

LAMPIRAN

Lampiran 1 Surat Kesepakatan Bimbingan Skripsi

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Erryka Maura Adysti Anantacia
NIM : 22090140
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Dwi Intan Af'idah, S.T., M.Kom
Status : Dosen
NIDN : 0620089203
Jabatan Fungsional : Lektor
Pangkat/Golongan : Penata Muda Tk.I - III/b

Pada hari ini Senin tanggal 9 Maret 2026 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan syarat adanya peningkatan progres dari proyek skripsi yang dilaporkan setiap satu pekan. Jika Pihak Pertama tidak memenuhi syarat tersebut, maka Pihak Kedua berhak memutuskan kesepakatan ini. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi.

Tegal, 9 Maret 2026

Pihak Pertama



Erryka Maura Adysti Anantacia

Pihak Kedua



Dwi Intan Af'idah, S.T., M.Kom.
NIPY. 11.020.470

Mengetahui

Ketua Program Studi Sarjana Terapan
Teknik Informatika



Dyah Aprilia, S.T., M.Kom.
NIPY. 09.015.225

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini:

Pihak Pertama

Nama : Erryka Maura Adysti Anantacia
NIM : 22090140
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Sharfina Febbi Handayani, M.Kom.
Status : Dosen
NIDN : 0617029201
Jabatan Fungsional : Lektor
Pangkat/Golongan : Penata muda Tk. I - III/b

Pada hari ini Rabu tanggal 6 Mei 2026 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat Pihak Pertama bersedia bimbingan secara rutin minimal 8 kali untuk menyelesaikan Skripsi, apabila dalam kurun waktu 3 bulan setelah kesepakatan ini dibuat pihak pertama tidak konsisten untuk menyanggupi maka pihak kedua berhak membatalkan kesepakatan ini dan pihak kedua tidak memberikan rekomendasi kepada pihak pertama untuk mendaftar sidang ujian skripsi. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi.

Tegal, 6 Mei 2026

Pihak Pertama



Erryka Maura Adysti Anantacia

Pihak Kedua



Sharfina Febbi Handayani, M.Kom.
NIPY. 08.020.451

Mengetahui
Ketua Program Studi Sarjana Terapan
Teknik Informatika



Universitas
Halu Odo Negeri
Dyah Aprianti, S.T., M.Kom.
NIPY 09.015.225

Lampiran 2 Surat Pernyataan Pengajuan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. Nama : Erryka Maura Adysti Anantacia
Kewarganegaraan : Indonesia
Alamat : Jl Seram RT/RW 010/010, Kelurahan Mintaragen, Kecamatan Tegal Timur, Kota Tegal, Provinsi Jawa Tengah 52121.
2. Nama : Dwi Intan Af'idah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Grinting RT/RW 03/01, Kecamatan Bulakamba, Kabupaten Brebes, Provinsi Jawa Tengah 52253.
3. Nama : Sharfina Febbi Handayani, M.Kom
Kewarganegaraan : Indonesia
Alamat : Grobog Kulon RT/RW 02/06, Kecamatan Pangkah, Kabupaten Tegal, Provinsi Jawa Tengah 52471.

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:
Berupa : Program Komputer
Berjudul : Aplikasi Web Layanan Pelatihan Amikom Center Yogyakarta dengan Integrasi Chatbot N8N.
 - Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
 - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
 - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
 - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
 - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
 - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.
2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami langgar, maka saya / kami bersedia secara sukarela bahwa:
 - a. permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau
 - b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.

- c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini saya/kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, 8 Juni 2026



Erryka Maura Adysti Anantacia
Pemegang Hak Cipta*

Dwi Untan Af'idah, S.T., M.Kom
Pemegang Hak Cipta*

Sharfina Febbi Handayani, M.Kom
Pemegang Hak Cipta*

* Semua pemegang hak cipta agar menandatangani di atas materai (untuk penulis pertama)

Lampiran 3 Surat Pengalihan HKI

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini :

1. Nama : Erryka Maura Adysti Anantacia
Kewarganegaraan : Indonesia
Alamat : Jl.Seram RT/RW 010/010, Kelurahan Mintaragen, Kecamatan Tegal Timur, Kota Tegal, Provinsi Jawa Tengah 52121.
2. Nama : Dwi Intan Af'idah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Grinting RT/RW 03/01, Kecamatan Bulakamba, Kabupaten Brebes, Provinsi Jawa Tengah 52253.
3. Nama : Sharfina Febbi Handayani, M.Kom.
Kewarganegaraan : Indonesia
Alamat : Grobog Kulon RT/RW 02/06, Kecamatan Pangkah, Kabupaten Tegal, Provinsi Jawa Tengah 52471.

Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada :

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M) Universitas Harkat Negeri
Alamat : Jl. Mataram No. 9, Pesurungan Lor, Kota Tegal.

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Pengembangan Aplikasi Web Layanan Pelatihan Amikom Center Yogyakarta dengan Integrasi Chatbot N8N" untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikianlah surat pengalihan hak ini kami buat, agar dapat dipergunakan sebagaimana mestinya.

Pemegang Hak Cipta
Ketua P3M

 **Universitas
Harkat Negeri**


(Sharfina Febbi Handayani, M.Kom)

Tegal, 8 Juni 2026
Pencipta



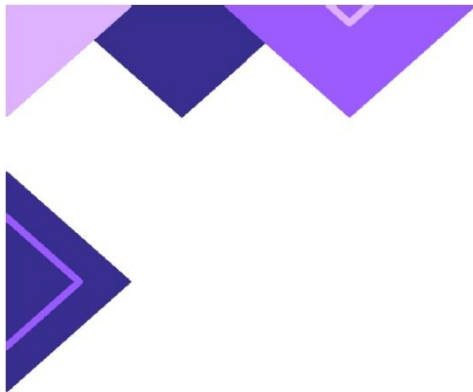
(Erryka Maura Adysti Anantacia)



(Dwi Intan Af'idah, S.T., M.Kom)



(Sharfina Febbi Handayani, M.Kom)



USER MANUAL

PENGEMBANGAN APLIKASI WEB LAYANAN PELATIHAN
AMIKOM CENTER YOGYAKARTA DENGAN INTEGRASI
CHATBOT N8N



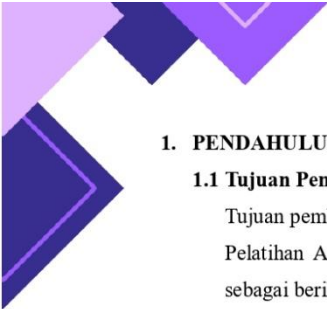
Disusun oleh:

Erryka Maura Adysti Anantacia

Dwi Intan Af'idah, S.T., M.Kom.

Sharfina Febbi Handayani, S.Kom., M.Kom.





1. PENDAHULUAN

1.1 Tujuan Pembuatan Dokumen

Tujuan pembuatan dokumen User Manual “Pengembangan Aplikasi Web Layanan Pelatihan Amikom Center Yogyakarta dengan Integrasi Chatbot N8N” adalah sebagai berikut:

1. Memberikan panduan penggunaan aplikasi Amikom Center kepada peserta.
2. Membantu peserta memahami fitur-fitur yang tersedia pada sistem LMS Amikom Center.
3. Membantu peserta menggunakan aplikasi Amikom Center dengan lebih mudah, efektif, dan terstruktur.

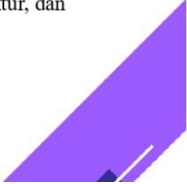
1.2 Deskripsi Umum Sistem

1.2.1 Deskripsi Umum Aplikasi

Amikom Center merupakan aplikasi Learning Management System (LMS) berbasis web yang digunakan sebagai media pembelajaran online bagi peserta pelatihan dan sertifikasi di Amikom Center Universitas AMIKOM Yogyakarta. Aplikasi ini menyediakan berbagai fitur pembelajaran seperti course online, materi video dan PDF, quiz, final exam, progress pembelajaran, sertifikat digital, artikel, dan chatbot informasi Amikom Center. Seluruh proses pembelajaran dilakukan secara terintegrasi dalam satu platform sehingga peserta dapat mengikuti pelatihan dengan lebih mudah dan fleksibel.

1.2.2 Deskripsi Umum Kebutuhan Aplikasi

Aplikasi LMS Amikom Center membutuhkan perangkat yang terhubung dengan internet agar peserta dapat mengakses seluruh fitur pembelajaran secara online. Sistem ini juga memerlukan browser yang kompatibel untuk menampilkan materi pembelajaran seperti video, file PDF, quiz, dan final exam dengan baik. Selain itu, aplikasi membutuhkan sistem penyimpanan progress belajar untuk memantau aktivitas peserta berdasarkan durasi menonton video dan aktivitas scroll pada materi PDF. Fitur evaluasi seperti quiz dan final exam juga diperlukan untuk mengetahui tingkat pemahaman peserta terhadap materi yang telah dipelajari. Dengan adanya fitur sertifikat digital dan chatbot informasi Amikom Center, aplikasi dapat membantu peserta mengikuti proses pembelajaran secara lebih mudah, terstruktur, dan efisien.





1.3 Deskripsi Dokumen

Dokumen User Manual ini berisi panduan penggunaan aplikasi “Pengembangan Aplikasi Web Layanan Pelatihan Amikom Center Yogyakarta dengan Integrasi Chatbot N8N” dari sudut pandang peserta. Dokumen menjelaskan kebutuhan perangkat yang digunakan untuk mengakses aplikasi serta langkah penggunaan fitur-fitur yang tersedia pada sistem. Isi dokumen dibagi menjadi beberapa bab, yaitu:

1. BAB 1 Pendahuluan

Berisi tujuan pembuatan dokumen, deskripsi umum sistem, dan deskripsi dokumen.

2. BAB 2 Perangkat yang Dibutuhkan

Berisi informasi perangkat keras dan perangkat lunak yang diperlukan untuk menjalankan aplikasi Amikom Center.

3. BAB 3 Cara Penggunaan

Berisi penjelasan menu-menu pada aplikasi serta langkah penggunaan fitur-fitur yang tersedia pada Amikom Center.

2. PERANGKAT YANG DIBUTUHKAN

2.1 Perangkat Lunak

Perangkan lunak yang digunakan adalah:

1. Windows sebagai operating system.
2. Google Chrome, Mozilla Firefox, dan Microsoft Edge sebagai web browser.
3. Visual Studio Code sebagai text editor.
4. XAMPP sebagai database server.

2.2 Perangkat Keras

Perangkat keras yang digunakan adalah:

1. Laptop atau Komputer.
2. Mouse sebagai peralatan antarmuka.
3. Keyboard sebagai peralatan antarmuka.
4. Koneksi internet untuk mengakses aplikasi Amikom Center.

2.3 Pengguna Aplikasi

Pengguna aplikasi yang menggunakan aplikasi Amikom Center adalah sebagai berikut:

1. Peserta: Pengguna yang dapat melakukan registrasi akun, login, enroll course, mengakses materi pembelajaran, mengerjakan quiz dan final exam, melihat progress pembelajaran, mengakses sertifikat, membaca artikel, dan menggunakan chatbot Amikom Center.

3. CARA PENGGUNAAN

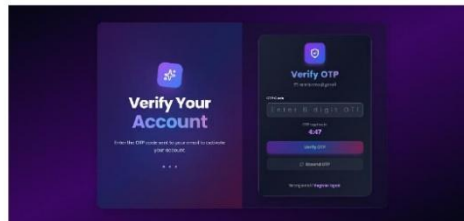
3.1 Registrasi Akun

Berikut langkah-langkah melakukan registrasi akun pada aplikasi Amikom Center:

1. Buka halaman register Amikom Center.



2. Masukkan semua data yang dibutuhkan untuk registrasi akun.
3. Klik tombol "Register".
4. Sistem akan mengirim kode OTP ke email pengguna.
5. Masukkan kode OTP pada halaman verifikasi email.



6. Jika OTP benar, akun berhasil dibuat dan pengguna diarahkan ke halaman Login Amikom Center.

3.2 Login

Berikut langkah-langkah melakukan login pada aplikasi Amikom Center:

1. Buka halaman login aplikasi Amikom Center.



2. Masukkan email dan password yang telah terdaftar.
3. Klik tombol “Login”.
4. Jika data benar, pengguna berhasil masuk ke dashboard aplikasi Amikom Center.

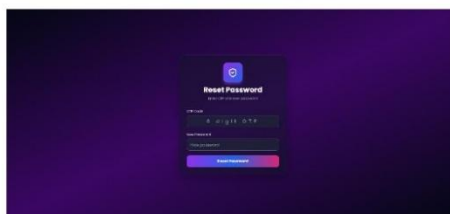
3.3 Forgot Password

Berikut langkah-langkah melakukan reset password:

1. Klik menu “Forgot Password” pada halaman login.



2. Masukkan email yang terdaftar pada sistem.
3. Klik tombol “Kirim OTP”.
4. Sistem akan mengirim kode OTP ke email pengguna.
5. Masukkan kode OTP pada halaman verifikasi.



6. Buat password baru.
7. Klik tombol “Reset Password”.
8. Password berhasil diperbarui dan pengguna dapat login menggunakan password baru.



3.4 Enroll Course

Berikut langkah-langkah enroll course:

1. Buka detail course yang di pilih.



2. Klik tombol "Enroll."
3. Masukkan semua data yang dibutuhkan untuk enroll course.
4. Klik tombol "Submit".
5. Pengguna berhasil enroll dan dapat mulai mengakses materi pembelajaran.

3.5 Mengakses Materi

Berikut langkah-langkah mengakses materi pembelajaran:

1. Pilih course yang telah di-enroll.

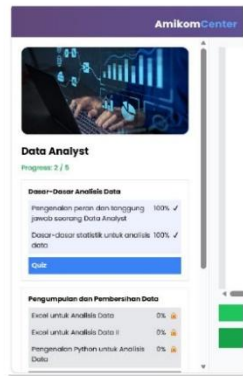
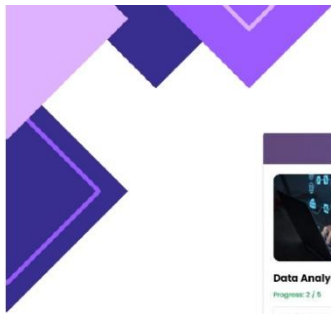


2. Pilih materi pembelajaran yang tersedia.
3. Materi dapat berupa video atau file PDF.
4. Sistem akan menyimpan progress pembelajaran secara otomatis.

3.6 Melihat Progress Pembelajaran

Berikut langkah-langkah melihat progress pembelajaran:

1. Buka halaman course yang sedang dipelajari.

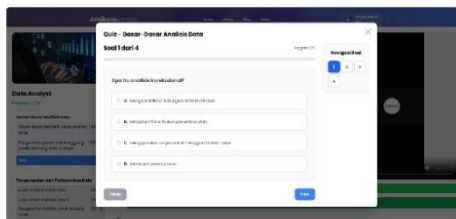


2. Sistem akan menampilkan progress pembelajaran pengguna.
3. Progress video dihitung berdasarkan durasi tontonan video.
4. Progress PDF dihitung berdasarkan aktivitas scroll pada materi PDF.
5. Materi akan berstatus selesai jika progress mencapai 100%.

3.7 Mengerjakan Quiz

Berikut langkah-langkah mengerjakan quiz:

1. Selesaikan seluruh materi pembelajaran terlebih dahulu.
2. Buka halaman quiz pada course.



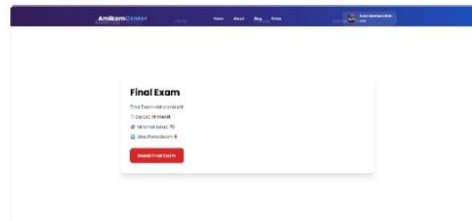
3. Pilih jawaban pada setiap soal pilihan ganda.
4. Klik tombol "Submit".
5. Sistem akan menampilkan hasil nilai quiz.



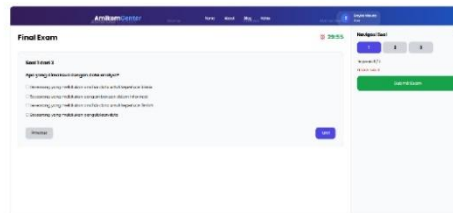
3.8 Mengerjakan Final Exam

Berikut langkah-langkah mengerjakan final exam:

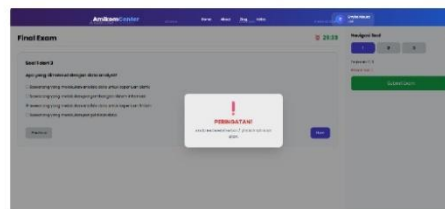
1. Selesaikan seluruh materi pembelajaran.
2. Buka halaman final exam pada course.



3. Klik tombol “Mulai Final Exam”.
4. Kerjakan soal yang tersedia.



5. Perhatikan durasi pengerjaan yang berjalan pada halaman exam.
6. Hindari berpindah tab/browser selama exam berlangsung karena sistem akan memberikan peringatan.



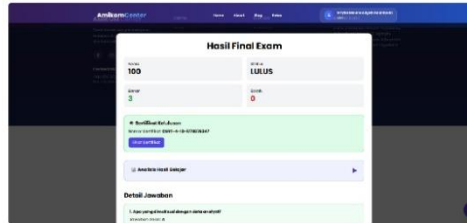
7. Klik tombol “Submit” setelah selesai mengerjakan exam.
8. Sistem akan menampilkan nilai akhir, status kelulusan, dan akses untuk melihat sertifikat.

3.9 Mengakses Sertifikat

Berikut langkah-langkah mengakses sertifikat:

1. Selesaikan final exam dengan status lulus.

2. Buka halaman hasil final exam.



3. Klik tombol “Lihat Sertifikat.”
4. Sistem akan menampilkan sertifikat kelulusan pengguna.

3.10 Membaca Artikel

Berikut langkah-langkah membaca artikel:

1. Buka menu “Blog”.

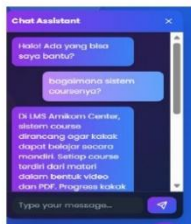


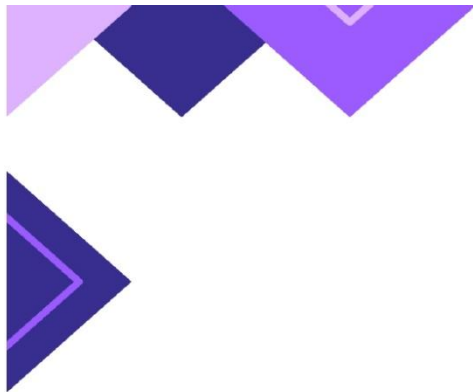
2. Sistem akan menampilkan daftar artikel yang tersedia.
3. Klik artikel yang ingin dibaca.
4. Sistem akan menampilkan detail isi artikel.

3.11 Menggunakan Chatbot

Berikut langkah-langkah menggunakan chatbot Amikom Center:

1. Buka fitur chatbot pada aplikasi Amikom Center.
2. Ketik pertanyaan seputar Amikom Center.
3. Klik tombol “Send”.
4. Chatbot akan menampilkan jawaban sesuai dengan pertanyaan pengguna.





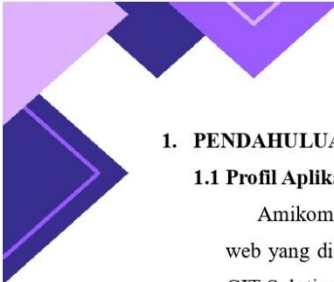
DOKUMEN TEKNIKAL

PENGEMBANGAN APLIKASI WEB LAYANAN PELATIHAN
AMIKOM CENTER YOGYAKARTA DENGAN INTEGRASI
CHATBOT N8N



Disusun oleh:
Erryka Maura Adysti Anantacia
Dwi Intan Af'idah, S.T., M.Kom.
Sharfina Febbi Handayani, S.Kom., M.Kom.





1. PENDAHULUAN

1.1 Profil Aplikasi

Amikom Center merupakan Aplikasi Learning Management (LMS) berbasis web yang di kembangkan untuk mendukung kegiatan pembelajaran online di PT GIT Solution. Aplikasi ini dirancang untuk memudahkan peserta dalam mengikuti pelatihan secara terstruktur melalui satu platform terintegrasi. Melalui Amikom Center, peserta dapat melakukan registrasi akun, mengikuti course yang tersedia, mengakses materi pembelajaran dalam bentuk video maupun PDF, serta memantau progress pembelajaran secara mandiri. Selain tu, sistem juga menyediakan fitur evaluasi berupa quiz dan final exam yang digunakan untuk mengukur pemahaman peserta terhadap materi yang telah dipelajari. Peserta yang berhasil memenuhi kriteria kelulusan dapat memperoleh sertifikat digital yang dapat diakses langsung melalui sistem.

Aplikasi ini juga dilengkapi dengan fitur artikel sebagai media informasi dan chatbot yang berfungsi memberikan informasi seputar Amikom Center. Sistem dikembangkan menggunakan React sebagai frontend, Flask sebagai backend, dan MySQL sebagai basis data yang dikelola melalui XAMPP. Kombinasi teknologi tersebut memungkinkan sistem berjalan secara responsif, mudah diakses, dan mampu mengelola data pembelajaran secara terpusat. Dengan adanya Amikom Center, proses pembelajaran dapat dilakukan secara lebih efektif, fleksibel, dan dapat diakses kapan saja melalui perangkat yang terhubung dengan internet.

1.2 Latar Belakang

Amikom Center merupakan pusat pelatihan yang menyediakan berbagai program pembelajaran bagi mahasiswa maupun masyarakat umum. Seiring meningkatnya kebutuhan pembelajaran digital, diperlukan sebuah sistem yang mampu menyediakan materi pembelajaran, evaluasi, dan sertifikat dalam satu platform terintegrasi. Oleh karena itu, Amikom Center dikembangkan untuk mempermudah peserta dalam mengikuti proses pembelajaran secara online. Sistem ini memungkinkan peserta mengakses materi, memantau progress belajar, mengikuti quiz dan final exam, serta memperoleh sertifikat secara digital.

1.3 Manfaat Aplikasi

A. Manfaat bagi Peserta

1. Memudahkan akses pembelajaran secara online.
2. Mempermudah pemantauan progress belajar.

3. Memudahkan proses evaluasi pembelajaran.
 4. Memperoleh sertifikasi secara digital.
- B. Manfaat bagi Amikom Center
1. Mempermudah pengelolaan course.
 2. Mempermudah pengelolaan materi pembelajaran.
 3. Meningkatkan kualitas layanan pembelajaran.

2. SPESIFIKASI TEKNIS

2.1 Perangkat Lunak

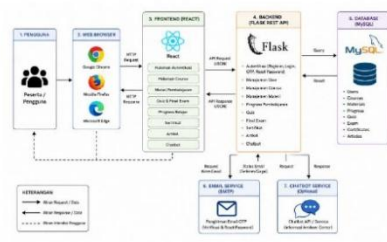
1. Windows 11 sebagai sistem operasi.
2. Vsual Studio Code sebagai text editor dan lingkungan pengembangan aplikasi.
3. Python 3.11 sebagai bahasa pemrograman backend.
4. Flask sebagai framework backend.
5. React.js sebagai framework frontend.
6. MySQL sebagai Database Management System.
7. XAMPP sebagai web server dan database server.
8. Google Chrome sebagai browser untuk pengembangan dan pengujian aplikasi.

2.2 Perangkat Keras

1. Laptop atau Komputer.
2. Mouse sebagai peralatan antarmuka.
3. Keyboard sebagai peralatan antarmuka.
4. Koneksi internet untuk mengakses aplikasi Amikom Center.

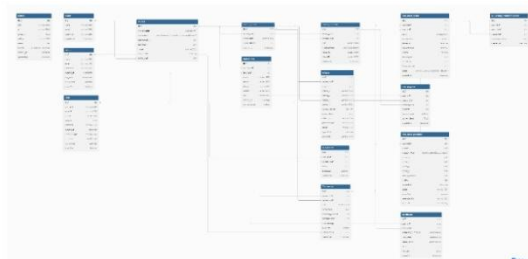
3. ARSITEKTUR SISTEM

3.1 Arsitektur Sistem



Amikom Center menerapkan arsitektur client-server dengan pemisahan antara frontend dan backend. Frontend dibangun menggunakan React.js yang bertugas menampilkan antarmuka pengguna dan menerima interaksi dari peserta. Backend dibangun menggunakan Flask yang berfungsi menangani proses bisnis aplikasi, seperti autentikasi pengguna, pengelolaan course, progress pembelajaran, quiz, final exam, sertifikat, artikel, dan chatbot. Data aplikasi disimpan pada database MySQL yang dikelola menggunakan XAMPP.

3.2 Arsitektur Database



Arsitektur database pada Amikom Center ini dirancang menggunakan model basis data relasional untuk mendukung pengelolaan data yang terstruktur dan konsisten. Setiap entitas dalam sistem direpresentasikan ke dalam tabel yang saling terhubung melalui relasi foreign key. Pendekatan ini dipilih untuk menjaga integritas data, mengurangi redundansi informasi, serta mempermudah proses pengelolaan dan pengembangan sistem di masa mendatang.

Tabel users berfungsi sebagai pusat penyimpanan data pengguna yang mengakses sistem pembelajaran. Informasi yang disimpan meliputi identitas pengguna, data kontak, kredensial login, serta data verifikasi akun. Tabel ini memiliki hubungan dengan berbagai tabel lain karena hampir seluruh aktivitas pembelajaran dilakukan oleh pengguna yang terdaftar pada sistem.

Tabel admin digunakan untuk menyimpan data administrator yang bertanggung jawab terhadap pengelolaan sistem. Data yang disimpan meliputi nama, email, kata sandi, dan informasi verifikasi akun. Administrator memiliki peran penting dalam pengelolaan ujian akhir karena menjadi pihak yang membuat dan mengatur pelaksanaan evaluasi pembelajaran.

Tabel tutor digunakan untuk menyimpan informasi pengajar yang bertanggung jawab terhadap penyusunan dan pengelolaan kursus. Setiap tutor dapat mengelola



lebih dari satu kursus sesuai dengan bidang keahlian yang dimiliki. Hubungan antara tabel tentor dan courses bersifat one-to-many karena satu pengajar dapat mengampu banyak kursus.

Tabel courses merupakan entitas utama yang menyimpan informasi mengenai program pembelajaran yang tersedia dalam sistem. Data yang disimpan mencakup nama kursus, tingkat kesulitan, deskripsi, gambar, dan informasi pengajar. Setiap kursus dapat memiliki beberapa bagian materi yang disusun secara berurutan untuk membentuk alur pembelajaran yang sistematis.

Struktur materi dalam setiap kursus dikelola melalui tabel section_course. Tabel ini berfungsi untuk membagi kursus menjadi beberapa bagian atau topik pembelajaran yang lebih spesifik. Dengan adanya pembagian section, peserta dapat mengikuti materi secara bertahap sesuai urutan yang telah ditentukan.

Materi pembelajaran disimpan pada tabel course_materials yang berelasi dengan tabel courses dan section_course. Sistem mendukung berbagai jenis materi, seperti video dan ebook, sehingga proses pembelajaran dapat dilakukan secara lebih fleksibel. Setiap materi memiliki informasi terkait lokasi penyimpanan file yang digunakan untuk menampilkan konten kepada pengguna.

Aktivitas pembelajaran pengguna terhadap materi dicatat pada tabel user_progress. Tabel ini menyimpan informasi mengenai tingkat penyelesaian materi, status penyelesaian, dan posisi terakhir pengguna saat mempelajari suatu materi. Data tersebut digunakan untuk membantu pengguna melanjutkan proses belajar tanpa harus mengulang dari awal.

Proses pendaftaran kursus dikelola melalui tabel enrollments. Tabel ini menghubungkan data pengguna dengan kursus yang diikuti serta menyimpan informasi tambahan seperti nama, email, nomor telepon, instansi, dan kota asal peserta. Relasi ini memungkinkan satu pengguna untuk mengikuti beberapa kursus yang tersedia dalam sistem.

Evaluasi pembelajaran pada tingkat section dilakukan menggunakan tabel quizzes. Tabel ini menyimpan soal-soal kuis beserta pilihan jawaban, jawaban benar, tingkat kesulitan, dan informasi terkait proses pembuatan soal. Setiap kuis terhubung dengan section tertentu sehingga dapat digunakan untuk mengukur pemahaman peserta terhadap materi yang baru dipelajari.

Hasil pengerjaan kuis oleh peserta disimpan pada tabel quiz_results. Informasi yang dicatat meliputi identitas pengguna, section yang diujikan, skor yang diperoleh,



serta status kelulusan. Data ini dapat digunakan untuk melakukan analisis terhadap tingkat pemahaman peserta pada setiap bagian pembelajaran.

Evaluasi akhir pembelajaran dikelola melalui tabel `final_exams`. Tabel ini menyimpan informasi mengenai ujian akhir yang terkait dengan suatu kursus atau section tertentu. Selain itu, tabel ini juga mencatat durasi ujian, nilai minimum kelulusan, jumlah percobaan maksimum, dan status aktif ujian.

Soal-soal yang digunakan dalam ujian akhir disimpan pada tabel `final_exam_questions`. Tabel ini mendukung beberapa tipe soal, seperti pilihan tunggal, pilihan ganda, dan esai. Sistem juga menyediakan atribut tambahan yang memungkinkan proses verifikasi soal serta pencatatan sumber pembuatan soal, baik secara manual maupun menggunakan teknologi kecerdasan buatan.

Data hasil pengerjaan ujian akhir dicatat pada tabel `final_exam_results`. Tabel ini menyimpan informasi skor peserta, jawaban yang diberikan, waktu pengerjaan, status ujian, dan berbagai indikator aktivitas selama ujian berlangsung. Informasi tersebut digunakan sebagai dasar penentuan kelulusan peserta pada suatu kursus.

Sistem menyediakan fitur rekomendasi pembelajaran berbasis kecerdasan buatan yang datanya disimpan pada tabel `ai_learning_recommendations`. Rekomendasi dihasilkan berdasarkan hasil evaluasi peserta pada ujian akhir yang telah diselesaikan. Dengan adanya fitur ini, sistem dapat memberikan saran pembelajaran yang lebih personal sesuai kebutuhan masing-masing pengguna.

Penerbitan sertifikat dikelola melalui tabel `certificates`. Tabel ini menyimpan informasi nomor sertifikat, identitas peserta, nama kursus, nilai yang diperoleh, lokasi penyimpanan file sertifikat, dan tanggal penerbitan. Sertifikat diberikan kepada peserta yang telah memenuhi kriteria kelulusan yang ditentukan oleh sistem.

Tabel `articles` digunakan untuk menyimpan artikel atau informasi pendukung yang dapat diakses oleh pengguna. Artikel dapat ditambahkan secara manual maupun diperoleh melalui proses scraping dari sumber eksternal. Keberadaan tabel ini bertujuan untuk memperkaya sumber pembelajaran dan informasi yang tersedia dalam sistem.

Secara keseluruhan, seluruh tabel dalam database saling terintegrasi untuk mendukung proses pembelajaran daring dari tahap pendaftaran hingga penerbitan sertifikat. Relasi antar tabel dirancang menggunakan foreign key untuk menjaga konsistensi dan validitas data. Struktur database ini memungkinkan sistem berjalan

•
•
•
•
•
•
•
•
•
•

• •
• •
• •
• •
• •
• •
• •
• •
• •
• •



secara efisien, mudah dikembangkan, serta mampu mendukung kebutuhan pembelajaran berbasis teknologi dan kecerdasan buatan.

4. IMPLEMENTASI SISTEM

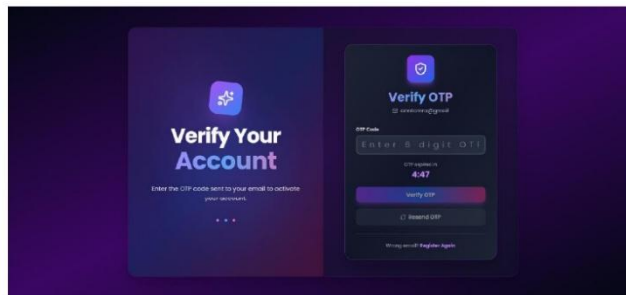
4.1 Implementasi Register



Kode registrasi pengguna berfungsi untuk memproses pembuatan akun baru. Sistem melakukan pengambilan data, validasi input, pengecekan email yang sudah

terdaftar, dan penyimpanan data pengguna ke database. Setelah proses berhasil, sistem mengirimkan respons sebagai tanda bahwa registrasi telah berhasil dilakukan.

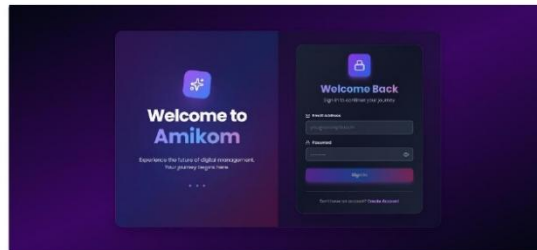
4.2 Implementasi Verifikasi OTP



Kode verifikasi OTP berfungsi untuk memvalidasi kode OTP yang dimasukkan oleh pengguna setelah proses registrasi. Sistem akan memeriksa kesesuaian kode

OTP, memastikan pengguna terdaftar, serta mengecek masa berlaku OTP. Jika OTP valid dan belum kedaluwarsa, status akun pengguna akan diperbarui menjadi terverifikasi (`is_verified = true`) sehingga pengguna dapat mengakses sistem.

4.3 Implementasi Login



```

// src/auth/login.js
import axios from 'axios';
import { login } from '../api/auth';
import { setToken } from '../utils/auth';
import { toast } from 'react-toastify';

export const loginUser = async (email, password) => {
  try {
    const response = await login(email, password);
    if (response.status === 200) {
      const { token, user } = response.data;
      setToken(token);
      toast.success('Login successful');
      return user;
    } else {
      toast.error('Invalid email or password');
      return null;
    }
  } catch (error) {
    toast.error('Something went wrong');
    return null;
  }
};

```

Kode login pengguna berfungsi untuk melakukan autentikasi akun berdasarkan email dan password yang dimasukkan. Sistem akan memvalidasi data login, memeriksa keberadaan akun, mencocokkan password dengan data yang tersimpan di database, serta memastikan akun telah terverifikasi melalui OTP. Jika seluruh

proses berhasil, sistem akan mengembalikan informasi pengguna dan memberikan akses masuk ke dalam aplikasi.

4.4 Implementasi Forgot Password

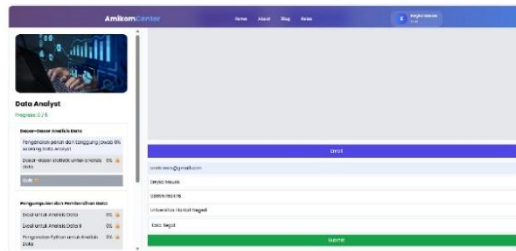


```
1
2 @user_bp.route('/forgot-password', methods=["POST"])
3 def forgot_password():
4     try:
5         data = request.get_json()
6         email = data.get("email")
7
8         if not email:
9             return jsonify({"message": "Email wajib diisi"}), 400
10
11         cur = mysql.connection.cursor()
12         cur.execute("""
13             SELECT id, is_verified
14             FROM users
15             WHERE email=%s
16             """, (email,))
17         user = cur.fetchone()
18         if not user:
19             return jsonify({"message": "Email tidak ditemukan"}), 404
20
21         # generate OTP
22         otp_code = generate_otp()
23         otp_expired = datetime.now() + timedelta(minutes=5)
24
25         cur.execute("""
26             UPDATE users
27             SET otp_code=%s, otp_expired=%s
28             WHERE email=%s
29             """, (otp_code, otp_expired, email))
30
31         mysql.connection.commit()
32
33         send_otp_email(email, otp_code)
34         cur.close()
35
36         return jsonify({
37             "message": "OTP reset password dikirim ke email"
38         }), 200
39
40 except Exception as e:
41     return jsonify({
42         "message": "Error forgot password",
43         "error": str(e)
44     }), 500
45
```

Kode forgot password berfungsi untuk membantu pengguna yang lupa kata sandi akun. Sistem akan memverifikasi keberadaan email yang terdaftar, kemudian menghasilkan kode OTP beserta batas waktu berlakunya dan menyimpannya ke

database. Setelah itu, OTP dikirim ke email pengguna untuk digunakan pada proses verifikasi dan penggantian kata sandi.

4.5 Implementasi Enroll Course



```
@course_enroll_bp.route("/enroll", methods=["POST"])
def enroll():
    try:
        data = request.get_json()

        course_id = data.get("course_id")
        user_id = data.get("user_id")
        email = data.get("email")
        name = data.get("name")
        phone = data.get("phone")
        instansi = data.get("instansi")
        city = data.get("city")

        if not course_id or not user_id:
            return jsonify({"message": "course_id dan user_id wajib"}), 400

        cur = mysql.connection.cursor()

        cur.execute(
            "SELECT * FROM enrollments WHERE course_id=%s AND user_id=%s",
            (course_id, user_id)
        )
        existing = cur.fetchone()

        if existing:
            return jsonify({"message": "User sudah terdaftar"}), 400

        cur.execute("""
            INSERT INTO enrollments
            (course_id, user_id, email, name, phone, instansi, city)
            VALUES (%s, %s, %s, %s, %s, %s, %s)
            """, (course_id, user_id, email, name, phone, instansi, city))

        cur.execute("""
            UPDATE courses
            SET total_enrolled = total_enrolled + 1
            WHERE id = %s
            """, (course_id,))

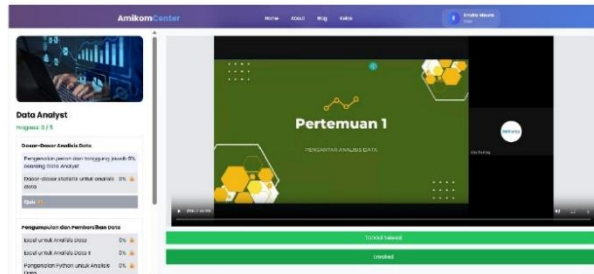
        mysql.connection.commit()
        cur.close()

        return jsonify({"message": "Enroll berhasil"})

    except Exception as e:
        return jsonify({
            "error": "gagal enroll",
            "detail": str(e)
        }), 500
```

Kode enroll course berfungsi untuk mendaftarkan pengguna ke dalam suatu kursus yang dipilih. Sistem akan memvalidasi data pendaftaran dan memeriksa apakah pengguna sudah terdaftar pada kursus tersebut untuk mencegah duplikasi data. Jika pendaftaran berhasil, data peserta akan disimpan ke tabel enrollments dan jumlah peserta pada kursus akan diperbarui secara otomatis.

4.6 Implementasi Materi Pembelajaran



```
@material_bp.route("/materials/cint:course_id", methods=["GET"])
def get_materials(course_id):

    cur = mysql.connection.cursor()

    cur.execute("""
        SELECT
            course_materials.*,
            section_course.section_title
        FROM course_materials
        LEFT JOIN section_course
        ON course_materials.section_id = section_course.id
        WHERE course_materials.course_id=%s
        ORDER BY section_course.section_order ASC
    """, (course_id,))

    columns = [col[0] for col in cur.description]
    rows = cur.fetchall()

    materials = []

    for row in rows:

        data = dict(zip(columns,row))

        if data["video_file"]:
            data["video_url"] = f"/videos/{data['video_file']}"

        if data["file_path"]:
            data["file_url"] = f"/materials/{data['file_path']}"

        materials.append(data)

    return jsonify(materials)
```

Kode get materials berfungsi untuk menampilkan daftar materi pembelajaran berdasarkan kursus yang dipilih. Sistem mengambil data materi beserta informasi section terkait dari database, kemudian mengurutkannya sesuai urutan pembelajaran yang telah ditentukan. Selain itu, sistem juga menghasilkan URL untuk file video dan dokumen sehingga materi dapat diakses langsung oleh pengguna melalui aplikasi.

4.7 Implementasi Progress Pembelajaran

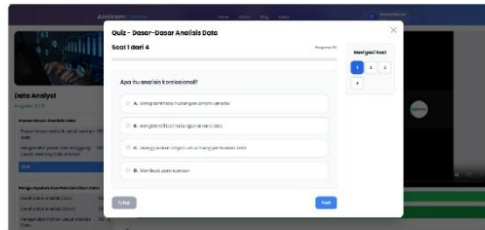


```
def save_progress(progress, method="save"):  
    try:  
        data = request.get_json()  
  
        user_id = data.get("user_id")  
        course_id = data.get("course_id")  
        material_id = data.get("material_id")  
        progress = int(data.get("progress"), 0)  
        is_completed = data.get("is_completed", False)  
        last_updated = float(data.get("last_updated", 0))  
  
        if not user_id or not material_id or not course_id:  
            return jsonify({"message": "Data tidak lengkap"}), 400  
  
        conn = mysql.connector  
        cursor = conn.cursor()  
  
        # Cek apakah data ada  
        cursor.execute(  
            "SELECT id FROM user_progress  
            WHERE user_id=%s AND material_id=%s  
            ", (user_id, material_id))  
        existing = cursor.fetchone()  
  
        if existing:  
            cursor.execute(  
                "UPDATE user_progress  
                SET progress=%s,  
                is_completed=%s,  
                last_updated=%s  
                WHERE user_id=%s AND material_id=%s  
                ", (progress,  
                    is_completed,  
                    last_updated,  
                    user_id,  
                    material_id))  
        else:  
            cursor.execute(  
                "INSERT INTO user_progress  
                (user_id, course_id, material_id, progress, is_completed, last_updated)  
                VALUES (%s, %s, %s, %s, %s, %s)  
                ", (user_id,  
                    course_id,  
                    material_id,  
                    progress,  
                    is_completed,  
                    last_updated))  
  
        conn.commit()  
        cursor.close()  
  
        return jsonify({"message": "Progress berhasil disimpan"}), 200  
    except Exception as e:  
        return jsonify({"message": "Error", "error": str(e)}), 500
```

Kode penyimpanan progres pembelajaran berfungsi untuk mencatat perkembangan belajar pengguna pada setiap materi yang dipelajari. Sistem akan memeriksa apakah data progres sudah tersedia, kemudian melakukan pembaruan data atau menambahkan data baru sesuai kondisi yang ditemukan. Informasi yang

disimpan meliputi persentase progres, status penyelesaian materi, dan posisi terakhir pengguna saat mengakses materi pembelajaran.

4.8 Implementasi Quiz



```
@quiz_bp.route("/quiz/student/<int:section_id>", methods=["GET"])
def get_quiz_student(section_id):

    try:

        cur = mysql.connection.cursor()

        cur.execute("""
            SELECT *
            FROM quizzes
            WHERE section_id=%s
            AND status='active'
            AND is_verified=1
            ORDER BY id DESC
            """, (section_id,))

        data = cur.fetchall()

        columns = [col[0] for col in cur.description]

        result = []

        for row in data:
            result.append(
                dict(zip(columns, row))
            )

        cur.close()

        return jsonify(result)

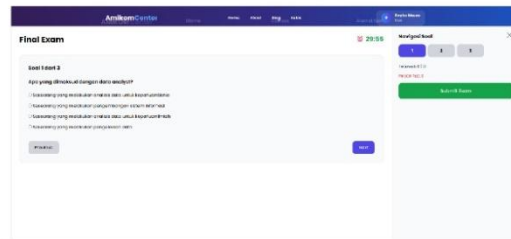
    except Exception as e:

        print("GET QUIZ STUDENT ERROR:", e)

        return jsonify([])
```

Kode menampilkan kuis untuk peserta berfungsi untuk mengambil daftar soal kuis yang tersedia pada suatu section pembelajaran. Sistem hanya menampilkan soal yang berstatus aktif dan telah diverifikasi agar peserta memperoleh soal yang valid. Data kuis yang berhasil diambil kemudian dikirim dalam format JSON sehingga dapat ditampilkan pada halaman pembelajaran.

4.9 Implementasi Final Exam



```
@final_exam_bp.route("/final-exams/<int:course_id>", methods=["GET"])
def get_final_exams(course_id):
    try:
        tentor_id = request.args.get("tentor_id")
        cur = mysql.connection.cursor()

        if tentor_id:
            cur.execute("""
                SELECT
                    fe.id, fe.course_id, fe.section_id, fe.title,
                    fe.description, fe.duration minutes,
                    fe.passing_score, fe.max_attempt,
                    fe.is_active, fe.created_by,
                    fe.created_at, sc.section_title
                FROM final_exams fe
                LEFT JOIN section_course sc
                ON fe.section_id = sc.id
                AND fe.course_id = sc.course_id
                WHERE fe.course_id=%s AND fe.created_by=%s
                ORDER BY fe.id DESC
            """, (course_id, tentor_id))
        else:
            cur.execute("""
                SELECT
                    fe.id, fe.course_id, fe.section_id, fe.title,
                    fe.description, fe.duration minutes,
                    fe.passing_score, fe.max_attempt,
                    fe.is_active, fe.created_by,
                    fe.created_at, sc.section_title
                FROM final_exams fe
                LEFT JOIN section_course sc
                ON fe.section_id = sc.id
                AND fe.course_id = sc.course_id
                WHERE fe.course_id=%s
                ORDER BY fe.id DESC
            """, (course_id,))

        rows = cur.fetchall()
        cur.close()

        return jsonify([
            {
                "id": r[0],
                "course_id": r[1],
                "section_id": r[2],
                "title": r[3],
                "description": r[4],
                "duration_minutes": r[5],
                "passing_score": r[6],
                "max_attempt": r[7],
                "is_active": r[8],
                "created_by": r[9],
                "created_at": str(r[10]),
                "section_title": r[11] if r[11] else ""
            } for r in rows])

    except Exception as e:
        return jsonify({"error": str(e)}), 500
```

Kode menampilkan ujian akhir berfungsi untuk mengambil daftar ujian akhir yang tersedia pada suatu kursus. Sistem dapat menampilkan seluruh ujian atau memfilter ujian berdasarkan pengajar (tutor) yang membuatnya. Data ujian yang diperoleh kemudian dikirim dalam format JSON beserta informasi section terkait sehingga dapat ditampilkan pada halaman pembelajaran atau evaluasi.

4.10 Implementasi Sertifikat



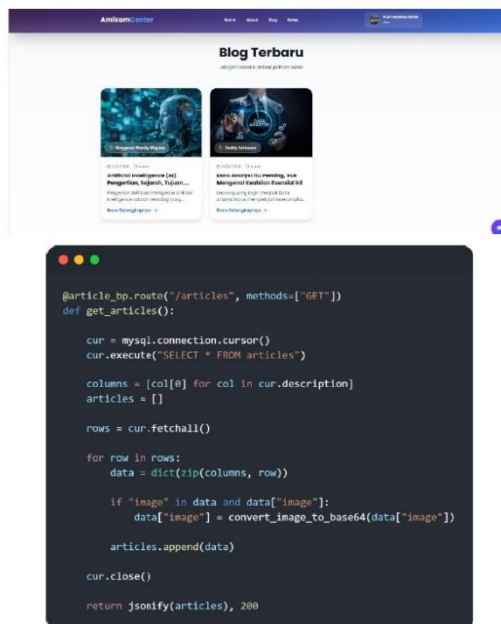
```
certificate_data = None
if passed == 1:
    try:
        cur.execute("""
            SELECT
                certificate_number,
                file_path
            FROM certificates
            WHERE user_id=?
            AND exam_id=?
            """, (
                user_id,
                exam_id
            ))
        cert = cur.fetchone()

        if cert:
            certificate_data = {
                "certificate_number": cert[0],
                "file_path": cert[1]
            }
        else:
            cur.execute("""
                SELECT
                    u.full_name,
                    c.course_name
                FROM users u,
                FINAL_NAMES fn
                ON u.full_name =
                fn.full_name
                JOIN courses c
                ON c.id = fn.course_id
                WHERE u.id=?
            """, (
                user_id,
                user_id
            ))
            row = cur.fetchone()

            user_name = {
                row[0]:
                if row[0] != "User"
            }
            course_name = {
                row[1]:
                if row[1] != "Course"
            }
            cert_number = {
                "cert_id": user_id + "-" +
                str(datetime.now().timestamp())
            }
            folder = "certificates"
            if not os.path.exists(folder):
                os.makedirs(folder)
            file_name = f"{cert_number}.pdf"
            file_path = os.path.join(
                folder,
                file_name
            )
            generate_certificate(
                file_path=file_path,
                user_name=user_name,
                course_name=course_name,
                exam_name=
                cert_number=cert_number
            )
```

Kode pembuatan sertifikat otomatis berfungsi untuk menghasilkan sertifikat bagi peserta yang dinyatakan lulus ujian akhir. Sistem akan memeriksa terlebih dahulu apakah sertifikat sudah pernah dibuat, kemudian mengambil data pengguna dan kursus untuk digunakan dalam sertifikat. Jika sertifikat belum tersedia, sistem akan membuat nomor sertifikat unik, menghasilkan file sertifikat dalam format PDF, dan menyimpannya untuk dapat diakses oleh peserta.

4.11 Implementasi Artikel



Kode `get_articles` berfungsi untuk mengambil seluruh data artikel yang tersimpan dalam database. Sistem akan mengonversi gambar artikel ke format Base64 agar dapat ditampilkan pada aplikasi bersama informasi artikel lainnya. Seluruh data kemudian dikirim dalam format JSON sehingga dapat ditampilkan sebagai daftar artikel kepada pengguna.

4.12 Implementasi Chatbot



```
app.route('/api/chatbot', methods=['POST'])
def chatbot():
    try:
        data = request.get_json()

        if not data:
            return jsonify({
                "error": "Request body required"
            }), 400

        message = {
            data.get("message")
            or data.get("chatinput")
        }

        if not message:
            return jsonify({
                "error": "message required"
            }), 400

        webhook_url = ("https://mn.gitstraining.com/webhook/chatbotto")

        payload = {
            "chatinput": message,
            "sessionId": data.get(
                "sessionId",
                "user-001"
            )
        }

        response = requests.post(
            webhook_url,
            json=payload
        )

        response.raise_for_status()

        return jsonify(
            response.json()
        ), 200

    except requests.exceptions.RequestException as e:
        return jsonify({
            "error": "Failed to connect to api",
            "detail": str(e)
        }), 500

    except Exception as e:
        return jsonify({
            "error": "Internal server error",
            "detail": str(e)
        }), 500
```



Kode layanan chatbot berfungsi untuk menerima pertanyaan atau pesan dari pengguna dan meneruskannya ke layanan chatbot melalui webhook n8n. Sistem melakukan validasi input, mengirimkan pesan beserta informasi sesi pengguna, kemudian menerima dan mengembalikan respons chatbot ke aplikasi. Dengan mekanisme ini, pengguna dapat berinteraksi dengan chatbot secara real-time untuk memperoleh informasi atau bantuan terkait pembelajaran.

•
•
•
•
•
•
•
•
•
•

• •
• •
• •
• •
• •
• •
• •
• •
• •
• •
• •



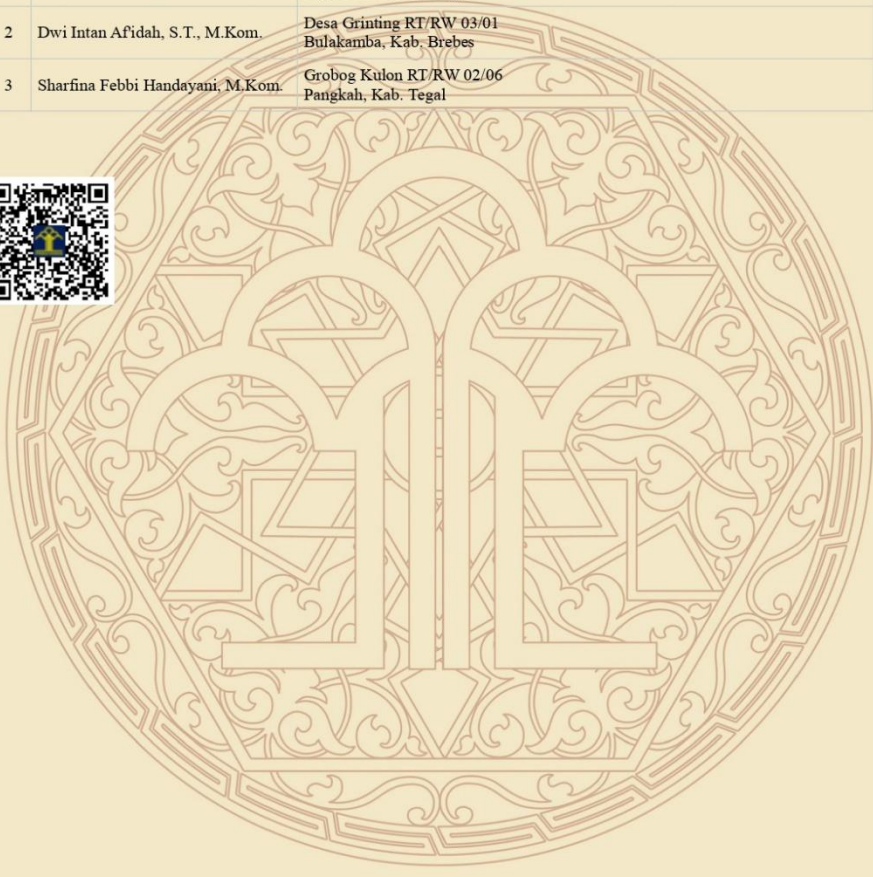
Lampiran 6 Sertifikat HKI

 REPUBLIK INDONESIA KEMENTERIAN HUKUM	
SURAT PENCATATAN CIPTAAN	
Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:	
Nomor dan tanggal permohonan	: EC002026083089, 9 Juni 2026
Pencipta	
Nama	: Erryka Maura Adysti Anantacia, Dwi Intan Afidah, S.T., M.Kom. dkk
Alamat	: Jl. Seram RT/RW 010/010, Kelurahan Mintaragen, Tegal Timur, Kota Tegal, Jawa Tengah, 52121
Kewarganegaraan	: Indonesia
Pemegang Hak Cipta	
Nama	: Unit Penelitian Abdimas Universitas Harkat Negeri
Alamat	: Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana 52142, Margadana, Kota Tegal, Jawa Tengah, 52142
Kewarganegaraan	: Indonesia
Jenis Ciptaan	: Program Komputer
Judul Ciptaan	: Aplikasi Web Layanan Pelatihan Amikom Center Yogyakarta dengan Integrasi Chatbot N8N.
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 9 Juni 2026, di Kota Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor Pencatatan	: 001274085
adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.	
	a.n. MENTERI HUKUM DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL u.b Direktur Hak Cipta dan Desain Industri  Agung Damarsasongko,SH.,MH. NIP. 196912261994031001
	Disclaimer: 1. Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan. 2. Surat Pencatatan ini telah disegel secara elektronik menggunakan segel elektronik yang diterbitkan oleh Balai Besar Sertifikasi Elektronik, Badan Siber dan Sandi Negara. 3. Surat Pencatatan ini dapat dibuktikan keasliannya dengan memindai kode QR pada dokumen ini dan informasi akan ditampilkan dalam browser.



LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Erryka Maura Adysti Anantacia	Jl. Seram RT/RW 010/010, Kelurahan Mintaragen Tegal Timur, Kota Tegal
2	Dwi Intan Afidah, S.T., M.Kom.	Desa Grinting RT/RW 03/01 Bulakamba, Kab. Brebes
3	Sharfina Febbi Handayani, M.Kom.	Grobog Kulon RT/RW 02/06 Pangkajene, Kab. Tegal





**Balai Besar
Sertifikasi
Elektronik**

Disclaimer:

1. Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.
2. Surat Pencatatan ini telah disegel secara elektronik menggunakan segel elektronik yang diterbitkan oleh Balai Besar Sertifikasi Elektronik, Badan Siber dan Sandi Negara.
3. Surat Pencatatan ini dapat dibuktikan keasliannya dengan memindai kode QR pada dokumen ini dan informasi akan ditampilkan dalam browser.

Lampiran 7 Lembar Bimbingan

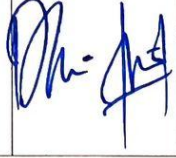


D IV TEKNIK INFORMATIKA
UNIVERSITAS HARKAT NEGERI

LEMBAR BIMBINGAN SKRIPSI

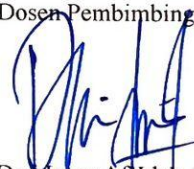
Nama : Erryka Maura Adysti Anantacia
NIM : 22090140
No. Ponsel : 089655453475
Judul TA : **PENGEMBANGAN APLIKASI WEB LAYANAN PELATIHAN AMIKOM CENTER YOGYAKARTA DENGAN INTEGRASI CHATBOT N8N**
Dosen Pembimbing I : Dwi Intan AfIdah, S.T., M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1.	9/3 - 26	Konsep aplikasi	Perlu referensi dari aplikasi sejenis untuk kematangan proses bisnis	
2.	6/4 - 26	Aplikasi	Lanjutkan aplikasi agar bisa dilakukan simulasi untuk satu course	
3.	13/4 - 26	Aplikasi	Lanjutkan dapat melakukan exam dengan standar ujian	

4.	21/4 - 26	Aplikasi	Langutkan agar dapat melakukan Simulasi Exam	
5.	5/5 - 26	Aplikasi	Adanya history di student role - pengujian	
6.	12/5 - 26	Aplikasi	Fitur tentor membuat bank soal → API AI - Rekomendasi dengan AI untuk hasil evaluasi course	
7.	26/5 - 26	Aplikasi	Pengujian fungsional - OTP, forgot password - user manual dokumen teknikal	
8.	8/6 - 26	Hak cipta	Langutkan pengajuan - Langutkan pengajuan laporan	
9.	22/6 - 26	Laporan	perbaiki sesi catat	

10.	24/6-26	Laporan	ACC	
-----	---------	---------	-----	---

Tegal, 14 Juni 2021
Dosen Pembimbing I



Dwi Intan Afidah, S.T., M.Kom.
NIPY. 11.020.470

LEMBAR BIMBINGAN SKRIPSI

Nama : Erryka Maura Adysti Anantacia
NIM : 22090140
No. Ponsel : 089655453475
Judul Skripsi : PENGEMBANGAN APLIKASI WEB
LAYANAN PELATIHAN AMIKOM CENTER
YOGYAKARTA DENGAN INTEGRASI
CHATBOT N8N
Dosen Pembimbing II : Sharfina Febbi Handayani, M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1.	6/5 - 2026	Proses bisnis dan aplikasi	- Bagian exam dan quiz ditambah studi kasus - Dibuat bank soal	Sharfina
2.	14/5 - 2026	Spesifikasi aplikasi	Membuat PRD	Sharfina
3.	20/5 - 2026	Aplikasi	- Forgot password admin dan tentor - Disable button jika sudah max attemp	Sharfina
4.	3/6 - 2026	HKI	Pengajuan HKI	Sharfina

5.	04/06/2026	HKI	Pengecekan Manual book dan dokumen teknikal	Sharifi
6.	08/06/2026	Laporan BAB I	Pengecekan Latar belakang, Rumusan Masalah, Tujuan, Manfaat dan Data Penelitian » Cek Catatan / komentar	Sharifi
7.	12/06/2026	Laporan BAB I	Pengecekan Tinjauan Pustaka dan Data Penelitian → lanjut susun BAB 2	Sharifi
8.	22/06/2026	Laporan BAB II	Pengecekan Produk dan Pengujian VAT » Ditambahkan hasil API testing	Sharifi
9.	23/06/2026	Laporan BAB III	Diselesaikan dengan HKI	Sharifi
10	24/06/2026	ACC Aplikasi ACC Laporan	ACC Aplikasi ACC Laporan	Sharifi

--	--	--	--	--

Tegal, 24 Juni 2026
Dosen Pembimbing II

Sharfina

Sharfina Febbi Handayani, M.Kom.
NIPY. 08.020.451