Berisi informasi nilai koperatif | *String* |
| 6 | *nilaiKonstruktif* | *Text*  *View* | Berisi informasi nilai konstruktif | *String* |
| 7 | *totalAspek* | *Text*  *View* | Berisi informasi nilai rata-rata dari seluruh aspek | *String* |



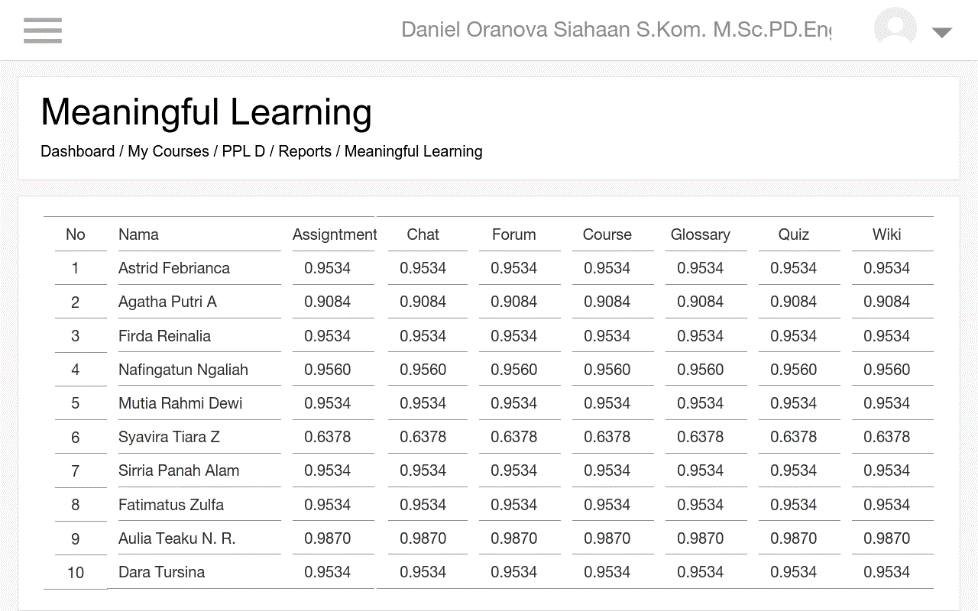
Gambar 3.15 Rancangan Halaman Antarmuka Memonitor Tingkat Meaningful Learning Pelajar

#### Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Kooperatif

Halaman ini digunakan oleh pengajar untuk melihat detail nilai yang didapatkan dari tiap aspek. Dalam satu aspek, nilai diambil dari beberapa aktivitas pada *course*, seperti *Assignment, Chat, Forum, Course, Glossary, Quiz, dan Wiki*. Rancangan halaman dapat dilihat pada Gambar 3.16. Atribut antarmuka dapat dilihat pada Tabel 3.29.

Tabel 3.29 Atribut Antarmuka Melihat Detail Nilai Aspek Kooperatif

| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| --- | --- | --- | --- | --- |
| 1 | *listDetail* | *Table* | Berisi daftar pelajar beserta nilai dari setiap aktivitas | *Array* |

****

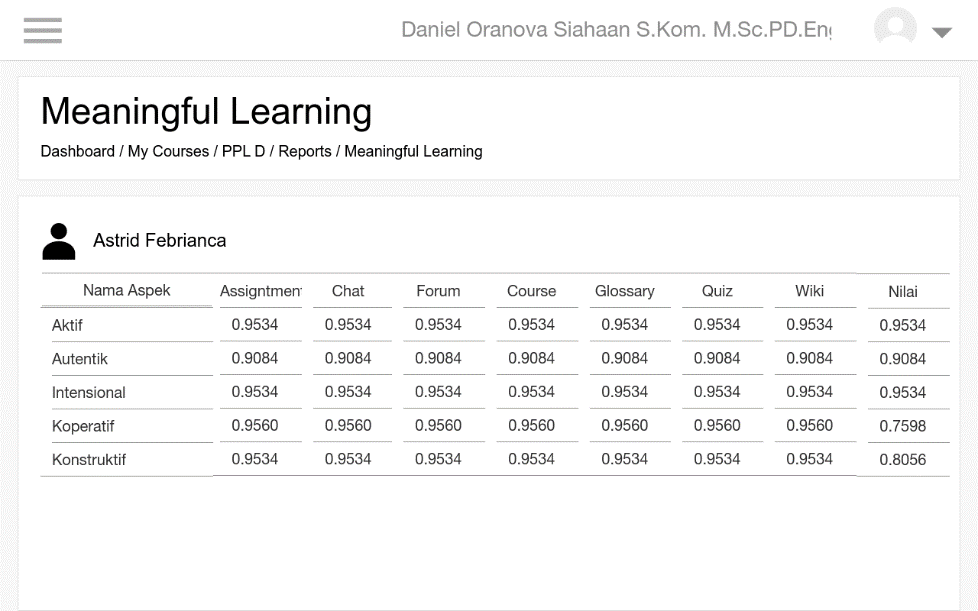
Gambar 3.16 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Kooperatif

#### Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar

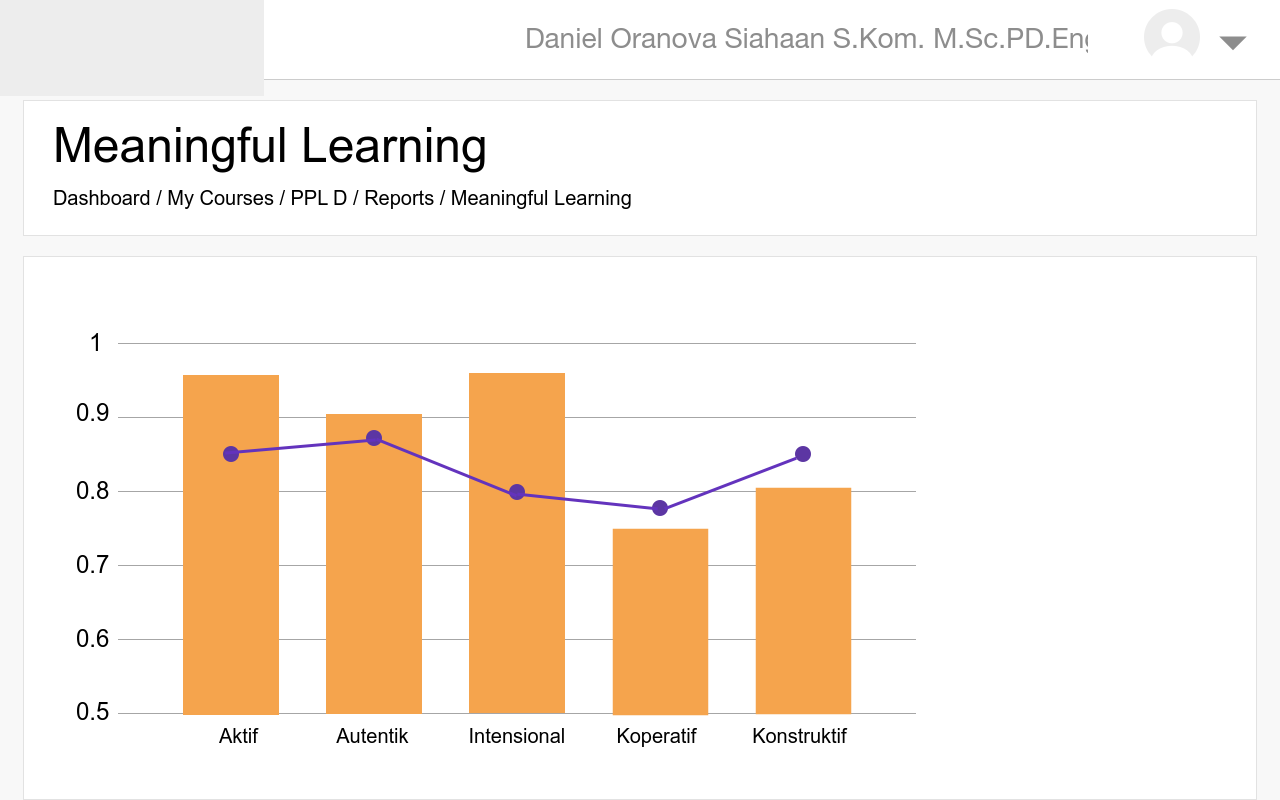
Halaman ini digunakan untuk melihat detail nilai yang didapatkan dari tiap pelajar. Halaman ini berisi detail nilai yang ditampilkan berdasarkan aspek *meaningful learning*, yaitu aktif, konstruktif, intensional, autentik, dan kooperatif, beserta nilai dari setiap aktivitas yang menunjang penilaian aspek tersebut, seperti *assignment, chat,* forum*, course, glossary, quiz, dan* wiki. Pada halaman ini juga menampilkan grafik perbandingan nilai pelajar pada minggu terakhir dengan sebelumnya dan perbandingan nilai pelajar dengan nilai rata-rata mata kuliah yang diampu. Rancangan halaman dapat dilihat pada Gambar 3.17, Gambar 3.18, dan Gambar 3.19. Atribut antarmuka dapat dilihat pada Tabel 3.30.

Tabel 3.30 Atribut Antarmuka Aktivitas Mahasiswa

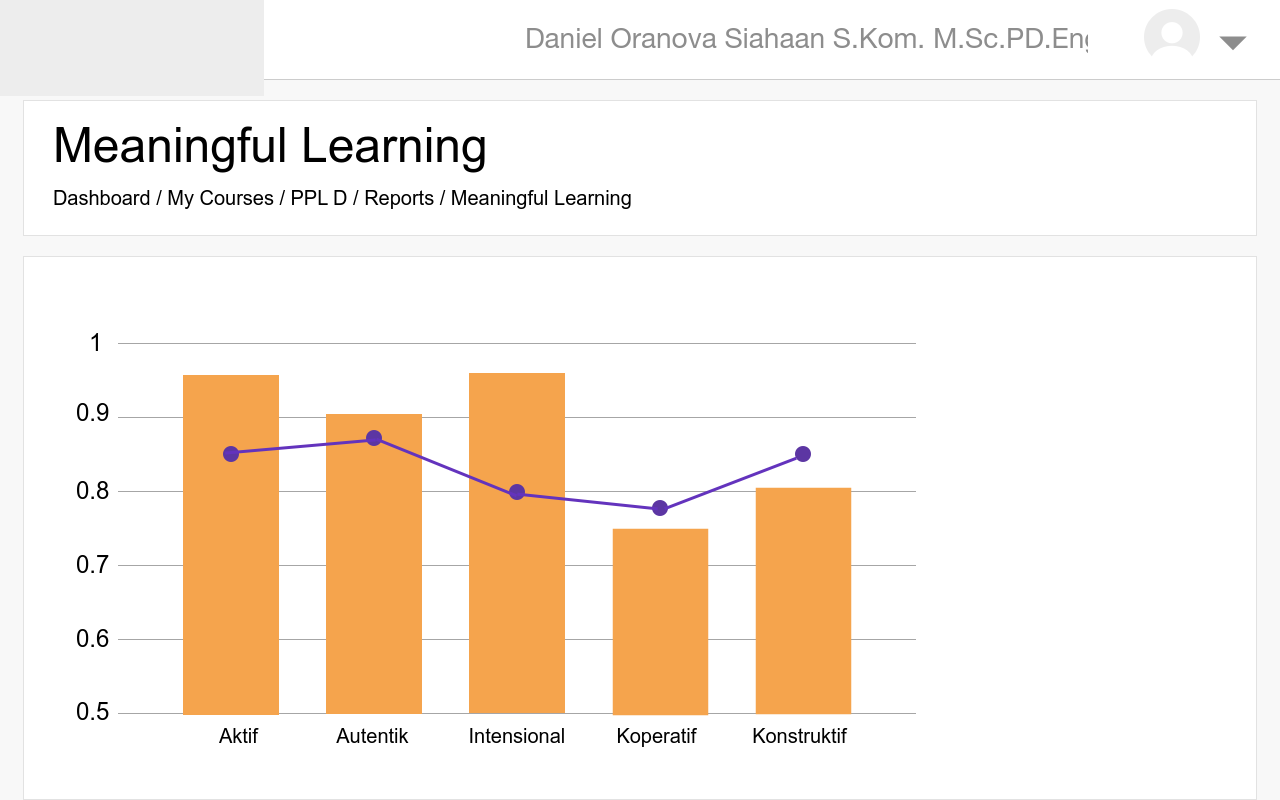
| No. | Nama Atribut Antarmuka | Jenis Atribut | Kegunaan | Masukan/ Keluaran |
| --- | --- | --- | --- | --- |
| 1 | *listAspek* | *Table* | Berisi daftar aspek beserta nilai dari setiap aktivitas | *Array* |
| 2 | *chartNilai* | *Chart* | Berisi grafik perbandingan antara nilai pelajar yang terakhir dengan nilai rata-rata mata kuliah yang diampu | *Chart* |
| 2 | *chartWeek* | *Chart* | Berisi grafik perbandingan antara nilai pelajar yang terakhir dengan sebelumnya | *Chart* |



Gambar 3.17 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar



Gambar 3.18 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar



Gambar 3.19 Rancangan Halaman Antarmuka Melihat Detail Nilai Aspek Pelajar

# BAB IV IMPLEMENTASI



## Lingkungan Implementasi

*Plugin* ini diimplementasikan dengan menggunakan kakas bantu:

1. Moodle versi 3.3.8.

2. Bahasa pemrograman PHP.

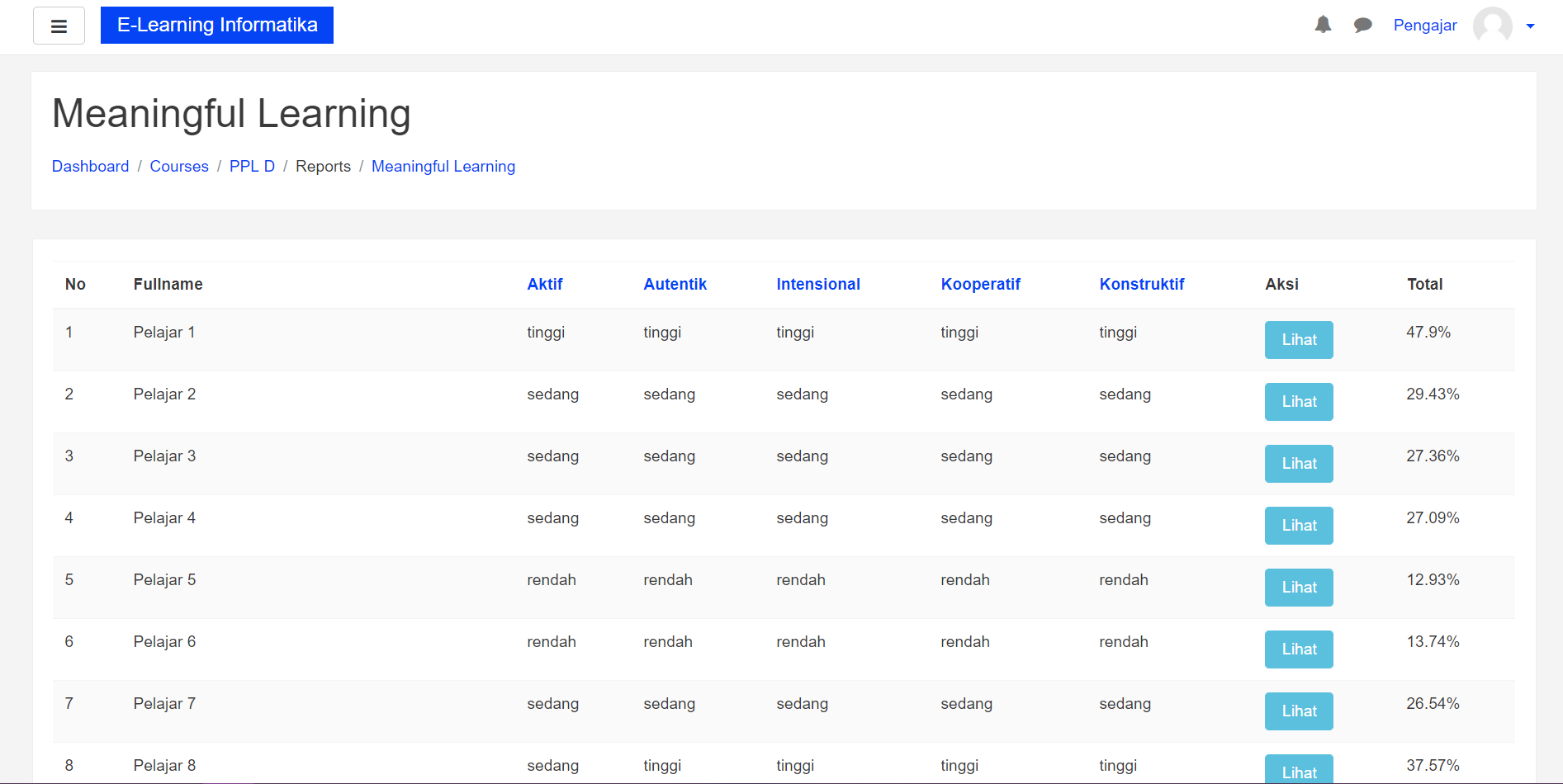
3. *Database* MySQL.

## Implementasi Tampilan Antarmuka

Subbab ini membahas tentang implementasi tampilan antarmuka yang telah dirancang dan dibahas pada Bab III. Selanjutnya akan dirinci berdasarkan halaman yang akan tampil dan dilihat oleh pengajar.

### Implementasi Halaman Monitor Tingkat *Meaningful Learning*

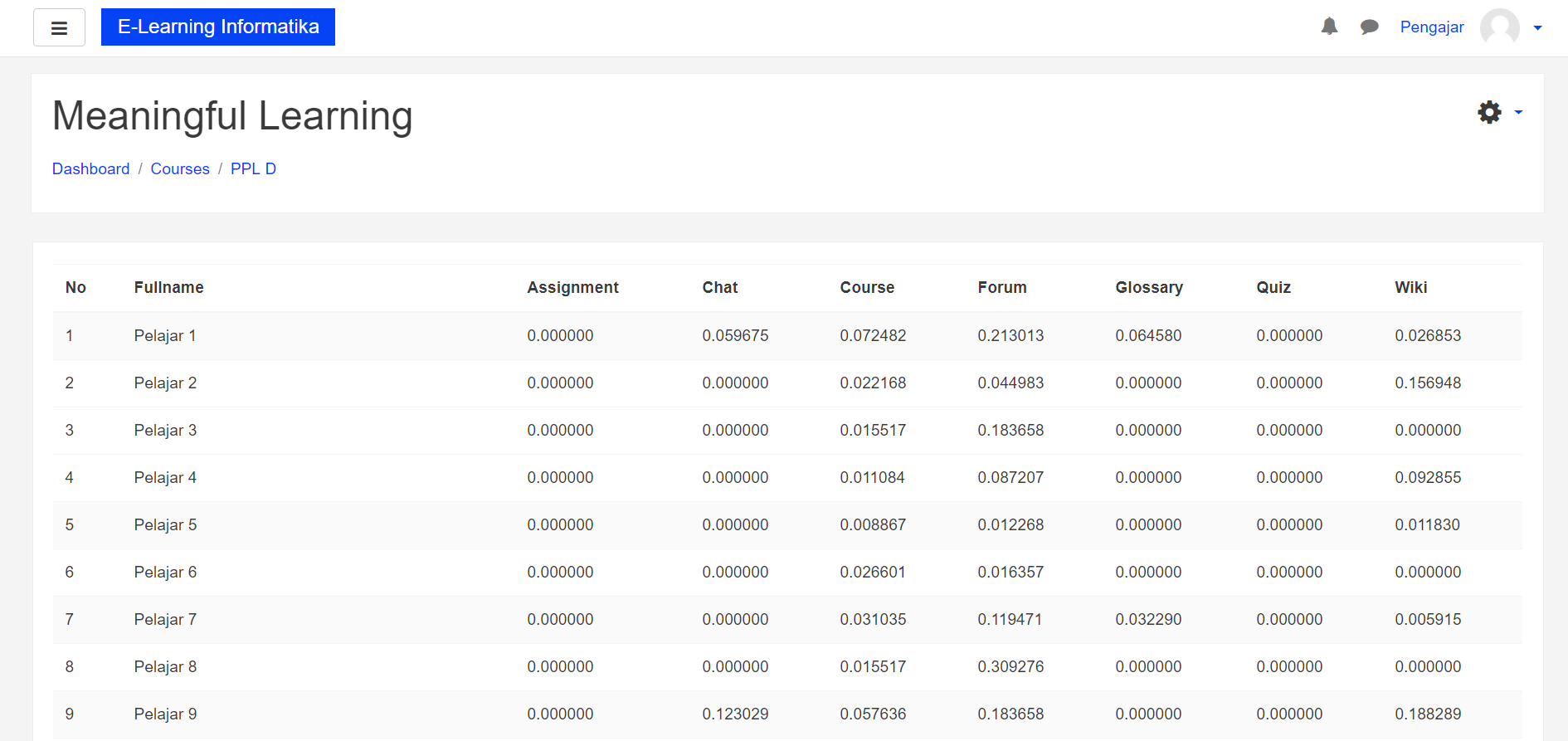
Halaman antarmuka monitor tingkat *meaningful learning* pada Gambar 4.1 merupakan halaman yang digunakan oleh pengajar untuk memonitor tingkat *meaningful learning* dari masing-masing pelajar dalam satu kelas yang dipilih. Terdapat tombol pada setiap aspek dan pada total nilai untuk melihat detail dari nilai yang dipilih. Halaman ini diimplementasikan pada berkas index.php yang dapat dilihat pada Kode Sumber 4.1



Gambar 4.1 Implementasi Monitor Tingkat Meaningful Learning

### Implementasi Halaman Melihat Detail Nilai Aspek Kooperatif

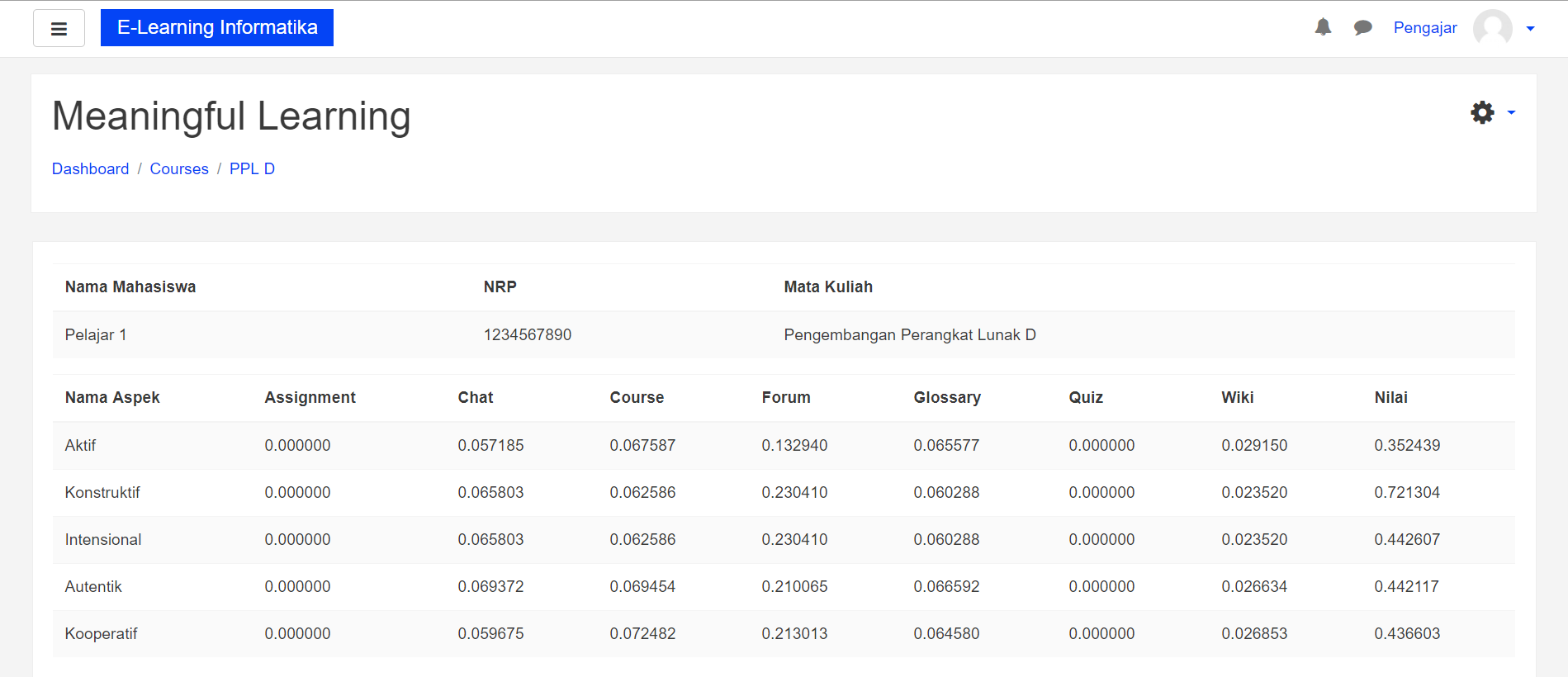
Halaman antarmuka melihat detail nilai aspek kooperatif pada Gambar 4.2 merupakan halaman yang akan ditampilkan pada plugin ketika pengajar menekan tombol dari salah satu aspek pada halaman monitor tingkat *meaningful learning*. Halaman ini diimplementasikan pada berkas aspek\_koperatif.php yang dapat dilihat pada Kode Sumber 4.2



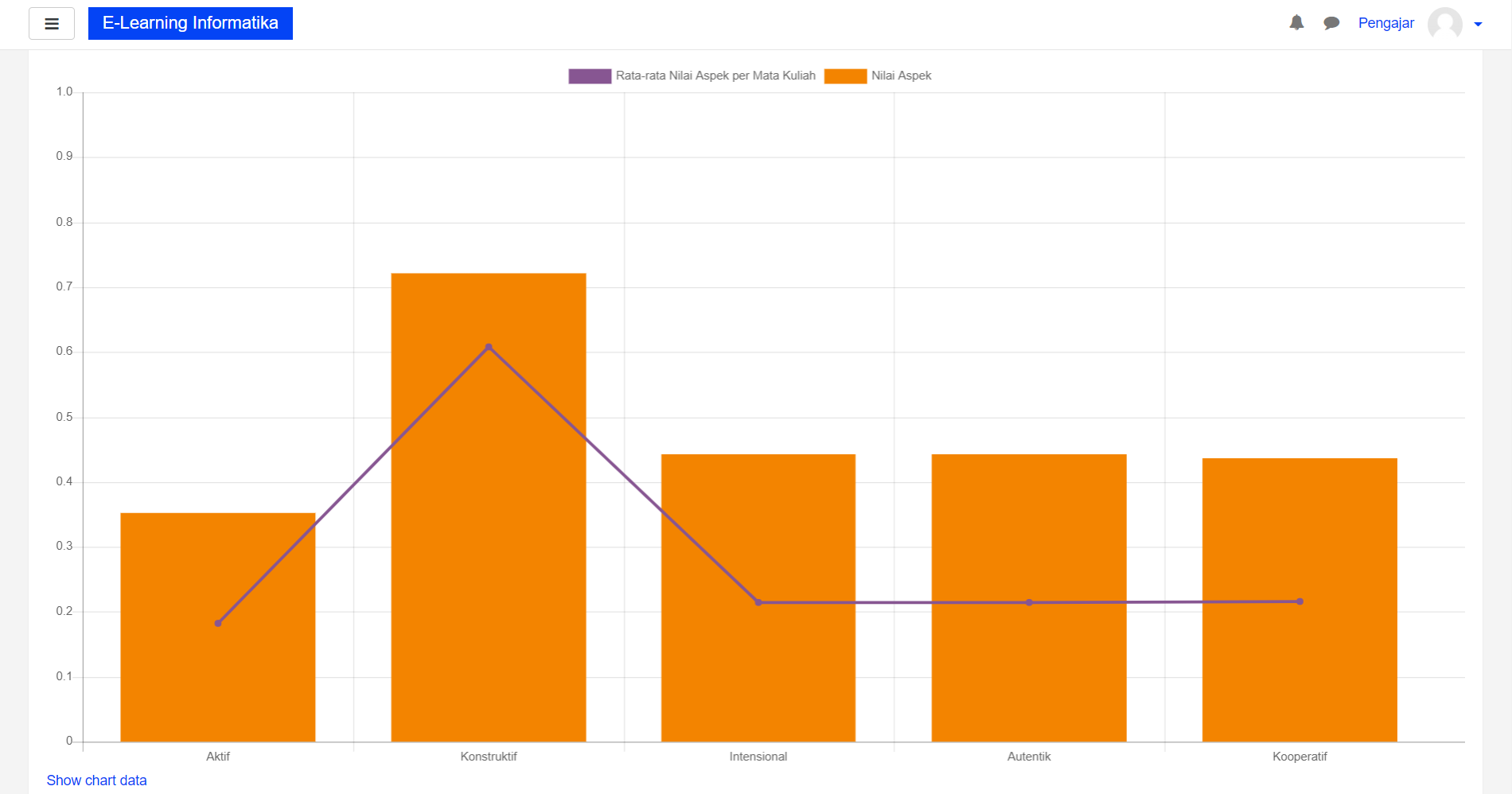
Gambar 4.2 Implementasi Halaman Melihat Detail Nilai Aspek Kooperatif

### Implementasi Halaman Melihat Detail Nilai Setiap Pelajar

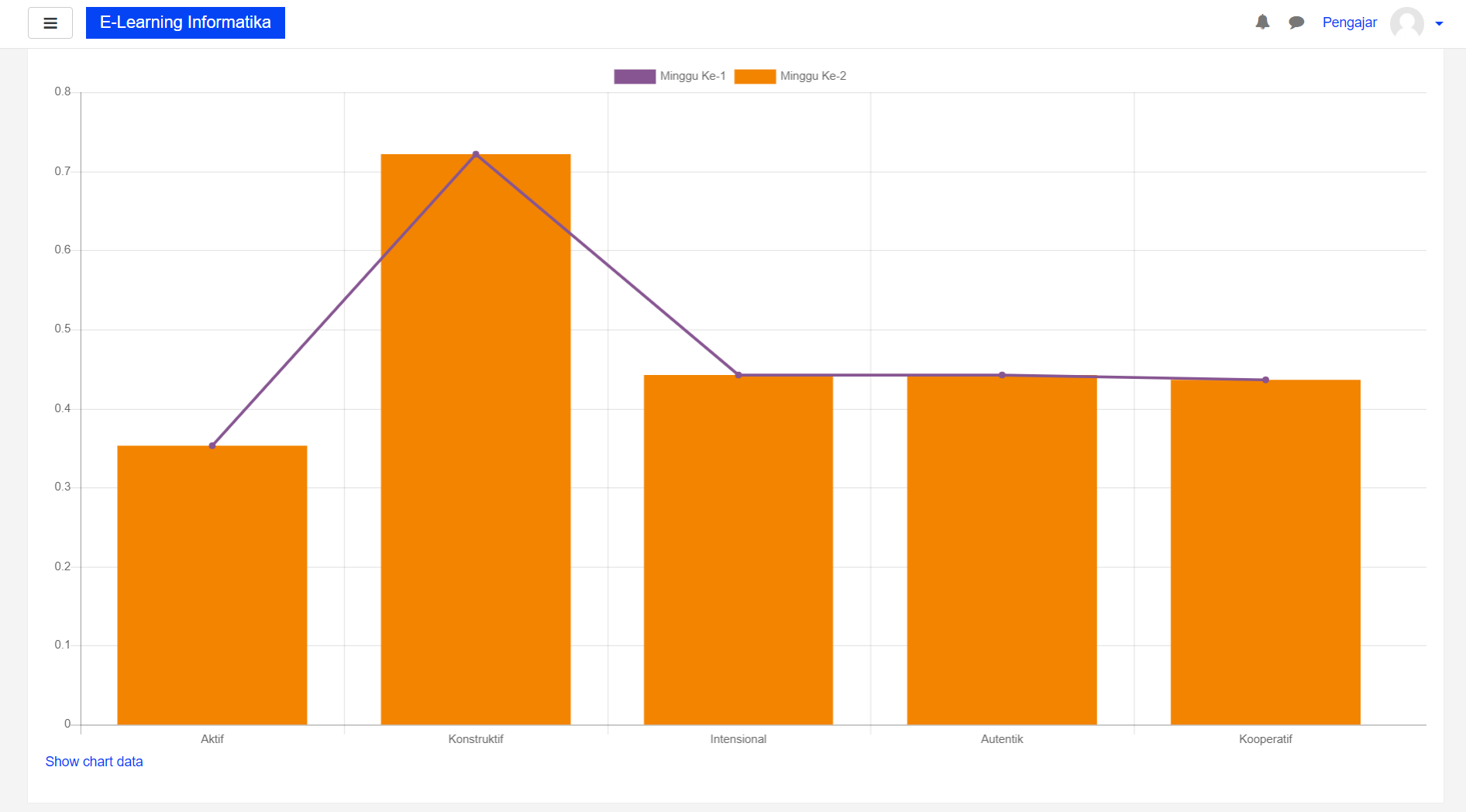
Halaman antarmuka melihat detail nilai setiap pelajar pada Gambar 4.3, dan Gambar 4.5 merupakan halaman yang digunakan oleh pengajar untuk memonitor nilai setiap aspek *meaningful learning* sesuai pelajar yang dipilih. Terdapat grafik yang menunjukkan perbandingan antara nilai aspek yang diperoleh pelajar dengan rata-rata nilai setiap aspek dari mata kuliah yang dipilih pada Gambar 4.4 dan terdapat grafik yang menunjukkan perbandingan antara nilai aspek yang diperoleh pelajar pada 2 minggu terakhir yang ditunjukkan pada Gambar 4.5. Halaman ini diimplementasikan pada berkas detail\_student.php yang dapat dilihat pada Kode Sumber 4.3



Gambar 4.3 Implementasi Halaman Detail Nilai setiap Pelajar



Gambar 4.4 Implementasi Halaman Detail Nilai setiap Pelajar



Gambar 4.5 Implementasi Halaman Detail Nilai setiap Pelajar

## Implementasi Kasus Penggunaan

Implementasi kasus penggunaan menggunakan Bahasa PHP dari Moodle. Pada subbab ini akan menjelaskan dan menampilkan kode yang digunakan dalam pembuatan *plugin report* pada beberapa kasus penggunaan.

### Implementasi Kasus Penggunaan Monitor Tingkat *Meaningful Learning*

|  |
| --- |
| 1. <?php 2. include('../../vendor/autoload.php'); 3. require\_once('../../config.php'); 4. require\_once($CFG->dirroot.'/course/lib.php'); 5. require\_once($CFG->libdir.'/adminlib.php'); 6. global $PAGE, $COURSE, $USER, $OUTPUT, $remotedb, $CFG; 7. $strtitle = get\_string('pluginname', 'report\_meaningfullearning'); 8. $courseid = optional\_param('id', 0, PARAM\_INT); 9. if (empty($courseid)) { 10. require\_login(); 11. $context = context\_system::instance(); 12. $coursename = format\_string($SITE->fullname, true, array('context' => $context)); 13. } else { 14. $course = get\_course($courseid); 15. require\_login($course); 16. $context = context\_course::instance($course->id); 17. $coursename = format\_string($course->fullname, true, array('context' => $context)); 18. } 19. require\_capability('report/meaningfullearning:view', $context); 20. $params = array(); 21. if ($courseid != 0) { 22. $params['id'] = $courseid; 23. } 24. $url = new moodle\_url('/report/meaningfullearning/index.php', $params); 25. $PAGE->navigation; 26. $PAGE->set\_url($url); 27. $PAGE->set\_context($context); 28. $PAGE->set\_title($strtitle); 29. $PAGE->set\_pagelayout('report'); 30. $PAGE->set\_heading($strtitle); 31. $sql = "SELECT mu.id, mu.username, mu.firstname 32. FROM (mdlnr\_course\_completions mc, mdlnr\_user mu, mdlnr\_course md) 33. WHERE mc.course = '$courseid' 34. AND mc.userid = mu.id 35. AND md.id = mc.course"; 36. $list\_pelajar = $DB->get\_records\_sql($sql); 37. $link\_aktif = '<a href="/report/meaningfullearning/aspek\_aktif.php?'.'&id='.$courseid.'">Aktif</a>'; 38. $link\_autentik = '<a href="/report/meaningfullearning/aspek\_autentik.php?'.'&id='.$courseid.'">Autentik</a>'; 39. $link\_intensional = '<a href="/report/meaningfullearning/aspek\_intensional.php?'.'&id='.$courseid.'">Intensional</a>'; 40. $link\_kooperatif = '<a href="/report/meaningfullearning/aspek\_koperatif.php?'.'&id='.$courseid.'">Kooperatif</a>'; 41. $link\_konstruktif = '<a href="/report/meaningfullearning/aspek\_konstruktif.php?'.'&id='.$courseid.'">Konstruktif</a>'; 42. $table = new html\_table(); 43. $table->head = array('No', 'Fullname', $link\_aktif, $link\_konstruktif, $link\_intensional, $link\_autentik, $link\_kooperatif, 'Aksi', 'Total'); 44. $key = '0'; 45. foreach ($list\_pelajar as $value) { 46. $total = 0; 47. $nama = $value->firstname; 48. $link\_detail = '<a href="/report/meaningfullearning/detail\_student.php?id='.$value->id.'&courseid='.$courseid.'" class="btn btn-info">Lihat</a>'; 49. ......... 50. //nilai aspek kooperatif 51. $sql = "SELECT mdlnr\_nilai\_akhir.id, mdlnr\_nilai\_akhir.level, mdlnr\_nilai\_akhir.nilai\_akhir FROM mdlnr\_nilai\_akhir 52. WHERE mdlnr\_nilai\_akhir.id =(SELECT MAX(id) FROM mdlnr\_nilai\_akhir where mdlnr\_nilai\_akhir.id\_aspek='5'and 53. mdlnr\_nilai\_akhir.id\_course ='$courseid' and mdlnr\_nilai\_akhir.id\_user ='$value->id')"; 54. $koo\_records = $DB->get\_records\_sql($sql); 55. foreach ($koo\_records as $value5) { 56. $nilai\_kooperatif = $value5->level; 57. $total = $total + $value5->nilai\_akhir; 58. } 59. //total rata-rata dari semua aspek 60. $total = ($total / 5) \* 100; 61. $total = round($total, 2); 62. $table->data[] = array($key+1, $nama, $nilai\_aktif, $nilai\_autentik, $nilai\_intensional, $nilai\_kooperatif, $nilai\_konstruktif, $link\_detail, $total."%"); 63. $key++; 64. } 65. echo $OUTPUT->header(); 66. $PAGE->set\_heading($strtitle); 67. if(!empty($table->data)) { 68. echo html\_writer::table($table); 69. } 70. echo $OUTPUT->footer(); |

Kode Sumber 4.1 Implementasi Kasus Penggunaan Monitor Tingkat Meaningful Learning

Tabel 4.1 Penjelasan Kode Sumber 4.1

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 2 | Menyertakan file dari folder vendor pada Moodle |
| 3 | Menyertakan file konfigurasi dari Moodle |
| 4 | Menampilkan *plugin* ‘meaningful learning’ pada menu *report* dalam *setting* *course* |
| 5 | Menampilkan *plugin* ‘meaningful learning’ pada menu *report* dalam *site administration* |
| 6 | Menginisialisasi variabel $PAGE, $COURSE, $USER, $OUTPUT, $remotedb, $CFG dari Moodle |
| 7 | Mendapatkan nama *plugin* yang sudah diinisialisasi pada file report\_meaningfullearning.php dalam folder lang/en |
| 8 | Menginisialisasi variabel courseid dengan tipe data integer |
| 9-18 | Mengecek ada tidaknya variabel courseid |
| 19 | Mengecek kapabilitas role yang dapat mengakses *plugin* |
| 20-23 | Menginisialisasi variabel params dengan courseid yang telah didapat |
| 24 | Menginisialisasi alamat *url* pada Moodle |
| 25-30 | Mengonfigurasi *navigation*, *url*, *context, title, page layout, heading* yang akan ditampilkan pada halaman *plugin* |
| 31-36 | Mendapatkan daftar pelajar yang tergabung dalam mata kuliah yang dipilih |
| 37-41 | Menuju ke halaman aspek tertentu |
| 42-64 | Mendapatkan nilai aspek kooperatif beserta total nilai rata-rata dari seluruh aspek untuk seluruh pelajar |
| 65-66 | Menampilkan tampilan *header* dari Moodle |
| 67-69 | Menampilkan tabel yang berisi daftar pelajar beserta nilai tiap aspek *meaningful learning* dan total rata-rata dari semua aspek |
| 70 | Menampilkan tampilan *footer* dari Moodle |

### Implementasi Kasus Penggunaan Melihat Detail Nilai Aspek Kooperatif

|  |
| --- |
| 1. <?php 2. include('../../vendor/autoload.php'); 3. require\_once('../../config.php'); 4. require\_once($CFG->dirroot.'/course/lib.php'); 5. require\_once($CFG->libdir.'/adminlib.php'); 6. global $PAGE, $COURSE, $USER, $OUTPUT, $remotedb, $CFG; 7. $strtitle = get\_string('pluginname', 'report\_meaningfullearning'); 8. $courseid = optional\_param('id', 0, PARAM\_INT); 9. if (empty($courseid)) { 10. require\_login(); 11. $context = context\_system::instance(); 12. $coursename = format\_string($SITE->fullname, true, array('context' => $context)); 13. } else { 14. $course = get\_course($courseid); 15. require\_login($course); 16. $context = context\_course::instance($course->id); 17. $coursename = format\_string($course->fullname, true, array('context' => $context)); 18. } 19. require\_capability('report/meaningfullearning:view', $context); 20. $url =new moodle\_url('/report/meaningfullearning/aspek\_kooperatif.php', ['id' => $courseid]); 21. $PAGE->navigation; 22. $PAGE->set\_url($url); 23. $PAGE->set\_context($context); 24. $PAGE->set\_title($strtitle); 25. $PAGE->set\_pagelayout('report'); 26. $PAGE->set\_heading($strtitle); 27. //query daftar pelajar 28. $sql = "SELECT mu.id, mu.username, mu.firstname 29. FROM (mdlnr\_course\_completions mc, mdlnr\_user mu, mdlnr\_course md) 30. WHERE mc.course = '$courseid' 31. AND mc.userid = mu.id 32. AND md.id = mc.course"; 33. $list\_pelajar = $DB->get\_records\_sql($sql); 34. $table = new html\_table(); 35. $table->head = array('No', 'Fullname', 'Assignment', 'Chat', 'Course', 'Forum', 'Glossary', 'Quiz', 'Wiki'); 36. $key = '0'; 37. foreach ($list\_pelajar as $value) { 38. //query nilai tiap aktivitas 39. $sql = "SELECT mdlnr\_log\_aksi.id, SUM(mdlnr\_log\_aksi.nilai) as total, mdlnr\_log\_aksi.id, mdlnr\_log\_aksi.nilai, mdlnr\_log\_aksi.id\_aspek,mdlnr\_frekuensi.id\_aktivitas\_aksi, mdlnr\_aktivitas\_aksi.id\_aktivitas, mdlnr\_frekuensi.id\_user, mdlnr\_frekuensi.id\_course 40. FROM mdlnr\_log\_aksi 41. INNER JOIN mdlnr\_frekuensi ON mdlnr\_frekuensi.id = mdlnr\_log\_aksi.id\_frekuensi 42. INNER JOIN mdlnr\_aktivitas\_aksi ON mdlnr\_aktivitas\_aksi.id = mdlnr\_frekuensi.id\_aktivitas\_aksi 43. WHERE mdlnr\_frekuensi.id\_user = '$value->id' AND mdlnr\_frekuensi.id\_course = '$courseid' AND mdlnr\_log\_aksi.id\_aspek = 4 44. GROUP BY mdlnr\_aktivitas\_aksi.id\_aktivitas"; 45. $result = $DB->get\_records\_sql($sql); 46. $nama = $value->firstname; 47. foreach ($result as $value) { 48. if ($value->id\_aktivitas == 1) { 49. $assign = $value->total; 50. } elseif ($value->id\_aktivitas == 3) { 51. $chat = $value->total; 52. } elseif ($value->id\_aktivitas == 4) { 53. $course = $value->total; 54. } elseif ($value->id\_aktivitas == 5) { 55. $forum = $value->total; 56. } elseif ($value->id\_aktivitas == 6) { 57. $glossary = $value->total; 58. } elseif ($value->id\_aktivitas == 7) { 59. $quiz = $value->total; 60. } elseif ($value->id\_aktivitas == 8) { 61. $wiki = $value->total; 62. } 63. } 64. $table->data[] = array($key+1, $nama, $assign, $chat, $course, $forum, $glossary, $quiz, $wiki); 65. $key++; 66. } 67. echo $OUTPUT->header(); 68. $PAGE->set\_heading($strtitle); 69. if(!empty($table->data)) { 70. echo html\_writer::table($table); 71. } 72. echo $OUTPUT->footer(); |

Kode Sumber 4.2 Implementasi Kasus Penggunaan Melihat Detail Nilai Aspek Kooperatif

Tabel 4.2 Penjelasan Kode Sumber 4.2

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 2 | Menyertakan file dari folder vendor pada Moodle |
| 3 | Menyertakan file konfigurasi dari Moodle |
| 4 | Menampilkan *plugin* ‘meaningful learning’ pada menu *report* dalam *setting* *course* |
| 5 | Menampilkan *plugin* ‘meaningful learning’ pada menu *report* dalam *site administration* |
| 6 | Menginisialisasi variabel $PAGE, $COURSE, $USER, $OUTPUT, $remotedb, $CFG dari Moodle |
| 7 | Mendapatkan nama *plugin* yang sudah diinisialisasi pada file report\_meaningfullearning.php dalam folder lang/en |
| 8 | Menginisialisasi variabel courseid dengan tipe data integer |
| 9-18 | Mendapatkan variabel *context* |
| 19 | Mengecek kapabilitas role yang dapat mengakses *plugin* |
| 20 | Menginisialisasi alamat *url* pada Moodle |
| 21-26 | Mengonfigurasi *navigation*, *url*, *context, title, page layout, heading* yang akan ditampilkan pada halaman *plugin* |
| 27-33 | Mendapatkan daftar pelajar yang tergabung dalam mata kuliah yang dipilih |
| 34-35 | Membuat tabel baru |
| 36-66 | Mendapatkan nilai tiap aktivitas yang menunjang penilaian aspek kooperatif dalam *meaningful learning* untuk seluruh pelajar |
| 67-68 | Menampilkan tampilan *header* dari Moodle |
| 69-71 | Menampilkan tabel yang berisi daftar pelajar beserta nilai tiap aktivitas yang menunjang penilaian aspek autentik dalam *meaningful learning* |
| 72 | Menampilkan tampilan *footer* dari Moodle |

### Implementasi Kasus Penggunaan Melihat Detail Nilai setiap Pelajar

|  |
| --- |
| 1. <?php 2. include('../../vendor/autoload.php'); 3. require\_once('../../config.php'); 4. require\_once($CFG->dirroot.'/course/lib.php'); 5. require\_once($CFG->libdir.'/adminlib.php'); 6. global $PAGE, $COURSE, $USER, $OUTPUT, $remotedb, $CFG, $DB; 7. $strtitle = get\_string('pluginname', 'report\_meaningfullearning'); 8. $studentid = optional\_param('id', 0, PARAM\_INT); 9. $courseid = optional\_param('courseid', 0, PARAM\_INT); 10. if (empty($courseid)) { 11. require\_login(); 12. $context = context\_system::instance(); 13. $coursename = format\_string($SITE->fullname, true, array('context' => $context)); 14. } else { 15. $course = get\_course($courseid); 16. require\_login($course); 17. $context = context\_course::instance($course->id); 18. $coursename = format\_string($course->fullname, true, array('context' => $context)); 19. } 20. require\_capability('report/meaningfullearning:view', $context); 21. $url = new moodle\_url('/report/meaningfullearning/detail\_student.php', ['studentid' => $studentid], ['courseid' => $courseid]); 22. $PAGE->set\_url($url); 23. $PAGE->set\_context($context); 24. $PAGE->set\_title($strtitle); 25. $PAGE->set\_pagelayout('report'); 26. $PAGE->set\_heading($strtitle); 27. $PAGE->navigation; 28. ..... 29. //data pelajar 30. $sql = "SELECT \* 31. FROM (mdlnr\_course\_completions mc, mdlnr\_user mu, mdlnr\_course md) 32. WHERE mc.course = '$courseid' 33. AND mc.userid = mu.id 34. AND md.id = mc.course 35. AND mu.id = '$studentid'"; 36. $users = $DB->get\_records\_sql($sql); 37. $table = new html\_table(); 38. $table->head = array('Nama Mahasiswa';, 'NRP', 'Mata Kuliah'); 39. foreach ($users as $u2) { 40. $table->data[] = array($u2->firstname, $u2->username, $u2->fullname); 41. } 42. $table2 = new html\_table(); 43. $table2->head = array('Nama Aspek', 'Assignment', 'Chat', 'Course', 'Forum', 'Glossary', 'Quiz', 'Wiki', 'Nilai'); 44. $sql = "SELECT mdlnr\_nilai\_akhir.id, mdlnr\_nilai\_akhir.week FROM mdlnr\_nilai\_akhir WHERE mdlnr\_nilai\_akhir.id =(SELECT MAX(id) FROM mdlnr\_nilai\_akhir where mdlnr\_nilai\_akhir.id\_aspek='5' and mdlnr\_nilai\_akhir.id\_course='$courseid' and mdlnr\_nilai\_akhir.id\_user='$studentid')"; 45. $last\_week = $DB->get\_records\_sql($sql); 46. foreach ($last\_week as $value) { 47. $sql = "SELECT \* FROM mdlnr\_nilai\_akhir 48. WHERE mdlnr\_nilai\_akhir.week = '$value->week' AND mdlnr\_nilai\_akhir.id\_course='$courseid' AND mdlnr\_nilai\_akhir.id\_user='$studentid'"; 49. $nilai\_akhir = $DB->get\_records\_sql($sql); 50. foreach ($nilai\_akhir as $key => $value) { 51. if($value->id\_aspek == 1){ 52. $na\_aktif = $value->nilai\_akhir; 53. } elseif($value->id\_aspek == 2){ 54. $na\_konstruktif = $value->nilai\_akhir; 55. } elseif($value->id\_aspek == 3){ 56. $na\_intensional = $value->nilai\_akhir; 57. } elseif($value->id\_aspek == 4){ 58. $na\_autentik = $value->nilai\_akhir; 59. } elseif($value->id\_aspek == 5){ 60. $na\_kooperatif = $value->nilai\_akhir; 61. } 62. } 63. $week=$value->week; 64. $prev\_week = $week - 1; 65. if($prev\_week != 0){ 66. $sql = "SELECT \* FROM mdlnr\_nilai\_akhir 67. WHERE mdlnr\_nilai\_akhir.week = '$prev\_week' AND mdlnr\_nilai\_akhir.id\_course='$courseid' AND mdlnr\_nilai\_akhir.id\_user='$studentid'"; 68. $prev\_nilai\_akhir = $DB->get\_records\_sql($sql); 69. foreach ($prev\_nilai\_akhir as $key => $value) { 70. if($value->id\_aspek == 1){ 71. $na\_aktif\_prev = $value->nilai\_akhir; 72. } elseif($value->id\_aspek == 2){ 73. $na\_konstruktif\_prev = $value->nilai\_akhir; 74. } elseif($value->id\_aspek == 3){ 75. $na\_intensional\_prev = $value->nilai\_akhir; 76. } elseif($value->id\_aspek == 4){ 77. $na\_autentik\_prev = $value->nilai\_akhir; 78. } elseif($value->id\_aspek == 5){ 79. $na\_kooperatif\_prev = $value->nilai\_akhir; 80. } 81. } 82. } 83. //masukin ke chart 84. $sql = "SELECT avg(jmh\_all.total) as avg from (select sum(nilai\_akhir) as total from mdlnr\_nilai\_akhir 85. where id\_aspek='5' and id\_course='$courseid' and week='$week' group by id\_user) jmh\_all"; 86. $koo\_records = $DB->get\_records\_sql($sql); 87. foreach ($koo\_records as $value) { 88. $koo=$value->avg; 89. } 90. } 91. ..... 92. $sql = "SELECT mdlnr\_log\_aksi.id, SUM(mdlnr\_log\_aksi.nilai) as total, mdlnr\_log\_aksi.id, mdlnr\_log\_aksi.nilai, mdlnr\_log\_aksi.id\_aspek,mdlnr\_frekuensi.id\_aktivitas\_aksi, mdlnr\_aktivitas\_aksi.id\_aktivitas, mdlnr\_frekuensi.id\_user, mdlnr\_frekuensi.id\_course 93. FROM mdlnr\_log\_aksi 94. INNER JOIN mdlnr\_frekuensi ON mdlnr\_frekuensi.id = mdlnr\_log\_aksi.id\_frekuensi 95. INNER JOIN mdlnr\_aktivitas\_aksi ON mdlnr\_aktivitas\_aksi.id = mdlnr\_frekuensi.id\_aktivitas\_aksi 96. WHERE mdlnr\_frekuensi.id\_user = '$studentid' AND mdlnr\_frekuensi.id\_course = '$courseid' AND mdlnr\_log\_aksi.id\_aspek = 5 97. GROUP BY mdlnr\_aktivitas\_aksi.id\_aktivitas"; 98. $query\_aktif = $DB->get\_records\_sql($sql); 99. foreach ($query\_aktif as $value) { 100. if ($value->id\_aktivitas == 1) { 101. $assign = $value->total; 102. } elseif ($value->id\_aktivitas == 3) { 103. $chat = $value->total; 104. } elseif ($value->id\_aktivitas == 4) { 105. $course = $value->total; 106. } elseif ($value->id\_aktivitas == 5) { 107. $forum = $value->total; 108. } elseif ($value->id\_aktivitas == 6) { 109. $glossary = $value->total; 110. } elseif ($value->id\_aktivitas == 7) { 111. $quiz = $value->total; 112. } elseif ($value->id\_aktivitas == 8) { 113. $wiki = $value->total; 114. } 115. } 116. $table2->data[] = array('Kooperatif', $assign, $chat, $course, $forum, $glossary, $quiz, $wiki, $na\_kooperatif); 117. $judul1 = 'Nilai Aspek'; 118. $judul2 = 'Rata-rata Nilai Aspek per Mata Kuliah'; 119. $chart = new \core\chart\_bar(); 120. $yaxis = $chart->get\_yaxis(0, true); 121. $yaxis->set\_min(0); 122. $yaxis->set\_max(1); 123. $series1 = new \core\chart\_series($judul1, [$na\_aktif, $na\_konstruktif, $na\_intensional, $na\_autentik, $na\_kooperatif]); 124. $series2 = new \core\chart\_series($judul2, [$akt, $kons, $intens, $auten, $koo]); 125. $series2->set\_type(\core\chart\_series::TYPE\_LINE); 126. $chart->add\_series($series2); 127. $chart->add\_series($series1); 128. $chart->set\_labels(['Aktif', 'Konstruktif', 'Intensional', 'Autentik', 'Kooperatif']); 129. $sql = "SELECT mdlnr\_nilai\_akhir.id, mdlnr\_nilai\_akhir.week FROM mdlnr\_nilai\_akhir WHERE mdlnr\_nilai\_akhir.id =(SELECT MAX(id) FROM mdlnr\_nilai\_akhir where mdlnr\_nilai\_akhir.id\_aspek='5' and mdlnr\_nilai\_akhir.id\_course='$courseid' and mdlnr\_nilai\_akhir.id\_user='$studentid')"; 130. $judul1 = 'Minggu Ke-'.$week; 131. $judul2 = 'Minggu Ke-'.$prev\_week; 132. $chart\_week = new \core\chart\_bar(); 133. $yaxis = $chart->get\_yaxis(0, true); 134. $yaxis->set\_min(0); 135. $yaxis->set\_max(1); 136. $series1 = new \core\chart\_series($judul1, [$na\_aktif, $na\_konstruktif, $na\_intensional, $na\_autentik, $na\_kooperatif]); 137. $series2 = new \core\chart\_series($judul2, [$na\_aktif\_prev, $na\_konstruktif\_prev, $na\_intensional\_prev, $na\_autentik\_prev, $na\_kooperatif\_prev]); 138. $series2->set\_type(\core\chart\_series::TYPE\_LINE); 139. $chart\_week->add\_series($series2); 140. $chart\_week->add\_series($series1); 141. $chart\_week->set\_labels(['Aktif', 'Konstruktif', 'Intensional', 'Autentik', 'Kooperatif']); 142. echo $OUTPUT->header(); 143. if(!empty($table->data)) { 144. echo html\_writer::table($table); 145. } 146. if(!empty($table2->data)) { 147. echo html\_writer::table($table2); 148. } 149. if (!empty($chart)) { 150. echo $OUTPUT->render($chart); 151. } 152. if (!empty($chart) && $prev\_week != 0) { 153. echo $OUTPUT->render($chart\_week); 154. } 155. echo $OUTPUT->footer(); |

Kode Sumber 4.3 Implementasi Kasus Penggunaan Melihat Detail Nilai setiap Pelajar

Tabel 4.3 Penjelasan Kode Sumber 4.3

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 2 | Menyertakan file dari folder vendor pada Moodle |
| 3 | Menyertakan file konfigurasi dari Moodle |
| 4 | Menampilkan *plugin* ‘meaningful learning’ pada menu *report* dalam *setting* *course* |
| 5 | Menampilkan *plugin* ‘meaningful learning’ pada menu *report* dalam *site administration* |
| 6 | Menginisialisasi variabel $PAGE, $COURSE, $USER, $OUTPUT, $remotedb, $CFG dari Moodle |
| 7 | Mendapatkan nama *plugin* yang sudah diinisialisasi pada file report\_meaningfullearning.php dalam folder lang/en |
| 8 | Menginisialisasi variabel studentid dengan tipe data integer |
| 9 | Menginisialisasi variabel courseid dengan tipe data integer |
| 10-19 | Mendapatkan variabel *context* |
| 20 | Mengecek kapabilitas role yang dapat mengakses *plugin* |
| 21 | Menginisialisasi alamat *url* pada Moodle |
| 22-27 | Mengonfigurasi *navigation*, *url*, *context, title, page layout, heading* yang akan ditampilkan pada halaman *plugin* |
| 29-36 | Mendapatkan data pelajar yang dipilih |
| 37-43 | Membuat tabel data pelajar |
| 44-45 | Mendapatkan variabel last\_week atau pertemuan terakhir |
| 46-82 | Mendapatkan nilai dari tiap aktivitas yang digunakan, yaitu *assignment*, forum, *chat, course, glossary,* wiki, dan *quiz* |
| 83-128 | Menampilkan *chart* untuk membandingkan nilai aspek pelajar dengan rata-rata nilai aspek mata kuliah yang dipilih |
| 129-141 | Menampilkan *chart* untuk membandingkan nilai aspek pelajar pada 2 minggu terakhir |
| 142-155 | Menampilkan tampilan *header, tabel* dan *footer* dari Moodle |

## Implementasi Perhitungan

Implementasi kasus perhitungan menggunakan Bahasa PHP dari Moodle. Pada subbab ini akan menjelaskan dan menampilkan kode perhitungan meaningful learning yang digunakan dalam pembuatan plugin report pada sisi aplikasi perangkat bergerak pada beberapa kasus penggunaan.

### Implementasi Perhitungan Frekuensi Tindakan Pelajar pada Aktivitas Moodle

Pada perhitungan frekuensi tindakan pelajar pada aktivitas Moodle terdapat 2 model perhitungan yaitu perhitungan ketika data yang diambil adalah data konten dan perhitungan ketika data yang diambil adalah data konteks.

#### Implementasi Perhitungan Frekuensi pada Data Konten

|  |
| --- |
| 1. foreach ($users as $studentid) { 2. //assignment view 3. $table\_logstore = 'logstore\_standard\_log'; 4. $select = " component = 'mod\_assign' AND crud = 'r' AND courseid = '$courseid->id'"; 5. $data = $DB->get\_records\_select($table\_logstore, $select); 6. $count\_data1 = count($data); 7. if($count\_data1 == 0){ 8. $frekuensi1 = '-1'; 9. } 10. else{ 11. $select = " component = 'mod\_assign' AND crud = 'r' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 12. $result = $DB->get\_records\_select($table\_logstore, $select); 13. $count\_result1 = count($result); 14. if($count\_result1>'0'){ 15. $frekuensi1 = '1'; 16. } 17. else{ 18. $frekuensi1 = '0'; 19. } 20. } 21. $select = " id\_aktivitas\_aksi = '1' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 22. if (!$DB->record\_exists\_select('frekuensi', $select)) { 23. $records = new stdClass(); 24. $records->id\_aktivitas\_aksi = '1'; 25. $records->id\_course = $courseid->id; 26. $records->id\_user = $studentid->userid; 27. $records->frekuensi = $frekuensi1; 28. $lastinsertid = $DB->insert\_record('frekuensi', $records); 29. } 30. else{ 31. $table\_frekuensi = 'frekuensi'; 32. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '1'"; 33. $namaid = 'id'; 34. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 35. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi1}' WHERE id = '{$id}'"); 36. } 37. //chat readlog 38. $select = " component = 'mod\_chat' AND crud = 'r' AND courseid = '$courseid->id'"; 39. $data = $DB->get\_records\_select($table\_logstore, $select); 40. $count\_data6 = count($data); 41. if($count\_data6 == 0){ 42. $frekuensi6 = '-1'; 43. } 44. else{ 45. $select = " component = 'mod\_chat' AND crud = 'r' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 46. $result = $DB->get\_records\_select($table\_logstore, $select); 47. $count\_result6 = count($result); 48. if($count\_result6>0){ 49. $sql = "SELECT avg(jmh\_all.total) as ave 50. from (select count(\*) as total 51. from mdlnr\_logstore\_standard\_log 52. where component='mod\_chat' and crud='r' and courseid='$courseid->id' group by userid) jmh\_all"; 53. $avg = $DB->get\_records\_sql($sql); 54. foreach ($avg as $listavg) { 55. if($count\_result6>=$listavg->ave){ 56. $frekuensi6=1; 57. } 58. else{ 59. $frekuensi6=$count\_result6/$listavg->ave; 60. } 61. } 62. } 63. else{ 64. $frekuensi6 = '0'; 65. } 66. $select = " id\_aktivitas\_aksi = '6' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 67. if (!$DB->record\_exists\_select('frekuensi', $select)) { 68. $records = new stdClass(); 69. $records->id\_aktivitas\_aksi = '6'; 70. $records->id\_course = $courseid->id; 71. $records->id\_user = $studentid->userid; 72. $records->frekuensi = $frekuensi6; 73. $lastinsertid = $DB->insert\_record('frekuensi', $records); 74. } 75. else{ 76. $table\_frekuensi = 'frekuensi'; 77. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '6'"; 78. $namaid = 'id'; 79. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 80. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi6}' WHERE id = '{$id}'"); 81. } 82. //course view 83. $select = " component = 'core' AND crud = 'r' AND target = 'course' AND courseid = '$courseid->id'"; 84. $data = $DB->get\_records\_select($table\_logstore, $select); 85. $count\_data7 = count($data); 86. if($count\_data7 == 0){ 87. $frekuensi7 = '-1'; 88. } 89. else{ 90. $select = " component = 'core' AND crud = 'r' AND target = 'course' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 91. $result = $DB->get\_records\_select($table\_logstore, $select); 92. $count\_result7 = count($result); 93. if($count\_result7>0){ 94. $sql = "SELECT avg(jmh\_all.total) as ave 95. from (select count(\*) as total 96. from mdlnr\_logstore\_standard\_log 97. where component='core' and target='course' and crud='r' and courseid='$courseid->id' group by userid) jmh\_all"; 98. $avg = $DB->get\_records\_sql($sql); 99. foreach ($avg as $listavg) { 100. if($count\_result7>=$listavg->ave){ 101. $frekuensi7=1; 102. } 103. else{ 104. $frekuensi7=$count\_result7/$listavg->ave; 105. } 106. } 107. } 108. else{ 109. $frekuensi7 = '0'; 110. } 111. $select = " id\_aktivitas\_aksi = '7' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 112. if (!$DB->record\_exists\_select('frekuensi', $select)) { 113. $records = new stdClass(); 114. $records->id\_aktivitas\_aksi = '7'; 115. $records->id\_course = $courseid->id; 116. $records->id\_user = $studentid->userid; 117. $records->frekuensi = $frekuensi7; 118. $lastinsertid = $DB->insert\_record('frekuensi', $records); 119. } 120. else{ 121. $table\_frekuensi = 'frekuensi'; 122. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '7'"; 123. $namaid = 'id'; 124. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 125. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi7}' WHERE id = '{$id}'"); 126. } 127. //forum viewdiscussion 128. $select = " component = 'mod\_forum' AND crud = 'r' AND courseid = '$courseid->id'"; 129. $data = $DB->get\_records\_select($table\_logstore, $select); 130. $count\_data8 = count($data); 131. if($count\_data8 == 0){ 132. $frekuensi8 = '-1'; 133. } 134. else{ 135. $select = " component = 'mod\_forum' AND crud = 'r' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 136. $result = $DB->get\_records\_select($table\_logstore, $select); 137. $count\_result8 = count($result); 138. if($count\_result8>0){ 139. $sql = "SELECT avg(jmh\_all.total) as ave 140. from (select count(\*) as total 141. from mdlnr\_logstore\_standard\_log 142. where component='mod\_forum' and crud='r' and courseid='$courseid->id' group by userid) jmh\_all"; 143. $avg = $DB->get\_records\_sql($sql); 144. foreach ($avg as $listavg) { 145. if($count\_result8>=$listavg->ave){ 146. $frekuensi8=1; 147. } 148. else{ 149. $frekuensi8=$count\_result8/$listavg->ave; 150. } 151. } 152. } 153. else{ 154. $frekuensi8 = '0'; 155. } 156. $select = " id\_aktivitas\_aksi = '8' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 157. if (!$DB->record\_exists\_select('frekuensi', $select)) { 158. $records = new stdClass(); 159. $records->id\_aktivitas\_aksi = '8'; 160. $records->id\_course = $courseid->id; 161. $records->id\_user = $studentid->userid; 162. $records->frekuensi = $frekuensi8; 163. $lastinsertid = $DB->insert\_record('frekuensi', $records); 164. } 165. else{ 166. $table\_frekuensi = 'frekuensi'; 167. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '8'"; 168. $namaid = 'id'; 169. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 170. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi8}' WHERE id = '{$id}'"); 171. } 172. //forum deleteownpost 173. $select = " component = 'mod\_forum' AND crud = 'd' AND courseid = '$courseid->id'"; 174. $data = $DB->get\_records\_select($table\_logstore, $select); 175. $count\_data11 = count($data); 176. if($count\_data11 == 0){ 177. $frekuensi11 = '-1'; 178. } 179. else{ 180. $select = " component = 'mod\_forum' AND crud = 'd' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 181. $result = $DB->get\_records\_select($table\_logstore, $select); 182. $count\_result11 = count($result); 183. if($count\_result11>0){ 184. $sql = "SELECT avg(jmh\_all.total) as ave 185. from (select count(\*) as total 186. from mdlnr\_logstore\_standard\_log 187. where component='mod\_forum' and crud='d' and courseid='$courseid->id' group by userid) jmh\_all"; 188. $avg = $DB->get\_records\_sql($sql); 189. foreach ($avg as $listavg) { 190. if($count\_result11>=$listavg->ave){ 191. $frekuensi11=1; 192. } 193. else{ 194. $frekuensi11=$count\_result11/$listavg->ave; 195. } 196. } 197. } 198. else{ 199. $frekuensi11 = '0'; 200. } 201. $select = " id\_aktivitas\_aksi = '11' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 202. if (!$DB->record\_exists\_select('frekuensi', $select)) { 203. $records = new stdClass(); 204. $records->id\_aktivitas\_aksi = '11'; 205. $records->id\_course = $courseid->id; 206. $records->id\_user = $studentid->userid; 207. $records->frekuensi = $frekuensi11; 208. $lastinsertid = $DB->insert\_record('frekuensi', $records); 209. } 210. else{ 211. $table\_frekuensi = 'frekuensi'; 212. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '11'"; 213. $namaid = 'id'; 214. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 215. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi11}' WHERE id = '{$id}'"); 216. } 217. //forum allowforcesubscribe 218. $select = " component = 'mod\_forum' AND crud = 'c' AND target = 'discussion\_subscription' AND courseid = '$courseid->id'"; 219. $data = $DB->get\_records\_select($table\_logstore, $select); 220. $count\_data12 = count($data); 221. if($count\_data12 == 0){ 222. $frekuensi12 = '-1'; 223. } 224. else{ 225. $select = " component = 'mod\_forum' AND crud = 'c' AND target = 'discussion\_subscription' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 226. $result = $DB->get\_records\_select($table\_logstore, $select); 227. $count\_result12 = count($result); 228. if($count\_result12>0){ 229. $sql = "SELECT avg(jmh\_all.total) as ave 230. from (select count(\*) as total 231. from mdlnr\_logstore\_standard\_log 232. where component='mod\_forum' and crud='c' and target='discussion\_subscription' and courseid='$courseid->id' group by userid) jmh\_all"; 233. $avg = $DB->get\_records\_sql($sql); 234. foreach ($avg as $listavg) { 235. if($count\_result12>=$listavg->ave){ 236. $frekuensi12=1; 237. } 238. else{ 239. $frekuensi12=$count\_result12/$listavg->ave; 240. } 241. } 242. } 243. else{ 244. $frekuensi12 = '0'; 245. } 246. } 247. $select = " id\_aktivitas\_aksi = '12' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 248. if (!$DB->record\_exists\_select('frekuensi', $select)) { 249. $records = new stdClass(); 250. $records->id\_aktivitas\_aksi = '12'; 251. $records->id\_course = $courseid->id; 252. $records->id\_user = $studentid->userid; 253. $records->frekuensi = $frekuensi12; 254. $lastinsertid = $DB->insert\_record('frekuensi', $records); 255. } 256. else{ 257. $table\_frekuensi = 'frekuensi'; 258. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '12'"; 259. $namaid = 'id'; 260. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 261. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi12}' WHERE id = '{$id}'"); 262. } 263. //glossary view 264. $select = " component = 'mod\_glossary' AND crud = 'r' AND courseid = '$courseid->id'"; 265. $data = $DB->get\_records\_select($table\_logstore, $select); 266. $count\_data14 = count($data); 267. if($count\_data14 == 0){ 268. $frekuensi14 = '-1'; 269. } 270. else{ 271. $select = " component = 'mod\_glossary' AND crud = 'r' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 272. $result = $DB->get\_records\_select($table\_logstore, $select); 273. $count\_result14 = count($result); 274. if($count\_result14>0){ 275. $sql = "SELECT avg(jmh\_all.total) as ave 276. from (select count(\*) as total 277. from mdlnr\_logstore\_standard\_log 278. where component='mod\_glossary' and crud='r' and courseid='$courseid->id' group by userid) jmh\_all"; 279. $avg = $DB->get\_records\_sql($sql); 280. foreach ($avg as $listavg) { 281. if($count\_result14>=$listavg->ave){ 282. $frekuensi14=1; 283. } 284. else{ 285. $frekuensi14=$count\_result14/$listavg->ave; 286. } 287. } 288. } 289. else{ 290. $frekuensi14 = '0'; 291. } 292. } 293. $select = " id\_aktivitas\_aksi = '14' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 294. if (!$DB->record\_exists\_select('frekuensi', $select)) { 295. $records = new stdClass(); 296. $records->id\_aktivitas\_aksi = '14'; 297. $records->id\_course = $courseid->id; 298. $records->id\_user = $studentid->userid; 299. $records->frekuensi = $frekuensi14; 300. $lastinsertid = $DB->insert\_record('frekuensi', $records); 301. } 302. else{ 303. $table\_frekuensi = 'frekuensi'; 304. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '14'"; 305. $namaid = 'id'; 306. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 307. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi14}' WHERE id = '{$id}'"); 308. } 309. //quiz attempt 310. $select = " component = 'mod\_quiz' AND crud = 'c' AND courseid = '$courseid->id'"; 311. $data = $DB->get\_records\_select($table\_logstore, $select); 312. $count\_data15 = count($data); 313. if($count\_data15 == 0){ 314. $frekuensi15 = '-1'; 315. } 316. else{ 317. $select = " component = 'mod\_quiz' AND crud = 'c' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 318. $result = $DB->get\_records\_select($table\_logstore, $select); 319. $count\_result15 = count($result); 320. $tablequiz= 'quiz'; 321. $selectquiz = "course = '$courseid->id'"; 322. $resultquiz = $DB->get\_records\_select($tablequiz, $selectquiz); 323. $count\_quiz = count($resultquiz); 324. if($count\_quiz =='0'){ 325. $frekuensi15 = '0'; 326. } 327. else{ 328. $frekuensi15 = $count\_result15/$count\_quiz; 329. } 330. } 331. $select = " id\_aktivitas\_aksi = '15' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 332. if (!$DB->record\_exists\_select('frekuensi', $select)) { 333. $records = new stdClass(); 334. $records->id\_aktivitas\_aksi = '15'; 335. $records->id\_course = $courseid->id; 336. $records->id\_user = $studentid->userid; 337. $records->frekuensi = $frekuensi15; 338. $lastinsertid = $DB->insert\_record('frekuensi', $records); 339. } 340. else{ 341. $table\_frekuensi = 'frekuensi'; 342. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '15'"; 343. $namaid = 'id'; 344. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 345. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi15}' WHERE id = '{$id}'"); 346. } 347. //quiz reviewmyattempts 348. $select = " component = 'mod\_quiz' AND crud = 'r' AND target = 'attempt' AND action = 'reviewed ' AND courseid = '$courseid->id'"; 349. $data = $DB->get\_records\_select($table\_logstore, $select); 350. $count\_data16 = count($data); 351. if($count\_data16 == 0){ 352. $frekuensi16 = '-1'; 353. } 354. else{ 355. $select = " component = 'mod\_quiz' AND crud = 'r' AND target = 'attempt' AND action = 'reviewed ' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 356. $result = $DB->get\_records\_select($table\_logstore, $select); 357. $count\_result16 = count($result); 358. if($count\_result16>0){ 359. $sql = "SELECT avg(jmh\_all.total) as ave 360. from (select count(\*) as total 361. from mdlnr\_logstore\_standard\_log 362. where component='mod\_quiz' and crud='r' and target='attempt' and action='reviewed' and courseid='$courseid->id' group by userid) jmh\_all"; 363. $avg = $DB->get\_records\_sql($sql); 364. foreach ($avg as $listavg) { 365. if($count\_result16>=$listavg->ave){ 366. $frekuensi16=1; 367. } 368. else{ 369. $frekuensi16=$count\_result16/$listavg->ave; 370. } 371. } 372. } 373. else{ 374. $frekuensi16 = '0'; 375. } 376. } 377. $select = " id\_aktivitas\_aksi = '16' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 378. if (!$DB->record\_exists\_select('frekuensi', $select)) { 379. $records = new stdClass(); 380. $records->id\_aktivitas\_aksi = '16'; 381. $records->id\_course = $courseid->id; 382. $records->id\_user = $studentid->userid; 383. $records->frekuensi = $frekuensi16; 384. $lastinsertid = $DB->insert\_record('frekuensi', $records); 385. } 386. else{ 387. $table\_frekuensi = 'frekuensi'; 388. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '16'"; 389. $namaid = 'id'; 390. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 391. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi16}' WHERE id = '{$id}'"); 392. } 393. //quiz view 394. $select = " component = 'mod\_quiz' AND crud = 'r' AND target = 'attempt' AND courseid = '$courseid->id'"; 395. $data = $DB->get\_records\_select($table\_logstore, $select); 396. $count\_data17 = count($data); 397. if($count\_data17 == 0){ 398. $frekuensi17 = '-1'; 399. } 400. else{ 401. $select = " component = 'mod\_quiz' AND crud = 'r' AND target = 'attempt' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 402. $result = $DB->get\_records\_select($table\_logstore, $select); 403. $count\_result17 = count($result); 404. if($count\_result17>'0'){ 405. $frekuensi17 = '1'; 406. } 407. else{ 408. $frekuensi17 = '0'; 409. } 410. } 411. $select = " id\_aktivitas\_aksi = '17' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 412. if (!$DB->record\_exists\_select('frekuensi', $select)) { 413. $records = new stdClass(); 414. $records->id\_aktivitas\_aksi = '17'; 415. $records->id\_course = $courseid->id; 416. $records->id\_user = $studentid->userid; 417. $records->frekuensi = $frekuensi17; 418. $lastinsertid = $DB->insert\_record('frekuensi', $records); 419. } 420. else{ 421. $table\_frekuensi = 'frekuensi'; 422. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '17'"; 423. $namaid = 'id'; 424. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 425. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi17}' WHERE id = '{$id}'"); 426. } 427. //wiki viewpage 428. $select = " component = 'mod\_wiki' AND crud = 'r' AND target = 'page' AND courseid = '$courseid->id'"; 429. $data = $DB->get\_records\_select($table\_logstore, $select); 430. $count\_data18 = count($data); 431. if($count\_data18 == 0){ 432. $frekuensi18 = '-1'; 433. } 434. else{ 435. $select = " component = 'mod\_wiki' AND crud = 'r' AND target = 'page' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 436. $result = $DB->get\_records\_select($table\_logstore, $select); 437. $count\_result18 = count($result); 438. if($count\_result18>0){ 439. $sql = "SELECT avg(jmh\_all.total) as ave 440. from (select count(\*) as total 441. from mdlnr\_logstore\_standard\_log 442. where component='mod\_wiki' and crud='r' and target='page' and courseid='$courseid->id' group by userid) jmh\_all"; 443. $avg = $DB->get\_records\_sql($sql); 444. foreach ($avg as $listavg) { 445. if($count\_result18>=$listavg->ave){ 446. $frekuensi18=1; 447. } 448. else{ 449. $frekuensi18=$count\_result18/$listavg->ave; 450. } 451. } 452. } 453. else{ 454. $frekuensi18 = '0'; 455. } 456. } 457. $select = " id\_aktivitas\_aksi = '18' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 458. if (!$DB->record\_exists\_select('frekuensi', $select)) { 459. $records = new stdClass(); 460. $records->id\_aktivitas\_aksi = '18'; 461. $records->id\_course = $courseid->id; 462. $records->id\_user = $studentid->userid; 463. $records->frekuensi = $frekuensi18; 464. $lastinsertid = $DB->insert\_record('frekuensi', $records); 465. } 466. else{ 467. $table\_frekuensi = 'frekuensi'; 468. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '18'"; 469. $namaid = 'id'; 470. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 471. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi18}' WHERE id = '{$id}'"); 472. } 473. //wiki createpage 474. $select = " component = 'mod\_wiki' AND crud = 'c' AND target = 'attempt' AND courseid = '$courseid->id'"; 475. $data = $DB->get\_records\_select($table\_logstore, $select); 476. $count\_data19 = count($data); 477. if($count\_data19 == 0){ 478. $frekuensi19 = '-1'; 479. } 480. else{ 481. $select = " component = 'mod\_wiki' AND crud = 'c' AND target = 'attempt' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 482. $result = $DB->get\_records\_select($table\_logstore, $select); 483. $count\_result19 = count($result); 484. if($count\_result19>0){ 485. $sql = "SELECT avg(jmh\_all.total) as ave 486. from (select count(\*) as total 487. from mdlnr\_logstore\_standard\_log 488. where component='mod\_wiki' and crud='c' and target='page' and courseid='$courseid->id' group by userid) jmh\_all"; 489. $avg = $DB->get\_records\_sql($sql); 490. foreach ($avg as $listavg) { 491. if($count\_result19>=$listavg->ave){ 492. $frekuensi19=1; 493. } 494. else{ 495. $frekuensi19=$count\_result19/$listavg->ave; 496. } 497. } 498. } 499. else{ 500. $frekuensi19 = '0'; 501. } 502. } 503. $select = " id\_aktivitas\_aksi = '19' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 504. if (!$DB->record\_exists\_select('frekuensi', $select)) { 505. $records = new stdClass(); 506. $records->id\_aktivitas\_aksi = '19'; 507. $records->id\_course = $courseid->id; 508. $records->id\_user = $studentid->userid; 509. $records->frekuensi = $frekuensi19; 510. $lastinsertid = $DB->insert\_record('frekuensi', $records); 511. } 512. else{ 513. $table\_frekuensi = 'frekuensi'; 514. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '19'"; 515. $namaid = 'id'; 516. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 517. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi19}' WHERE id = '{$id}'"); 518. } 519. //wiki editpage 520. $select = " component = 'mod\_wiki' AND crud = 'u' AND target = 'page' AND courseid = '$courseid->id'"; 521. $data = $DB->get\_records\_select($table\_logstore, $select); 522. $count\_data20 = count($data); 523. if($count\_data20 == 0){ 524. $frekuensi20 = '-1'; 525. } 526. else{ 527. $select = " component = 'mod\_wiki' AND crud = 'u' AND target = 'page' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 528. $result = $DB->get\_records\_select($table\_logstore, $select); 529. $count\_result20 = count($result); 530. if($count\_result20>0){ 531. $sql = "SELECT avg(jmh\_all.total) as ave 532. from (select count(\*) as total 533. from mdlnr\_logstore\_standard\_log 534. where component='mod\_wiki' and crud='u' and target='page' and courseid='$courseid->id' group by userid) jmh\_all"; 535. $avg = $DB->get\_records\_sql($sql); 536. foreach ($avg as $listavg) { 537. if($count\_result20>=$listavg->ave){ 538. $frekuensi20=1; 539. } 540. else{ 541. $frekuensi20=$count\_result20/$listavg->ave; 542. } 543. } 544. } 545. else{ 546. $frekuensi20 = '0'; 547. } 548. } 549. $select = " id\_aktivitas\_aksi = '20' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 550. if (!$DB->record\_exists\_select('frekuensi', $select)) { 551. $records = new stdClass(); 552. $records->id\_aktivitas\_aksi = '20'; 553. $records->id\_course = $courseid->id; 554. $records->id\_user = $studentid->userid; 555. $records->frekuensi = $frekuensi20; 556. $lastinsertid = $DB->insert\_record('frekuensi', $records); 557. } 558. else{ 559. $table\_frekuensi = 'frekuensi'; 560. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '20'"; 561. $namaid = 'id'; 562. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 563. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi20}' WHERE id = '{$id}'"); 564. } 565. //wiki viewcomment 566. $select = " component = 'mod\_wiki' AND crud = 'c' AND target = 'comments' AND courseid = '$courseid->id'"; 567. $data = $DB->get\_records\_select($table\_logstore, $select); 568. $count\_data22 = count($data); 569. if($count\_data22 == 0){ 570. $frekuensi22 = '-1'; 571. } 572. else{ 573. $select = " component = 'mod\_wiki' AND crud = 'c' AND target = 'comments' AND courseid = '$courseid->id' AND userid = '$studentid->userid'"; 574. $result = $DB->get\_records\_select($table\_logstore, $select); 575. $count\_result22 = count($result); 576. if($count\_result22>0){ 577. $sql = "SELECT avg(jmh\_all.total) as ave 578. from (select count(\*) as total 579. from mdlnr\_logstore\_standard\_log 580. where component='mod\_wiki' and crud='c' and target='comments' and courseid='$courseid->id' group by userid) jmh\_all"; 581. $avg = $DB->get\_records\_sql($sql); 582. foreach ($avg as $listavg) { 583. if($count\_result22>=$listavg->ave){ 584. $frekuensi22=1; 585. } 586. else{ 587. $frekuensi22=$count\_result22/$listavg->ave; 588. } 589. } 590. } 591. else{ 592. $frekuensi22 = '0'; 593. } 594. } 595. $select = " id\_aktivitas\_aksi = '22' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 596. if (!$DB->record\_exists\_select('frekuensi', $select)) { 597. $records = new stdClass(); 598. $records->id\_aktivitas\_aksi = '22'; 599. $records->id\_course = $courseid->id; 600. $records->id\_user = $studentid->userid; 601. $records->frekuensi = $frekuensi22; 602. $lastinsertid = $DB->insert\_record('frekuensi', $records); 603. } 604. else{ 605. $table\_frekuensi = 'frekuensi'; 606. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '22'"; 607. $namaid = 'id'; 608. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 609. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi22}' WHERE id = '{$id}'"); 610. } 611. } |

Kode Sumber 4.4 Perhitungan Frekuensi pada Data Konten

Tabel 4.4 Penjelasan Kode Sumber 4.4

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 1 | Mencari user sesuai *course* yang dipilih |
| 3 | Mendefinisikan *variable* untuk menyimpan tabel mdlnr\_logstore\_standard\_log |
| 4-36 | Menghitung frekuensi aktivitas *assign view* |
| 37-82 | Menghitung frekuensi aktivitas *chat readlog* |
| 83-128 | Menghitung frekuensi aktivitas *course view* |
| 129-174 | Menghitung frekuensi aktivitas *forum viewdiscussion* |
| 175-220 | Menghitung frekuensi aktivitas *forum deleteownpost* |
| 221-266 | Menghitung frekuensi aktivitas *forum allowforcesubscribe* |
| 267-312 | Menghitung frekuensi aktivitas *glossary view* |
| 313-350 | Menghitung frekuensi aktivitas *quiz attempt* |
| 353-396 | Menghitung frekuensi aktivitas *quiz reviewmyattempts* |
| 397-430 | Menghitung frekuensi aktivitas *quiz view* |
| 431-476 | Menghitung frekuensi aktivitas *wiki viewpage* |
| 477-522 | Menghitung frekuensi aktivitas *wiki createpage* |
| 523-568 | Menghitung frekuensi aktivitas *wiki editpage* |
| 569-611 | Menghitung frekuensi aktivitas *wiki viewcomment* |

#### Implementasi Perhitungan Frekuensi pada Data Konteks

|  |
| --- |
| 1. class nlp { 2. public function process($string) { 3. $key = "trnsl.1.1.20181025T060531Z.cf332ce2136d3182.79d50e276d17e71f553193818aa3843155931c7e"; 4. try { 5. // translate 6. $translator = new Translator($key); 7. $translate1 = $translator->translate($string, 'id-en'); 8. // normalisasi 9. $norm = Normalizer::factory("English"); 10. $norm1 = $norm->normalizeAll(explode(" ", $translate1)); 11. $stemmer = new PorterStemmer(); 12. $stem1 = $stemmer->stemAll($norm1); 13. $string1 = preg\_replace('/[!?\']/','',$stem1);//hapus tanda baca 14. $string2 = preg\_replace('/[^a-zA-Z0-9]/i',' ',$string1);//dipisahin pake spasi 15. return $string2; 16. } catch (Exception $e) { 18. } 19. } 20. } 21. // assignment submit 22. $table\_logstore = 'logstore\_standard\_log'; 23. $select = "component = 'mod\_assign' AND crud = 'u' AND objecttable='assign\_submission' AND courseid = '$courseid->id'"; 24. $data = $DB->get\_records\_select($table\_logstore, $select); 25. $count\_data2 = count($data); 26. if($count\_data2 == 0){ 27. $frekuensi2 = '-1'; 28. } 29. else{ 30. $sql = "SELECT \* FROM mdlnr\_assign\_grades ma, mdlnr\_assign mb 31. WHERE ma.assignment=mb.id 32. AND ma.userid = '$studentid->userid' 33. AND mb.course = '$courseid->id' 34. AND ma.grade >= 41"; 35. $assign=$DB->get\_records\_sql($sql); 36. $countassign1 = count($assign); 37. $flag2 = 0; 38. $frekuensi2 = 0; 39. $sum\_flag2 = 0; 40. if ($countassign1 > 0) { 41. foreach ($assign as $value) { 42. if ($value->grade >= 41) { 43. $flag2 = $flag2 +1; 44. } 45. $sum\_flag2 += $flag2; 46. $tableassign= 'assign'; 47. $selectassign = "course = '$courseid->id'"; 48. $resultassign = $DB->get\_records\_select($tableassign, $selectassign); 49. $count\_assign = count($resultassign); 50. $frekuensi2 = $sum\_flag2/$count\_assign; 51. } 52. } 53. } 54. $select = " id\_aktivitas\_aksi = '2' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 55. if (!$DB->record\_exists\_select('frekuensi', $select)) { 56. $records = new stdClass(); 57. $records->id\_aktivitas\_aksi = '2'; 58. $records->id\_course = $courseid->id; 59. $records->id\_user = $studentid->userid; 60. $records->frekuensi = $frekuensi2; 61. $lastinsertid = $DB->insert\_record('frekuensi', $records); 62. } 63. else{ 64. $table\_frekuensi = 'frekuensi'; 65. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '2'"; 66. $namaid = 'id'; 67. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 68. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi2}' WHERE id = '{$id}'"); 69. } 70. //chat chat 71. $table\_logstore = 'logstore\_standard\_log'; 72. $select = "component = 'mod\_chat' AND crud = 'c' AND courseid = '$courseid->id'"; 73. $data = $DB->get\_records\_select($table\_logstore, $select); 74. $count\_data5 = count($data); 75. if($count\_data5 == 0){ 76. $frekuensi5 = '-1'; 77. } 78. else{ 79. $sql = "SELECT mb.\* from mdlnr\_chat ma, mdlnr\_chat\_messages mb 80. WHERE ma.id = mb.chatid 81. AND ma.course = '$courseid->id' 82. AND mb.userid = '$studentid->userid' 83. AND mb.issystem = 0"; 84. $chat=$DB->get\_records\_sql($sql); 85. $countchat1 = count($chat); 86. $flag5 = 0; 87. $frekuensi5 = 0; 88. $sum\_flag5 = 0; 89. if ($countchat1 > 0) { 90. foreach ($chat as $value) { 91. $sql = "SELECT distinct ma.\* from mdlnr\_chat ma, mdlnr\_chat\_messages mb 92. WHERE ma.id = mb.chatid 93. AND ma.course = 2 94. AND mb.issystem = 0"; 95. $intro=$DB->get\_records\_sql($sql); 96. foreach ($intro as $value2) { 97. $intro1=$value2->intro; 98. $chat1=$value->message; 99. $nlp = new nlp(); 100. $string1 = $nlp->process($intro1); 101. $string2 = $nlp->process($chat1); 102. $cos = new CosineSimilarity(); 103. $result = $cos->similarity($string1,$string2); 104. if ($result >= 0.581) { 105. $flag5 = $flag5 +1; 106. } 107. $sum\_flag5 += $flag5; 108. $avg = $sum\_flag5/$count\_pelajar; 109. if($flag5 >= $avg){ 110. $frekuensi5 = 1; 111. } 112. else{ 113. $frekuensi5 = $flag5/$avg; 114. } 115. } 116. } 117. } 118. } 119. $select = " id\_aktivitas\_aksi = '5' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 120. if (!$DB->record\_exists\_select('frekuensi', $select)) { 121. $records = new stdClass(); 122. $records->id\_aktivitas\_aksi = '5'; 123. $records->id\_course = $courseid->id; 124. $records->id\_user = $studentid->userid; 125. $records->frekuensi = $frekuensi5; 126. $lastinsertid = $DB->insert\_record('frekuensi', $records); 127. } 128. else{ 129. $table\_frekuensi = 'frekuensi'; 130. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '5'"; 131. $namaid = 'id'; 132. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 133. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi5}' WHERE id = '{$id}'"); 134. } 135. // forum startdiscussion 136. $select = "component = 'mod\_forum' AND crud = 'c' AND target='discussion' AND courseid = '$courseid->id'"; 137. $data = $DB->get\_records\_select($table\_logstore, $select); 138. $count\_data9 = count($data); 139. if($count\_data9 == 0){ 140. $frekuensi9 = '-1'; 141. } 142. else{ 143. $select = "userid = '$studentid->userid'"; 144. $data = $DB->get\_records\_select('forum\_posts', $select); 145. $count\_forum = count($data); 146. $frekuensi9 = 0; 147. if($count\_forum > 0) 148. { 149. $sql = "SELECT distinct ma.\* from mdlnr\_forum\_posts ma, mdlnr\_forum\_discussions mb 150. WHERE ma.discussion = mb.id 151. AND mb.course ='$courseid->id' 152. AND ma.userid = '$studentid->userid' 153. AND ma.parent = 0"; 154. $forum=$DB->get\_records\_sql($sql); 155. $countforum1 = count($forum); 156. $flag9 = 0; 157. $sum\_flag9 = 0; 158. if ($countforum1 > 0) { 159. foreach ($forum as $value) { 160. $sql = "SELECT distinct mc.\* from mdlnr\_forum\_posts ma, mdlnr\_forum\_discussions mb, mdlnr\_forum mc 161. WHERE ma.discussion = mb.id 162. AND mb.course ='$courseid->id' 163. AND ma.userid = '$studentid->userid' 164. AND ma.parent = 0 165. AND mb.forum = mc.id"; 166. $intro=$DB->get\_records\_sql($sql); 167. $countintro = count($intro); 168. if ($countintro > 0) 169. { 170. foreach ($intro as $value2) { 171. $intro1=$value2->intro; 172. if (!$intro1) { 173. $intro1='empty'; 174. } 175. $forum1=$value->message; 176. if (!$forum1) { 177. $forum1='empty'; 178. } 179. $nlp = new nlp(); 180. $string1 = $nlp->process($intro1); 181. $string2 = $nlp->process($forum1); 182. $cos = new CosineSimilarity(); 183. $result = $cos->similarity($string1,$string2); 184. if ($result >= 0.6) { 185. $flag9 = $flag9 +1; 186. } 187. $sum\_flag9 += $flag9; 188. $avg = $sum\_flag9/$count\_pelajar; 189. if($flag9 >= $avg){ 190. $frekuensi9 = 1; 191. } 192. else{ 193. $frekuensi9 = $flag9/$avg; 194. } 195. } 196. } 197. } 198. } 199. } 200. } 201. $select = " id\_aktivitas\_aksi = '9' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 202. if (!$DB->record\_exists\_select('frekuensi', $select)) { 203. $records = new stdClass(); 204. $records->id\_aktivitas\_aksi = '9'; 205. $records->id\_course = $courseid->id; 206. $records->id\_user = $studentid->userid; 207. $records->frekuensi = $frekuensi9; 208. $lastinsertid = $DB->insert\_record('frekuensi', $records); 209. } 210. else{ 211. $table\_frekuensi = 'frekuensi'; 212. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '9'"; 213. $namaid = 'id'; 214. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 215. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi9}' WHERE id = '{$id}'"); 216. } 217. //forum replypost 218. $select = "component = 'mod\_forum' AND crud = 'c' AND target='post' AND courseid = '$courseid->id'"; 219. $data = $DB->get\_records\_select($table\_logstore, $select); 220. $count\_data10 = count($data); 221. if($count\_data10 == 0){ 222. $frekuensi10 = '-1'; 223. } 224. else{ 225. $select = "userid = '$studentid->userid'"; 226. $data = $DB->get\_records\_select('forum\_posts', $select); 227. $count\_forum = count($data); 228. $frekuensi10 = 0; 229. if($count\_forum > 0) 230. { 231. $sql = "SELECT distinct ma.\* from mdlnr\_forum\_posts ma, mdlnr\_forum\_discussions mb 232. WHERE ma.discussion = mb.id AND mb.course ='$courseid->id' AND ma.userid = '$studentid->userid'"; 233. $forum=$DB->get\_records\_sql($sql); 234. $countforum1 = count($forum); 235. $flag10 = 0; 236. $sum\_flag10 = 0; 237. if ($countforum1 > 0) { 238. foreach ($forum as $value) { 239. $sql = "SELECT \* FROM mdlnr\_forum\_posts ma WHERE ma.discussion = '$value->discussion' AND ma.parent = 0"; 240. $intro=$DB->get\_records\_sql($sql); 241. foreach ($intro as $value2) { 242. $intro1=$value2->message; 243. if (!$intro1) { 244. $intro1='empty'; 245. } 246. $forum1=$value->message; 247. if (!$forum1) { 248. $forum1='empty'; 249. } 250. $nlp = new nlp(); 251. $string1 = $nlp->process($intro1); 252. $string2 = $nlp->process($forum1); 253. $cos = new CosineSimilarity(); 254. $result = $cos->similarity($string1,$string2); 255. if ($result >= 0.59) { 256. $flag10 = $flag10 +1; 257. } 258. $sum\_flag10 += $flag10; 259. $avg = $sum\_flag10/$count\_pelajar; 260. if($flag10 >= $avg){ 261. $frekuensi10 = 1; 262. } 263. else{ 264. $frekuensi10 = $flag10/$avg; 265. } 266. } 267. } 268. } 269. } 270. } 271. $select = " id\_aktivitas\_aksi = '10' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 272. if (!$DB->record\_exists\_select('frekuensi', $select)) { 273. $records = new stdClass(); 274. $records->id\_aktivitas\_aksi = '10'; 275. $records->id\_course = $courseid->id; 276. $records->id\_user = $studentid->userid; 277. $records->frekuensi = $frekuensi10; 278. $lastinsertid = $DB->insert\_record('frekuensi', $records); 279. } 280. else{ 281. $table\_frekuensi = 'frekuensi'; 282. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '10'"; 283. $namaid = 'id'; 284. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 285. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi10}' WHERE id = '{$id}'"); 286. } 287. //glossary write 288. $select = "component = 'mod\_glossary' AND crud = 'c' AND target='entry' AND courseid = '$courseid->id'"; 289. $data = $DB->get\_records\_select($table\_logstore, $select); 290. $count\_data13 = count($data); 291. if($count\_data13 == 0){ 292. $frekuensi13 = '-1'; 293. } 294. else{ 295. $sql = "SELECT distinct mb.\* from mdlnr\_glossary ma, mdlnr\_glossary\_entries mb 296. WHERE ma.id = mb.glossaryid 297. AND ma.course = '$courseid->id' 298. AND mb.userid = '$studentid->userid'"; 299. $glossary=$DB->get\_records\_sql($sql); 300. $countglossary1 = count($glossary); 301. $flag13 = 0; 302. $frekuensi13 = 0; 303. $sum\_flag13 = 0; 304. if ($countglossary1 > 0) { 305. foreach ($glossary as $value) { 306. $sql = "SELECT distinct ma.\* from mdlnr\_glossary ma, mdlnr\_glossary\_entries mb 307. WHERE ma.id = mb.glossaryid 308. AND ma.course = '$courseid->id' 309. AND mb.userid = '$studentid->userid'"; 310. $intro=$DB->get\_records\_sql($sql); 311. foreach ($intro as $value2) { 312. $intro1=$value2->intro; 313. $glossary1=$value->definition; 314. $nlp = new nlp(); 315. $string1 = $nlp->process($intro1); 316. $string2 = $nlp->process($glossary1); 317. $cos = new CosineSimilarity(); 318. $result = $cos->similarity($string1,$string2); 319. if ($result >= 0.66) { 320. $flag13 = $flag13 +1; 321. } 322. $sum\_flag13 += $flag13; 323. $avg = $sum\_flag13/$count\_pelajar; 324. if($flag13 >= $avg){ 325. $frekuensi13 = 1; 326. } 327. else{ 328. $frekuensi13 = $flag13/$avg; 329. } 330. } 331. } 332. } 333. } 334. $select = " id\_aktivitas\_aksi = '13' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 335. if (!$DB->record\_exists\_select('frekuensi', $select)) { 336. $records = new stdClass(); 337. $records->id\_aktivitas\_aksi = '13'; 338. $records->id\_course = $courseid->id; 339. $records->id\_user = $studentid->userid; 340. $records->frekuensi = $frekuensi13; 341. $lastinsertid = $DB->insert\_record('frekuensi', $records); 342. } 343. else{ 344. $table\_frekuensi = 'frekuensi'; 345. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '13'"; 346. $namaid = 'id'; 347. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 348. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi13}' WHERE id = '{$id}'"); 349. } 350. // wiki participate 351. $select = "component = 'mod\_wiki' AND crud = 'u' AND target='page' AND courseid = '$courseid->id'"; 352. $data = $DB->get\_records\_select($table\_logstore, $select); 353. $count\_data21 = count($data); 354. if($count\_data21 == 0){ 355. $frekuensi21 = '-1'; 356. } 357. else{ 358. $select = "userid = '$studentid->userid'"; 359. $data = $DB->get\_records\_select('wiki\_pages', $select); 360. $count\_wiki = count($data); 361. $frekuensi21 = 0; 362. if($count\_wiki > 0) 363. { 364. $sql = "SELECT DISTINCT ma.\* FROM mdlnr\_wiki\_pages ma, mdlnr\_wiki\_subwikis mb, mdlnr\_wiki mc 365. WHERE ma.subwikiid=mb.id 366. AND mb.wikiid=mc.id 367. AND ma.userid = '$studentid->userid' 368. AND mc.course='$courseid->id'"; 369. $wiki=$DB->get\_records\_sql($sql); 370. $countwiki1 = count($wiki); 371. $flag21 = 0; 372. $sum\_flag21 = 0; 373. if ($countwiki1 > 0) { 374. foreach ($wiki as $value) { 375. $sql = "SELECT DISTINCT mc.\* FROM mdlnr\_wiki\_pages ma, mdlnr\_wiki\_subwikis mb, mdlnr\_wiki mc 376. WHERE ma.subwikiid=mb.id 377. AND mb.wikiid=mc.id 378. AND ma.userid = '$studentid->userid' 379. AND mc.course='$courseid->id'"; 380. $intro=$DB->get\_records\_sql($sql); 381. $countintro = count($intro); 382. if ($countintro > 0){ 383. foreach ($intro as $value2) { 384. $intro1=$value2->intro; 385. if (!$intro1) { 386. $intro1='empty'; 387. } 388. $wiki1=$value->cachedcontent; 389. if (!$wiki1) { 390. $wiki1='empty'; 391. } 392. $nlp = new nlp(); 393. $string1 = $nlp->process($intro1); 394. $string2 = $nlp->process($wiki1); 395. $cos = new CosineSimilarity(); 396. $result = $cos->similarity($string1,$string2); 397. if ($result >= 0.82) { 398. $flag21 = $flag21 +1; 399. } 400. $sum\_flag21 += $flag21; 401. $avg = $sum\_flag21/$count\_pelajar; 402. if($flag21 >= $avg){ 403. $frekuensi21 = 1; 404. } 405. else{ 406. $frekuensi21 = $flag21/$avg; 407. } 408. } 409. } 410. } 411. } 412. } 413. } 414. $select = " id\_aktivitas\_aksi = '21' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 415. if (!$DB->record\_exists\_select('frekuensi', $select)) { 416. $records = new stdClass(); 417. $records->id\_aktivitas\_aksi = '21'; 418. $records->id\_course = $courseid->id; 419. $records->id\_user = $studentid->userid; 420. $records->frekuensi = $frekuensi21; 421. $lastinsertid = $DB->insert\_record('frekuensi', $records); 422. } 423. else{ 424. $table\_frekuensi = 'frekuensi'; 425. $select = " id\_course = '$courseid->id' AND id\_user = '$studentid->userid' AND id\_aktivitas\_aksi = '21'"; 426. $namaid = 'id'; 427. $id = $DB->get\_field\_select($table\_frekuensi, $namaid, $select); 428. $DB->execute("UPDATE {frekuensi} SET frekuensi = '{$frekuensi21}' WHERE id = '{$id}'"); 429. } |

Kode Sumber 4.5 Perhitungan Frekuensi pada Data Konteks

Tabel 4.5 Penjelasan Kode Sumber 4.5

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 1-24 | Memroses data konteks menggunakan translator dan nlp php |
| 25-73 | Menghitung frekuensi aktivitas *assignment submit* |
| 74-138 | Menghitung frekuensi aktivitas *chat chat* |
| 139-220 | Menghitung frekuensi aktivitas *forum startdiscussion* |
| 221-290 | Menghitung frekuensi aktivitas *forum replypost* |
| 291-353 | Menghitung frekuensi aktivitas *glossary write* |
| 354-433 | Menghitung frekuensi aktivitas *wiki participate* |

### Implementasi Perhitungan Bobot Tindakan Aktivitas Moodle

|  |
| --- |
| 1. $sql\_bobot = "SELECT id, bobot, bobot\_baru FROM mdlnr\_bobot"; 2. $bobot\_records = $DB->get\_records\_sql($sql\_bobot); 3. foreach ($bobot\_records as $value) { 4. $DB->execute("UPDATE {bobot} SET bobot\_baru = '{$value->bobot}' WHERE id = '{$value->id}'"); 5. } 6. $sql\_bobot = "SELECT id, id\_course, id\_aktivitas\_aksi, frekuensi FROM mdlnr\_frekuensi 7. WHERE frekuensi = '-1' AND id\_course = '$courseid->id' 8. GROUP BY id\_aktivitas\_aksi"; 9. $bobot\_records = $DB->get\_records\_sql($sql\_bobot); 10. $count\_bobot = count($bobot\_records); 11. foreach ($bobot\_records as $value) { 12. if($count\_bobot > '0'){ 13. $sql = "SELECT mdlnr\_bobot.id FROM mdlnr\_bobot WHERE mdlnr\_bobot.id\_aktivitas\_aksi = '$value->id\_aktivitas\_aksi'"; 14. $get\_bobot = $DB->get\_records\_sql($sql); 15. foreach ($get\_bobot as $bbt\_new) { 16. $DB->execute("UPDATE {bobot} SET bobot\_baru = '0' WHERE id = '{$bbt\_new->id}'"); 17. } 18. } 19. } 20. $sql = "SELECT id, nama\_aspek FROM mdlnr\_aspek"; 21. $aspek\_records = $DB->get\_records\_sql($sql); 22. foreach ($aspek\_records as $aspek) { 23. $sql = "SELECT id, id\_aktivitas\_aksi, bobot, bobot\_baru FROM mdlnr\_bobot WHERE id\_aspek = '$aspek->id'"; 24. $aktivitas\_aksi = $DB->get\_records\_sql($sql); 25. $sum\_bobot\_nol = '0'; 26. $sum\_bobot\_baru = '0'; 27. foreach ($aktivitas\_aksi as $key => $value) { 28. if($value->bobot\_baru == '0'){ 29. $sum\_bobot\_nol = $sum\_bobot\_nol + $value->bobot; 30. } 31. $sum\_bobot\_baru = $sum\_bobot\_baru + $value->bobot\_baru; 32. } 33. $update\_bobot = $DB->get\_records\_sql($sql); 34. foreach ($update\_bobot as $key => $value) { 35. if($value->bobot\_baru != '0'){ 36. $bobot\_akhir = $value->bobot\_baru + ($sum\_bobot\_nol \* ($value->bobot\_baru / $sum\_bobot\_baru)); 37. $DB->execute("UPDATE {bobot} SET bobot\_baru = '$bobot\_akhir' WHERE id = '{$value->id}'"); 38. } 39. } 40. } |

Kode Sumber 4.6 Perhitungan Bobot Tindakan Aktivitas Moodle

Tabel 4.6 Penjelasan Kode Sumber 4.6

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 1-5 | Menyalin data bobot lama ke kolom bobot baru |
| 6-10 | Menghitung jumlah frekuensi yang bernilai -1 (artinya frekuensi NULL) |
| 12-19 | Mengubah kolom bobot baru menjadi 0, ketika ada id\_aktivitas\_aksi yang frekuensinya bernilai -1 (NULL) |
| 20-24 | Memilih aktivitas dan bobot dengan aspek yang dipilih |
| 25-31 | Menghitung persebaran bobot |
| 33-40 | Meng*update* bobot baru dengan hasil perhitungan dari persebaran bobot |

### Implementasi Perhitungan Aktivitas Pelajar pada Moodle

|  |
| --- |
| * + - 1. foreach ($nilai\_records as $nilai) {       2. if($nilai->frekuensi=='-1'){       3. $hasil='0'\*$nilai->bobot\_baru;       4. }       5. else{       6. $hasil=$nilai->frekuensi\*$nilai->bobot\_baru;       7. }       8. $select = " id\_frekuensi = '$nilai->id\_frek' AND id\_bobot = '$nilai->id\_bobot' AND id\_aspek = '$nilai->id\_aspek'";       9. if (!$DB->record\_exists\_select('log\_aksi', $select)) {       10. $records = new stdClass();       11. $records->id\_frekuensi = $nilai->id\_frek;       12. $records->id\_bobot = $nilai->id\_bobot;       13. $records->id\_aspek = $nilai->id\_aspek;       14. $records->nilai = $hasil;       15. $lastinsertid = $DB->insert\_record('log\_aksi', $records);       16. }       17. else{       18. $table\_log\_aksi = 'log\_aksi';       19. $select = " id\_frekuensi = '$nilai->id\_frek' AND id\_bobot = '$nilai->id\_bobot' AND id\_aspek = '$nilai->id\_aspek'";       20. $namaid = 'id';       21. $id = $DB->get\_field\_select($table\_log\_aksi, $namaid, $select);       22. $DB->execute("UPDATE {log\_aksi} SET nilai = '{$hasil}' WHERE id = '{$id}'");       23. }       24. } |

Kode Sumber 4.7 Perhitungan Aktivitas Pelajar pada Moodle

Tabel 4.7 Penjelasan Kode Sumber 4.7

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 2-4 | Menghitung nilai tindakan aktivitas pelajar (frekuensi\*bobot) dengan frekuensi -1 |
| 6 | Menghitung nilai tindakan aktivitas pelajar (frekuensi\*bobot) jika tidak ada frekuensi dengan nilai -1 |
| 8-16 | Memasukkan nilai tindakan aktivitas pelajar ke dalam table log\_aksi |
| 17-24 | Meng­-*update* nilai tindakan aktivitas pelajar ke daam table log\_aksi |

### Implementasi Perhitungan Aspek Kooperatif

|  |
| --- |
| 1. $sql = "SELECT md.id 2. FROM (mdlnr\_course\_completions mc, mdlnr\_course md) 3. WHERE md.id = mc.course GROUP BY md.id"; 4. $courses = $DB->get\_records\_sql($sql); 5. foreach ($courses as $courseid) { 6. $sql = "SELECT mc.id, mc.userid 7. FROM (mdlnr\_course\_completions mc, mdlnr\_user mu, mdlnr\_course md) 8. WHERE mc.course = '$courseid->id' 9. AND mc.userid = mu.id 10. AND md.id = mc.course"; 11. $users = $DB->get\_records\_sql($sql); 12. foreach ($users as $studentid){ 13. $sql = "SELECT DISTINCT mla.id, mla.nilai from mdlnr\_log\_aksi mla, mdlnr\_frekuensi mf, mdlnr\_course mc, mdlnr\_user mu, mdlnr\_aktivitas\_aksi maa where mla.id\_frekuensi=mf.id and mf.id\_course='$courseid->id' and mf.id\_user='$studentid->userid' and mla.id\_aspek='5' group by mf.id\_aktivitas\_aksi;"; 14. $nilaiakhir\_records = $DB->get\_records\_sql($sql); 15. $na=0; 16. foreach ($nilaiakhir\_records as $value) { 17. $na=$na+$value->nilai; 18. } 19. $select = " id\_aspek = '5' AND id\_course = '$courseid->id' AND id\_user = '$studentid->userid'"; 20. if (!$DB->record\_exists\_select('nilai\_akhir', $select)) { 21. $week = '1'; 22. } 23. else{ 24. $sql = "SELECT mdlnr\_nilai\_akhir.id, mdlnr\_nilai\_akhir.week FROM mdlnr\_nilai\_akhir WHERE mdlnr\_nilai\_akhir.id =(SELECT MAX(id) FROM mdlnr\_nilai\_akhir where mdlnr\_nilai\_akhir.id\_aspek='5'and mdlnr\_nilai\_akhir.id\_course='$courseid->id' and mdlnr\_nilai\_akhir.id\_user='$studentid->userid');"; 25. $week\_records = $DB->get\_records\_sql($sql); 26. foreach ($week\_records as $listweek) { 27. $week=$listweek->week+'1'; 28. } 29. } 30. $records = new stdClass(); 31. $records->id\_aspek = '5'; 32. $records->nilai\_akhir = $na; 33. $records->week = $week; 34. $records->id\_course = $courseid->id; 35. $records->id\_user = $studentid->userid; 36. $records->level = $lev; 37. $lastinsertid = $DB->insert\_record('nilai\_akhir', $records); 38. } 39. } |

Kode Sumber 4.8 Perhitungan Aspek Kooperatif

Tabel 4.8 Penjelasan Kode Sumber 4.8

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 1-14 | *Query* untuk memilih nilai akhir berdasarkan *course* dan *user* tertentu |
| 15-18 | Menghitung nilai akhir pelajar |
| 19-39 | Memasukkan nilai akhir pelajar ke dalam tabel mdlnr\_nilai\_akhir |

### Implementasi Perhitungan Tingkat Keterlibatan Pelajar

|  |
| --- |
| * + 1. $sql="SELECT id, nilai\_akhir from mdlnr\_nilai\_akhir where id\_course='$courseid->id' and id\_aspek='1' and week='1'";     2. $total\_nilai = $DB->get\_records\_sql($sql);     3. $sum\_nilai = 0;     4. foreach($total\_nilai as $nilai){     5. $count\_nilai = pow(($nilai->nilai\_akhir), 2);     6. $sum\_nilai += $count\_nilai;     7. $sql="SELECT count(nilai\_akhir) as jmhmurid, sum(nilai\_akhir) as total from mdlnr\_nilai\_akhir where id\_course='$courseid->id' and id\_aspek='1' and week='1'";     8. $datanilai = $DB->get\_records\_sql($sql);     9. foreach($datanilai as $listnilai){     10. $all\_nilai = pow(($listnilai->total), 2);     11. $average = $listnilai->total/$listnilai->jmhmurid;     12. $mul=$listnilai->jmhmurid\*($listnilai->jmhmurid-1);     13. if($mul <= 0){     14. $stand = 0;     15. }     16. else{     17. $stand = (float)sqrt((($listnilai->jmhmurid\*($sum\_nilai))-$all\_nilai)/($listnilai->jmhmurid\*($listnilai->jmhmurid-1)));     18. }     19. $batas\_bawah = $average-(0.5\*($stand));     20. $batas\_atas = $average+(0.5\*($stand));     21. $sql="SELECT id, nilai\_akhir from mdlnr\_nilai\_akhir where id\_course='$courseid->id' and id\_aspek='1' and id\_user='$studentid->userid' and week='1'";     22. $nilaimhs = $DB->get\_records\_sql($sql);     23. foreach($nilaimhs as $nilaifix){     24. if($nilaifix->nilai\_akhir < $batas\_bawah){     25. $lev = 'rendah';     26. }     27. elseif($nilaifix->nilai\_akhir >= $batas\_bawah && $nilaifix->nilai\_akhir < $batas\_atas)     28. {     29. $lev = 'sedang';     30. }     31. elseif($nilaifix->nilai\_akhir >= $batas\_atas)     32. {     33. $lev = 'tinggi';     34. }     35. }     36. }     37. } |

Kode Sumber 4.9 Perhitungan Tingkat Keterlibatan Pelajar

Tabel 4.9 Penjelasan Kode Sumber 4.9

|  |  |
| --- | --- |
| No. Baris | Kegunaan |
| 1 | *Query* untuk memilih nilai akhir berdasarkan *course,* aspek dan *week*. |
| 2 | Mendefinisikan *record* hasil query dari *line* 1 ke dalam bentuk variabel $total\_nilai. |
| 3 | Inisialisasi varibel $sum\_nilai sebelum dilakukan *looping* |
| 5 | Menghitung pangkat untuk tiap nilai akhir dan disimpan dalam variabel $count\_nilai |
| 6 | Menyimpan jumlah semua variabel $count\_nilai ke dalam variabel $sum\_nilai |
| 7 | *Query* untuk memilih total nilai akhir dan jumlah pelajar berdasarkan *course,* aspek dan *week*. |
| 8 | Mendefinisikan *record* hasil *query* dari *line* 8 ke dalam bentuk variabel $datanilai |
| 10 | Menghitung pangkat total nilai akhir dan disimpan dalam variabel $all\_nilai. |
| 11 | Menghitung rata-rata nilai akhir tiap *course* dan disimpan dalam vaiabel $average |
| 12 | Mendefinisikan pembagi standar deviasi ke dalam bentuk variabel $mul |
| 13-18 | Menghitung standar deviasi dan disimpan dalam variabel $stand |
| 19 | Menghitung batas bawah dan disimpan dalam variabel $batas\_bawah |
| 20 | Menghitung batas atas dan disimpan dalam variabel $batas\_atas |
| 21 | *Query* untuk memilih nilai akhir berdasarkan *course,* pelajar, aspek dan *week*. |
| 22 | Mendefinisikan *record* hasil *query* dari *line* 21 ke dalam bentuk variabel $nilai\_fix |
| 24-33 | Menentukan tingkat keterlibatan pelajar terhadap *meaningful learning* dan disimpan dalam variabel $lev |

# 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.

## 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. Kakas bantu yang dibutuhkan untuk pengujian *plugin* ini adalah:

1. Moodle versi 3.3.8.

2. *Database* MySQL dari web elearning-tc.web.id.

## Pengujian Fungsionalitas

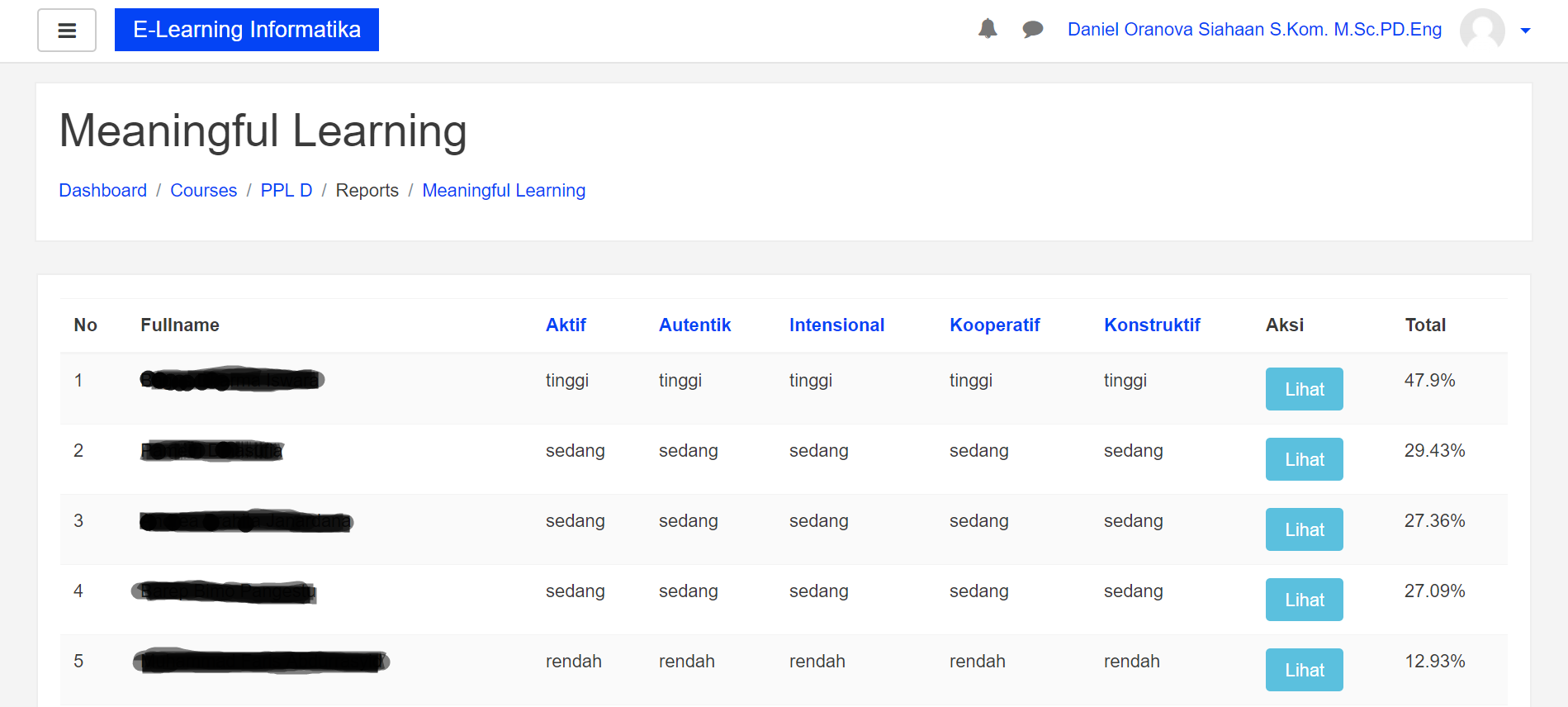
Pengujian fungsionalitas ini adalah pengujian fungsi-fungsi yang berjalan pada sistem berdasarkan kasus penggunaan. Data log dari aktivitas moodle yang digunakan dalam pengujian ini adalah data yang didapatkan dari web elearning-tc.web.id. Web elearning-tc.web.id merupakan web moodle yang kami buat untuk mengumpulkan data log aktivitas moodle dari dua kelas paralel pada bidang Rekayasa Perangkat Lunak semester ganjil tahun ajaran 2018/2019 pada strata satu. Pengujian fungsionalitas ini akan dijelaskan di setiap skenario pengujian.

### Pengujian Monitor Tingkat Meaningful Learning

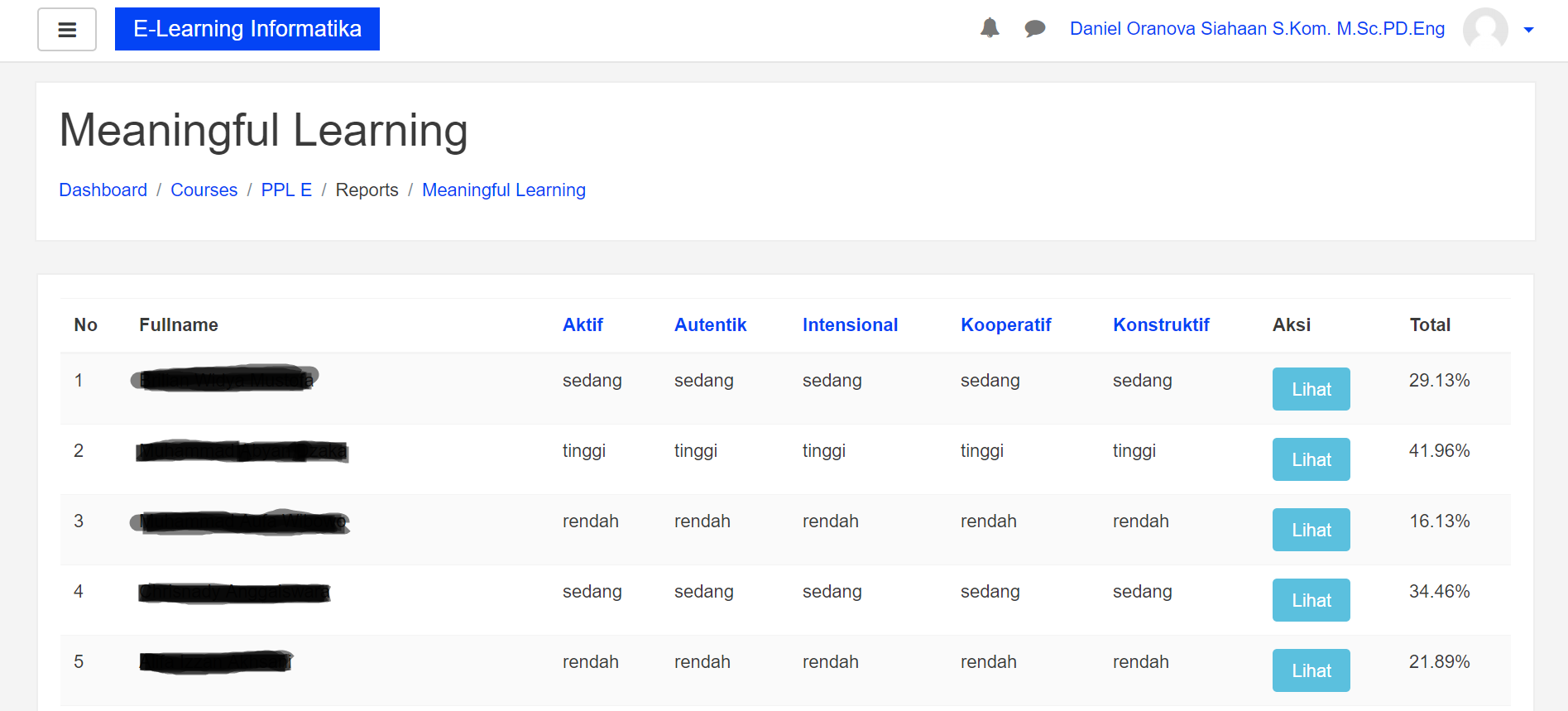
Pengujian memonitoring tingkat *meaningful learning* meruapakan pengujian terhadap kemampuan sistem (*plugin*) untuk menampilkan data tingkat *meaningful learning* sesuai dengan matakuliah yang dipilih. Pengujian ini dilakukan ketika pengajar menekan tombol “More” pada bagian menu setting dari course, kemudian pengajar memilih menu “Meaningful Learning” pada bagian Reports. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.1. Gambar contoh hasil dapat dilihat pada Gambar 5.1.

Tabel 5.1 Pengujian Memonitor Tingkat Meaningful Learning

|  |  |
| --- | --- |
| No. Pengujian | UC-001 |
| Referensi Kasus Penggunaan | UC-001 |
| Nama | Pengujian memonitor tingkat *meaningful learning* |
| Tujuan Pengujian | Menguji fungsionalitas untuk menampilkan tingkat *meaningful learning* dari setiap pelajar dengan *course* (matakuliah) yang dipilih. |
| Kondisi Awal | Pengajar berada pada halaman *course* (mata kuliah) yang dipilih. |
| Data Uji | - |
| Langkah Pengujian | 1. Pengajar memilih “More” pada tombol setting. 2. Pengajar memilih “Meaningful Learning” pada bagian Reports |
| Hasil yang diharapkan | Sistem dapat menampilkan data tingkat *meaningful learning* sesuai *course* yang dipilih. |
| Hasil yang didapat | Data tingkat *meaningful learning* (tinggi, sedang, rendah) ditampilkan. |
| Hasil Pengujian | Berhasil |
| Kondisi Akhir | Data tingkat *meaningful learning* berhasil ditampilkan |



Gambar 5.1 Pengujian Halaman Monitoring Tingkat *Meaningful Learning* Kelas PPL D



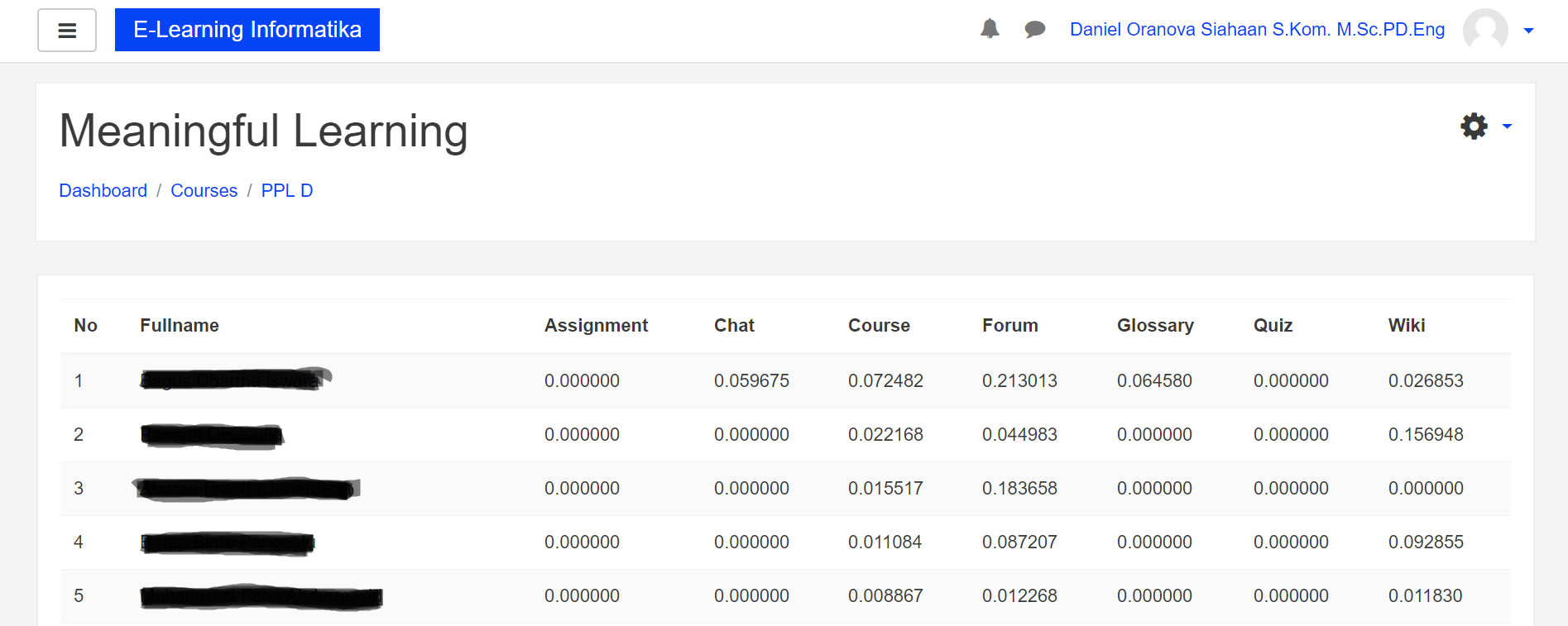
Gambar 5.2 Pengujian Halaman Monitoring Tingkat Meaningful Learning Kelas PPL E

### Pengujian Melihat Detail Nilai Aspek

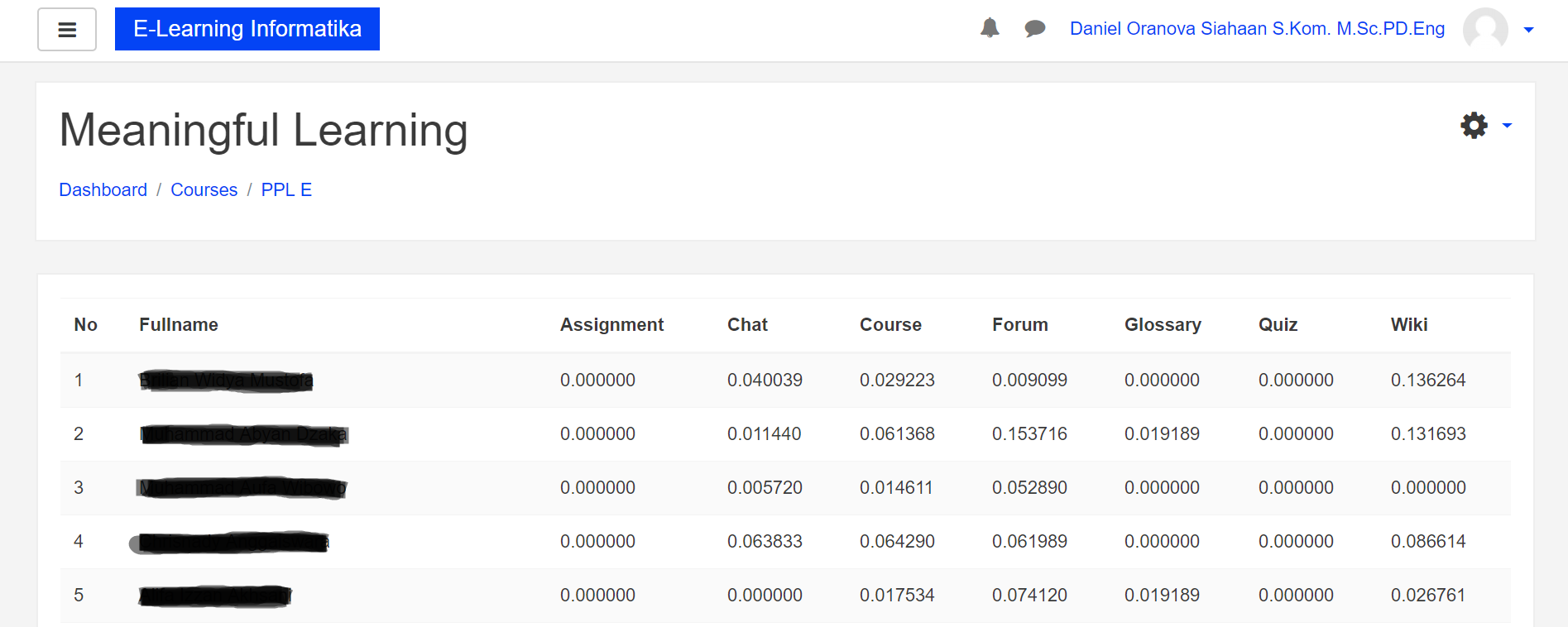
Pengujian melihat detail nilai aspek merupakan pengujian terhadap kemampuan sistem (*plugin*) untuk menampilkan data dari setiap aspek. Pengujian ini dilakukan ketika pengajar menekan aspek yang akan dilihat detailnya pada kolom masing-masing aspek. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.2. Gambar contoh hasil dapat dilihat pada Gambar 5.3.

Tabel 5.2 Pengujian Melihat Detail Nilai Aspek

|  |  |
| --- | --- |
| No. Pengujian | UC-002 |
| Referensi Kasus Penggunaan | UC-002 |
| Nama | Pengujian melihat detail nilai aspek |
| Tujuan Pengujian | Menguji fungsionalitas untuk melihat detail nilai aspek tertentu. |
| Kondisi Awal | Pengajar berada pada halaman monitoring tingkat *meaningful learning* |
| Data Uji | - |
| Langkah Pengujian | Pengajar memilih aspek yang ingin dilihat detail nilainya. |
| Hasil yang diharapkan | Sistem dapat menampilkan data detail dari nilai aspek yang terdiri dari beberapa aktivitas moodle. |
| Hasil yang didapat | Data detail dari nilai aspek yang terdiri dari beberapa aktivitas moodle ditampilkan. |
| Hasil Pengujian | Berhasil |
| Kondisi Akhir | Data detail dari nilai aspek berhasil ditampilkan |



Gambar 5.3 Pengujian Halaman Detail Nilai Aspek Kooperatif Kelas PPL D



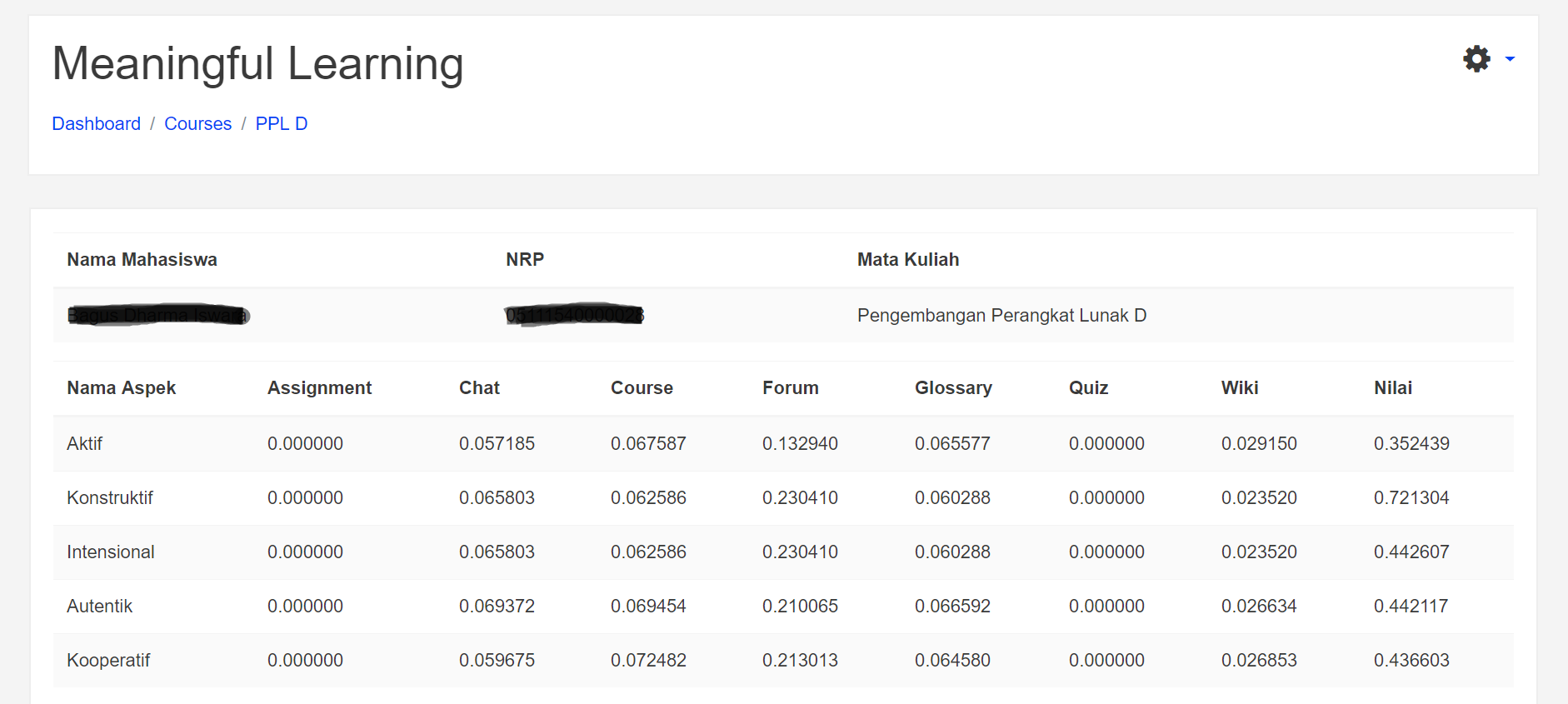
Gambar 5.4 Pengujian Halaman Detail Nilai Aspek Kooperatif Kelas PPL E

### Pengujian Melihat Detail Nilai setiap Pelajar

Pengujian melihat detail nilai setiap pelajar merupakan pengujian terhadap kemampuan sistem (*plugin*) untuk menampilkan data dari setiap pelajar yang terdiri dari beberapa aspek (aktif, konstruktif, intensional, autentik, dan kooperatif). Pengujian ini dilakukan ketika pengajar menekan nama pelajar yang akan dilihat detail nilainya pada kolom nama mahasiswa. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.3. Gambar contoh hasil dapat dilihat pada Gambar 5.5.

Tabel 5.3 Pengujian Melihat Detail Nilai setiap Pelajar

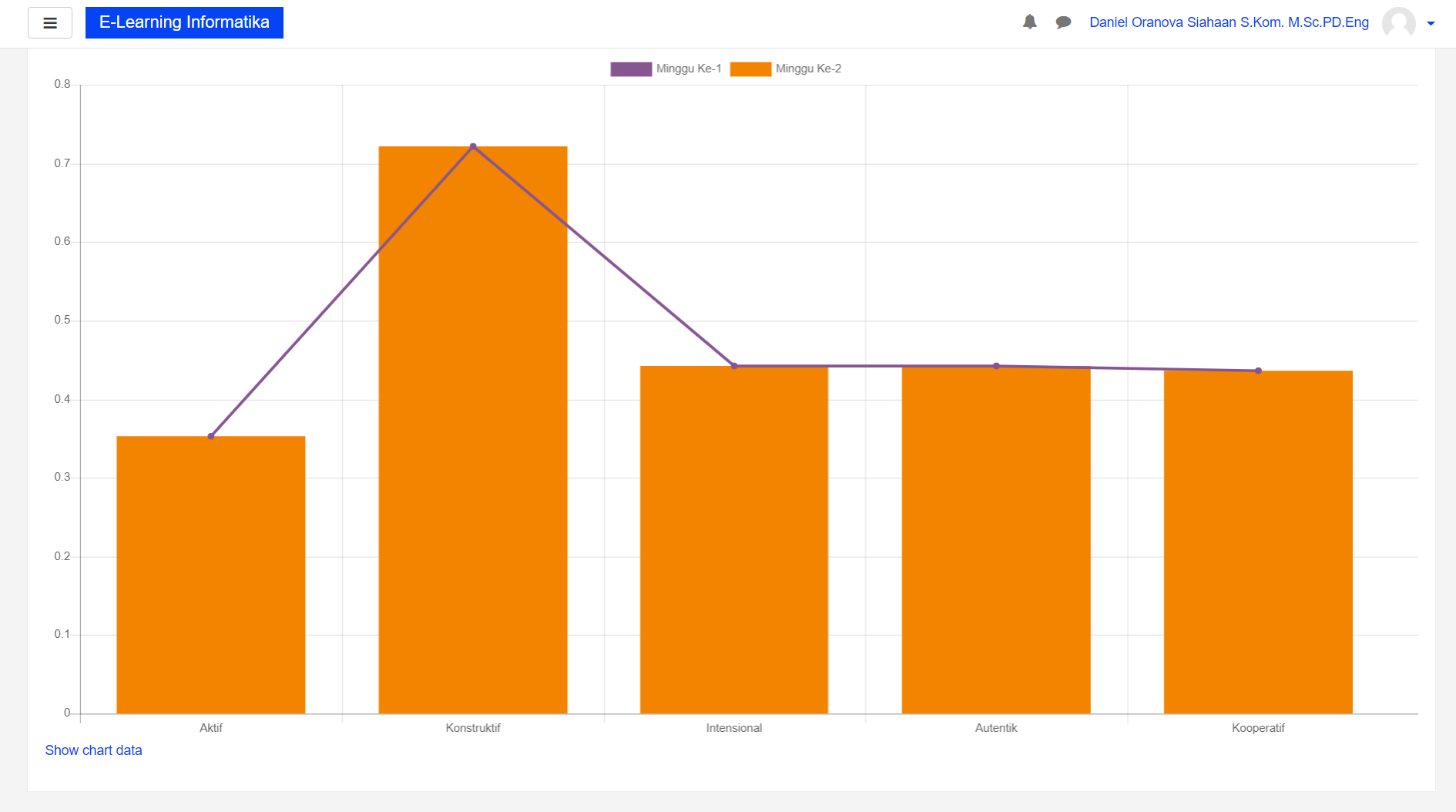
|  |  |
| --- | --- |
| No. Pengujian | UC-003 |
| Referensi Kasus Penggunaan | UC-003 |
| Nama | Pengujian melihat detail nilai setiap pelajar. |
| Tujuan Pengujian | Menguji fungsionalitas untuk melihat detail nilai setiap pelajar |
| Kondisi Awal | Pengajar berada pada halaman monitoring tingkat meaningful learning. |
| Data Uji | - |
| Langkah Pengujian | Pengajar memilih nama pelajar yang akan dilihat detail nilai tiap aspeknya. |
| Hasil yang diharapkan | Sistem dapat menampilkan detail nilai dari pelajar yang dipilih |
| Hasil yang didapat | Data detail nilai dari pelajar yang dipilih ditampilkan |
| Hasil Pengujian | Berhasil |
| Kondisi Akhir | Berhasil menampilkan detail nilai dari pelajar yang dipilih |



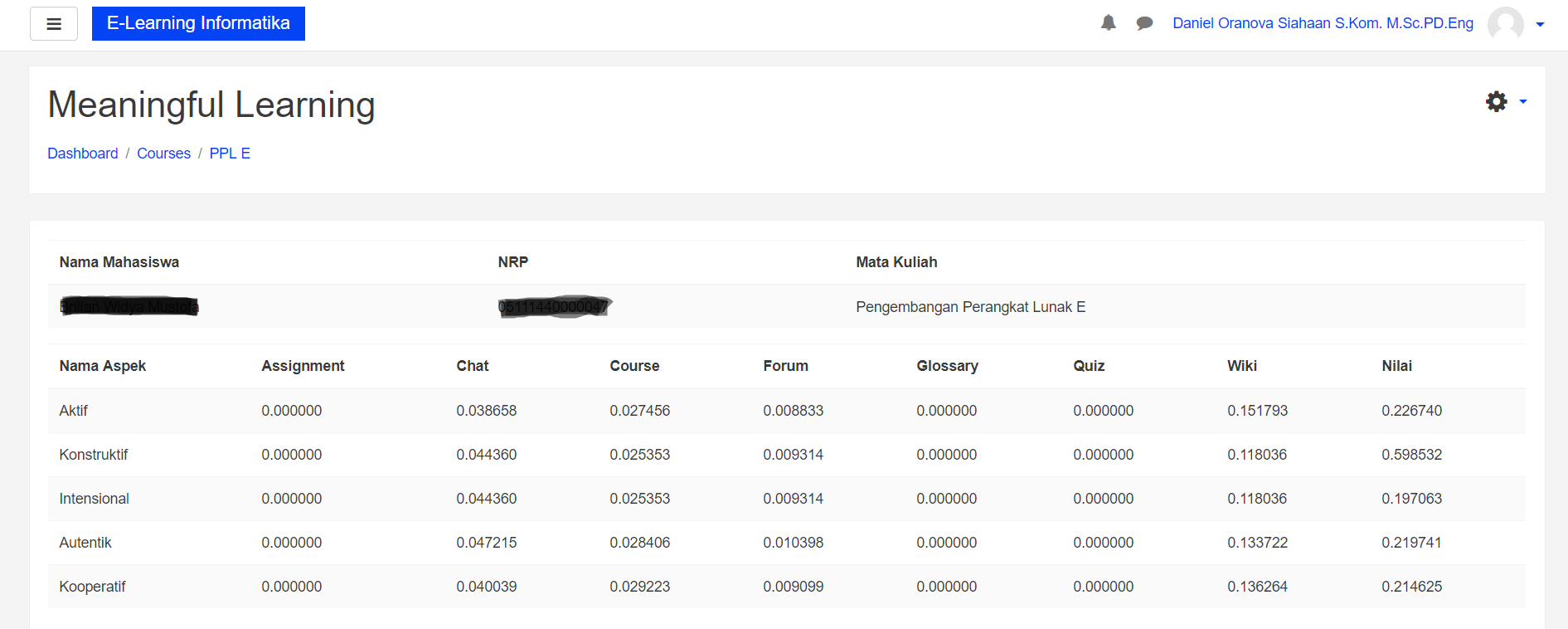
Gambar 5.5 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL D



Gambar 5.6 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL D



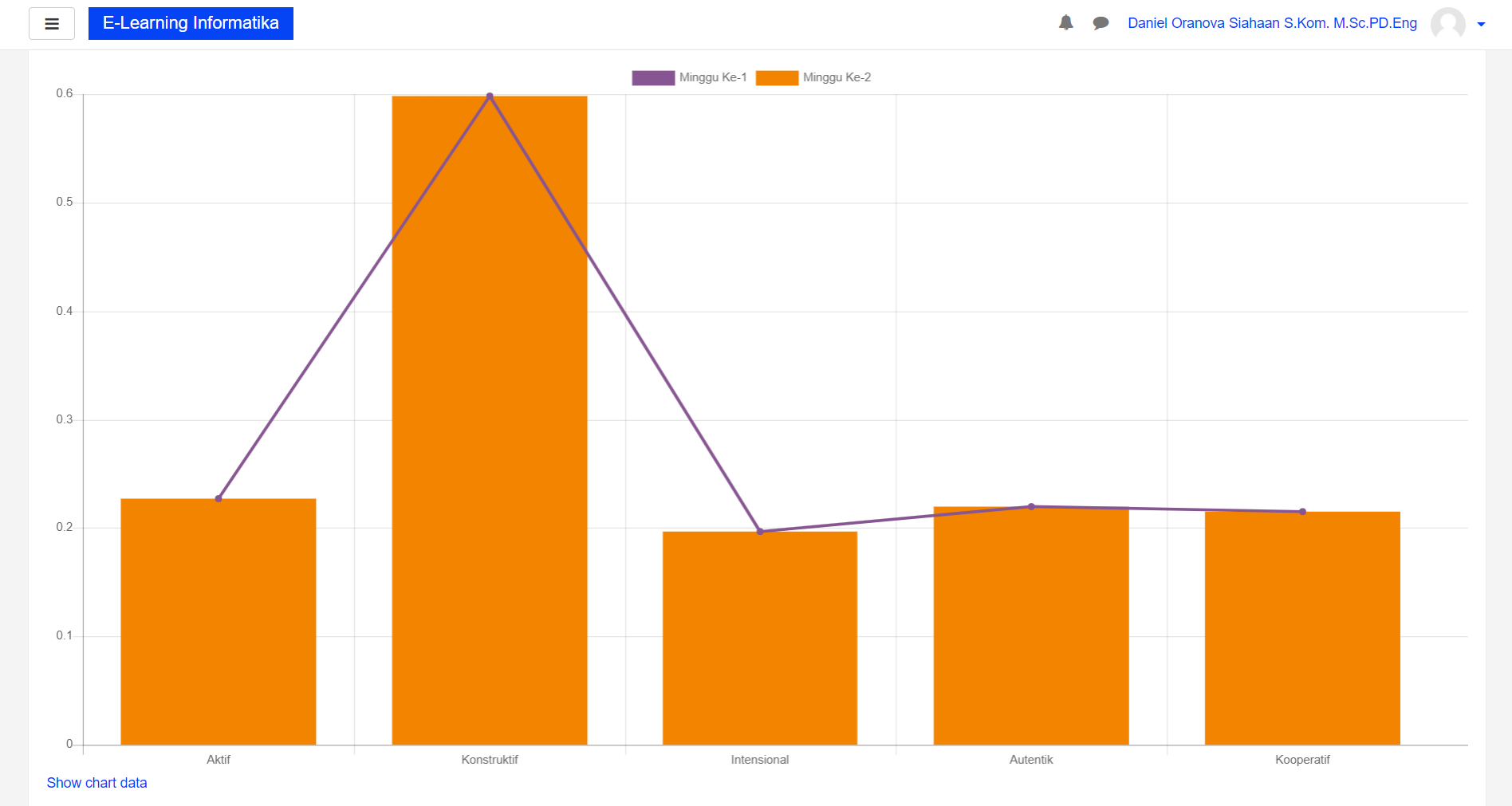
Gambar 5.7 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL D



Gambar 5.8 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL E



Gambar 5.9 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL E



Gambar 5.10 Pengujian Halaman Detail Nilai setiap Pelajar Kelas PPL E

## Evaluasi Pengujian Fungsionalitas

Berdasarkan pengujian yang telah dilakukan, semua pengujian fungsionalitas memberikan hasil yang sesuai dengan skenario yang direncanakan. Rangkuman mengenai hasil pengujian fungsionalitas yang telah diujicobakan menggunakan data dari 2 mata kuliah bidang Rekayasa Perangkat Lunak semester ganjil tahun ajaran 2018/2019 dapat dilihat pada Tabel 5.4

Tabel 5.4 Evaluasi Pengujian Fungsionalitas

| **No** | **Fungsionalitas** | **Terpenuhi** |
| --- | --- | --- |
| 1. | Memonitor Tingkat *Meaningful Learning* | ✓ |
| 2. | Melihat Detail Nilai Aspek | ✓ |
| 3. | Melihat Detail Nilai setiap Pelajar | ✓ |

Berdasarkan data pada Tabel 5.4, 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

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil pengujian yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan *plugin report* di masa mendatang.



## Kesimpulan

* + - 1. *Plugin* *report* yang dibangun dapat digunakan untuk mengukur aspek kooperatif yang mendukung pembelajaran bermakna.
      2. Mengolah data log dari setiap aktivitas Moodle yaitu dengan mencari dan menghitung frekuensi dan bobot dari data konten dan data konteks, kemudian mengubah ke dalam bentuk nilai kuantitatif yang dibutuhkan sebagai parameter pada setiap karakteristik di pembelajaran bermakna*.*
      3. Merumuskan perhitungan aspek kooperatif dengan kondisi tidak semua aktivitas Moodle dipakai oleh pengajar adalah dengan perhitungan persebaran bobot.
      4. Menentukan level yang digunakan untuk merepresentasikan tingkat keterlibatan pelajar pada aspek kooperatif dalam pembelajaran bermakna adalah dengan menggunakan persamaan standar deviasi.
      5. Mengintegrasikan pengukuran aspek kooperatif dengan aspek lain dalam pembelajaran bermakna dilakukan pada halaman antarmuka *plugin report* yang dibangun. Selain terdapat nilai aspek kooperatif setiap pelajar pada *course* yang dipilih, terdapat pula nilai aspek aktif, konstruktif, intensional dan autentik pelajar. Terdapat pula nilai rata-rata pembelajaran bermakna pelajar yang didapatkan dari penghitungan rata-rata nilai setiap aspek pembelajaran bermakna pelajar.

## Saran

1. *Plugin* dapat mengukur relevansi antara konten non teks yang disampaikan oleh pelajar pada aktivitas *assignment, chat, forum, glossary* dan *wiki* dengan deskripsi mata kuliah yang diinputkan oleh pengajar.
2. Dilakukan pendeteksian bahasa dan kata yang disingkat ataupun salah ketik pada aktivitas *assignment, chat, forum, glossary* dan *wiki* sebelum dilakukan perhitungan keserupaan semantik.
3. Dilakukan pengujian yang melibatkan pengajar secara langsung pada *e-learning.*

# DAFTAR PUSTAKA

[1] W. Wongvilaisakul and S. Lekcharoen, “The acceptance of e-Learning using SEM approach: A case of IT Literacy development for PIM students,” in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2015 12th International Conference on*, 2015, pp. 1–6.

[2] J. K. Njenga and L. C. H. Fourie, “The myths about e-learning in higher education,” *Br. J. Educ. Technol.*, vol. 41, no. 2, pp. 199–212, 2010.

[3] R. W. Dahar, *Teori-teori belajar*. Departmen Pendidikan dan Kebudayaan, Direktorat Jenderal Pendidikan Tinggi~…, 1988.

[4] D. H. Jonassen, J. Howland, J. Morre, and R. M. Marra, “Learning to solve problems with technology. A constructivist perspective . Columbus, Ohio: Pearson Education,” *Inc., Up. Saddle River, New Jersey*, 2003.

[5] A. Tenriawaru, “Pengembangan Model Pengukuran Meaningful Learning Berdasarkan Semantik Aktivitas Pelajar Dalam Lingkungan E-Learning,” 2018.

[6] D. P. Ausubel, “The psychology of meaningful verbal learning.,” 1963.

[7] W. Rice, *Moodle 1.9 E-Learning Course Development*. Packt Publishing Ltd, 2008.

[8] J. Cole and H. Foster, *Using Moodle: Teaching with the popular open source course management system*. “ O’Reilly Media, Inc.,” 2007.

[9] “Cron - MoodleDocs.” [Online]. Available: https://docs.moodle.org/36/en/Cron. [Accessed: 25-Dec-2018].

[10] T. Simpson and T. Dao, “WordNet-based semantic similarity measurement,” *Code Proj. com*, 2005.

[11] “Natural language processing tools | NlpTools PHP.” [Online]. Available: http://php-nlp-tools.com/. [Accessed: 26-Dec-2018].

[12] “Yandex — Technologies — Machine Translation,” *Yandex*.

[13] “Translate API — About this guide — Yandex Technologies.” [Online]. Available: https://tech.yandex.com/translate/doc/dg/concepts/About-docpage/. [Accessed: 26-Dec-2018].

# BIODATA PENULIS

Penulis, **Firda Rheinalia Nugroho**, lahir di Malang, 17 Juli 1998. Penulis adalah anak kedua dari tiga bersaudara. Penulis menempuh pendidikan sekolah dasar di SD Negeri Banjaran IV Kediri. Melanjutkan pendidikan sekolah menengah pertama di SMP Negeri 1 Kediri dan penulis menempuh pendidikan menengah atas di SMA Negeri 2 Kediri. Selanjutnya penulis melanjutkan pendidikan sarjana di Departemen Informatika, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya. Selama kuliah, penulis aktif menjadi administrator Laboratorium Rekayasa Perangkat Lunak di Departemen Informatika dan aktif dalam berbagai organisasi kemahasiswaan antara lain sebagai staf Departemen Kaderisasi dan Pemetaan Himpunan Mahasiswa Teknik Computer-Informatika ITS 2016-2017, staf Dana dan Usaha Schematics 2016, staf ahli Dana dan Usaha Schematics 2017, dan *Chief Administrative Officer 2* Badan Eksekutif Mahasiswa Fakultas Teknologi Informasi dan Komunikasi ITS 2017-2018.

Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Rekayasa Perangkat Lunak (RPL) dan memiliki ketertarikan pada bidang rancang bangun perangkat lunak dan pengembangan *web*. Penulis dapat dihubungi melalui email: firdarheinalia@gmail.com.