

RANCANG BANGUN *FRONTEND* SISTEM INFORMASI CAPAIAN PEMBELAJARAN LULUSAN (CPL) BERBASIS WEB MENGGUNAKAN METODE AGILE KANBAN

Ivan Simangunsong^{1*}, Wahyu Eko Sulistiono², Puput Budi Wintoro³

^{1,2,3} Fakultas Teknik Universitas Lampung, Jalan Prof. Dr Jl. Prof. Dr. Ir. Sumantri Brojonegoro No.1, Kota Bandar Lampung, Lampung 35141

Keywords:

Capaian Pembelajaran Lulusan, *Frontend*, *React.js*, *Agile Kanban*, *Black Box Testing*

Correspondent Email:

ivansimangunsong17@gmail.com

Abstrak. Pendidikan tinggi memiliki peran penting dalam menghasilkan lulusan yang berkualitas dan kompetitif, sehingga setiap program studi perlu memastikan capaian pembelajaran lulusannya (CPL) sesuai standar yang ditetapkan. Saat ini, pengolahan dan penyajian data CPL di Universitas Lampung belum dilakukan secara sistematis, sehingga diperlukan sistem yang mampu mendukung pencatatan, perhitungan, dan penyajian informasi CPL secara terstruktur. Penelitian ini bertujuan merancang dan membangun *frontend* Sistem Informasi CPL berbasis *web* menggunakan *React.js* dengan metode *Agile Kanban*. Tahapan penelitian meliputi *studi literatur*, analisis kebutuhan *fungsional* dan *nonfungsional*, perancangan arsitektur *frontend*, serta pengembangan menggunakan papan *Kanban* yang terdiri atas kolom *Backlog/To Do*, *In Progress*, *Testing*, dan *Done*. Hasil penelitian menunjukkan bahwa *frontend* sistem berhasil dibangun dengan struktur *modular*, *layer* tampilan terpisah, *hooks*, *service*, dan *API* yang mendukung pengelolaan alur data secara efektif. Pengujian fungsional menggunakan metode *black box testing* menunjukkan tingkat keberhasilan sebesar 78,34% yang menandakan sebagian besar *test case* berjalan sesuai ekspektasi dan fungsi utama *frontend* dapat beroperasi dengan baik. Penelitian ini menghasilkan artefak teknis berupa *frontend* Sistem Informasi CPL yang dapat dijadikan referensi atau *prototype* untuk pengembangan sistem lebih lanjut.



Copyright © [JITET](#) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

Abstract. Higher education plays an important role in producing qualified and competitive graduates, making it essential for each study program to ensure that its graduates' learning outcomes (Capaian Pembelajaran Lulusan / CPL) align with established standards. Currently, the processing and presentation of CPL data at the University of Lampung are not carried out systematically, highlighting the need for a system that can support structured recording, calculation, and presentation of CPL information. This study aims to design and develop a web-based frontend for the CPL Information System using *React.js* and the *Agile Kanban* method. The research stages include a literature review, analysis of functional and non-functional requirements, frontend architecture design, and development using a Kanban board consisting of *Backlog/To Do*, *In Progress*, *Testing*, and *Done* columns. The results show that the frontend system was successfully built with a modular structure, separate view layers, hooks, services, and APIs, which effectively support data flow management. Functional testing using the black-box testing method yielded a success rate of 78.34%, indicating that the majority of test cases ran as expected and the main frontend functions operated properly. This study produces a technical artifact in the form of a CPL information system frontend that can serve as a reference or prototype for further system development.

1. PENDAHULUAN

Perkembangan pendidikan tinggi menuntut setiap program studi mampu menghasilkan lulusan yang berkualitas, adaptif, dan berdaya saing melalui ketercapaian capaian pembelajaran lulusan (CPL) yang mencakup aspek pengetahuan, keterampilan, dan sikap. CPL menjadi landasan dalam pengembangan kurikulum, evaluasi proses pembelajaran, serta penjaminan mutu pendidikan tinggi, termasuk dalam mekanisme akreditasi berbasis *outcome-based accreditation*[1]. Oleh karena itu, pengelolaan data CPL yang sistematis, terstruktur, dan mudah diakses merupakan kebutuhan penting bagi program studi dalam mendukung pengambilan keputusan akademik berbasis data.

Namun demikian, pada praktiknya pengolahan dan penyajian data CPL di tingkat program studi masih belum didukung oleh sistem informasi yang memadai. Proses pencatatan, perhitungan, serta penyajian informasi CPL sering kali dilakukan secara terpisah dan kurang terintegrasi sehingga pemantauan ketercapaian CPL belum optimal.

Perkembangan teknologi *frontend* berbasis *JavaScript framework*, seperti *React.js*, memberikan peluang untuk membangun antarmuka sistem informasi yang interaktif, efisien, dan mudah dipelihara melalui pendekatan berbasis komponen dan *Virtual DOM*[2]. Selain itu, penerapan metode pengembangan perangkat lunak *Agile Kanban* memungkinkan proses pembangunan dilakukan secara iteratif dan adaptif terhadap kebutuhan pengguna. Untuk memastikan kesesuaian fungsi sistem dengan spesifikasi yang dirancang, pengujian *black-box* digunakan guna memvalidasi keluaran sistem berdasarkan masukan yang diberikan tanpa meninjau struktur *internal* perangkat lunak[3].

Oleh karena itu, penelitian ini bertujuan menganalisis kebutuhan *fungsi* dan *nonfungsi*, merancang arsitektur serta struktur komponen *frontend*, dan membangun *frontend* sistem informasi CPL berbasis web menggunakan metode *Agile Kanban*. Hasil penelitian diharapkan menghasilkan *prototipe frontend* dan dokumentasi teknis yang dapat menjadi referensi pengembangan sistem informasi CPL pada lingkungan pendidikan tinggi.

2. TINJAUAN PUSTAKA

2.1. Capaian Pembelajaran Lulusan

Capaian Pembelajaran Lulusan (CPL) merupakan pernyataan kemampuan yang diharapkan dimiliki lulusan setelah menyelesaikan proses pembelajaran pada suatu program studi, mencakup aspek pengetahuan, keterampilan, serta sikap dan tata nilai sebagai acuan utama penyelenggaraan pendidikan tinggi[4]. Dalam implementasinya, CPL dijabarkan ke dalam Capaian Pembelajaran Mata Kuliah (CPMK) yang berfungsi sebagai indikator ketercapaian CPL melalui hasil penilaian pada setiap mata kuliah, sehingga hubungan keduanya bersifat hierarkis dan terintegrasi[5].

Perhitungan Capaian Pembelajaran Lulusan (CPL) didasarkan pada pencapaian Capaian Pembelajaran Mata Kuliah (CPMK) dari mata kuliah yang ditempuh mahasiswa melalui pengolahan nilai tugas, Ujian Tengah Semester (UTS), dan Ujian Akhir Semester (UAS). Proses ini meliputi penentuan bobot kontribusi setiap CPMK terhadap CPL, pengumpulan data nilai mahasiswa, perhitungan nilai CPMK, serta akumulasi nilai CPMK yang telah dikalikan dengan bobotnya. Secara matematis, nilai CPL dihitung menggunakan rata-rata tertimbang CPMK, yaitu menjumlahkan hasil perkalian nilai CPMK dengan bobot masing-masing kemudian membaginya dengan total bobot, sehingga CPMK dengan bobot lebih besar memberikan pengaruh yang lebih signifikan terhadap nilai CPL. Secara matematis, perhitungan nilai CPL dapat dirumuskan sebagai berikut[4]:

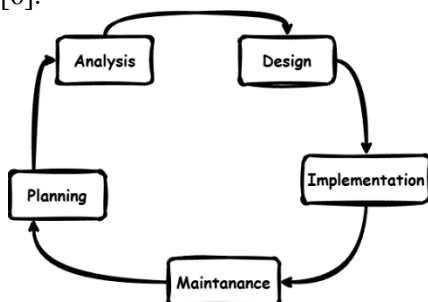
$$\text{Nilai CPL} = \frac{(\text{Nilai CPMK} \times \text{bobot}) + (\dots) + (\dots)}{\text{Jumlah keseluruhan bobot}}$$

Gambar 1 Rumus Perhitungan Nilai CPL

2.2. Software Development Life Cycle

Software Development Life Cycle (SDLC) merupakan serangkaian tahapan yang terstruktur dan sistematis dalam proses pengembangan maupun modifikasi sistem perangkat lunak. Proses ini melibatkan penggunaan berbagai model dan metodologi yang dirancang untuk membantu pengembang dalam merancang, membangun, menguji, serta

menerapkan perangkat lunak secara efisien dan efektif[6].



Gambar 2 Software Development Life Cycle

2.3. Agile

Agile merupakan sekumpulan metode dan pendekatan dalam rekayasa perangkat lunak yang bertujuan meningkatkan efektivitas kerja, kualitas pengambilan keputusan, serta efisiensi proses pengembangan melalui praktik iteratif dan adaptif.[7].

Salah satu metode yang umum digunakan adalah *Kanban*. Dalam implementasi *Agile*, alur kerja *Kanban* umumnya meliputi tahap *backlog*, *to do*, *in progress*, *feedback*, dan *done*, yang memungkinkan tim mengelola prioritas pekerjaan, memantau kemajuan, menerima umpan balik, serta memastikan setiap tugas diselesaikan sesuai standar yang ditetapkan[8].

2.4. Unified Modeling Language

UML (*Unified Modeling Language*) merupakan salah satu standar bahasa pemodelan yang banyak digunakan di industri untuk mendefinisikan kebutuhan sistem, melakukan analisis dan desain, serta menggambarkan arsitektur sistem dalam pengembangan perangkat lunak berbasis pemrograman berorientasi objek[6].

2.5. Use Case Diagram

Use Case merupakan salah satu cara untuk menangkap kebutuhan sistem dengan mendefinisikan apa saja yang seharusnya dilakukan oleh sistem. Dalam konsep *Use Case* terdapat tiga elemen utama, yaitu *aktor*, *use case*, dan subjek[9].

2.6. Activity Diagram

Activity Diagram merupakan diagram yang digunakan untuk memodelkan aktivitas atau alur kerja (*workflow*) dalam suatu sistem atau organisasi. Diagram ini menggambarkan rangkaian aktivitas yang terjadi dalam suatu proses, baik yang berasal dari dalam sistem maupun dari luar sistem[9].

2.7. Sequence Diagram

Sequence Diagram merupakan jenis *interaction diagram* yang paling umum digunakan dalam UML. Diagram ini berfokus pada pertukaran pesan antar objek atau *lifeline* dengan menekankan urutan pesan yang terjadi[9].

2.8. BlackBox Testing

Blackbox Testing merupakan metode pengujian perangkat lunak yang berfokus pada fungsi eksternal sistem tanpa memperhatikan kode program. Penguji hanya menggunakan spesifikasi *fungsi* untuk menentukan input dan *output* yang diharapkan[3].

Analisis hasil pengujian dilakukan dengan menghitung tingkat keberhasilan pengujian menggunakan metrik standar pengujian perangkat lunak, yaitu persentase *test case* yang berhasil (*test case pass rate*). Perhitungan persentase keberhasilan pengujian dilakukan menggunakan rumus sebagai berikut[10]:

$$Test\ Case\ Pass = \left(\frac{Test\ Case\ Passed}{Total\ Test\ Case} \right) \times 100\%$$

Gambar 3 Rumus pengujian Test Case Pass

2.9. React.js

React.js adalah *library* JavaScript yang digunakan untuk membangun antarmuka pengguna (*user interface/UI*) yang interaktif dan efisien[2].

2.10. Tailwind

Tailwind CSS adalah sebuah *utility-first CSS framework* yang dikembangkan untuk mempermudah dan mempercepat proses pembuatan antarmuka web modern[11].

2.11. Penelitian Terdahulu

Dalam rangka mendukung pelaksanaan penelitian serta menghindari plagiarisme, dilakukan studi literatur terhadap beberapa penelitian terdahulu yang relevan dengan topik evaluasi Capaian Pembelajaran Lulusan (CPL). Penelitian oleh Dony Tanu Wijaya, Istiqomah Sumadikarta, dan Bosar Panjaitan (2023) berjudul *Analisa dan Perancangan Aplikasi Evaluasi Capaian Pembelajaran Lulusan* mengembangkan sistem menggunakan metode waterfall dengan pengujian melalui test case dan User Acceptance Testing (UAT) [12].

Selanjutnya, penelitian oleh Nada S. Adilah, Lillyan Hadjaratie, dan Rampi Yusuf (2022) berjudul *Pengembangan Sistem Informasi*

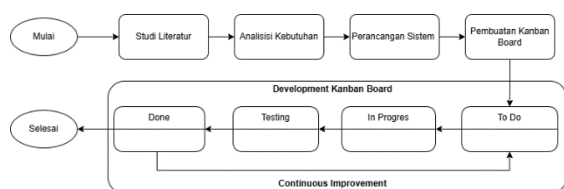
Rencana Pembelajaran Semester dan Evaluasi Capaian Pembelajaran Lulusan Berbasis Progressive Web App dilatarbelakangi oleh penggunaan Microsoft Excel dalam penyusunan RPS dan evaluasi CPL, sehingga dikembangkan sistem berbasis web menggunakan metode prototype [4].

Penelitian oleh Puspita Dewi Cahyawardani dan Hendrik berjudul *Pengembangan Sistem Informasi Evaluasi Capaian Pembelajaran Lulusan Jurusan Informatika FTI UII* membangun aplikasi berbasis web menggunakan PHP, framework Laravel, serta plugin jExcel dengan model pengembangan Agile metode Scrum dan pengujian melalui skenario pengujian serta UAT untuk mengatasi belum tersedianya sistem otomatis dalam perhitungan dan rekapitulasi nilai [13].

Selain itu, penelitian oleh Ezra Driasyawan dan Ir. Ratnanto Fitriadi, S.T., M.T. berjudul *Perancangan Sistem Penilaian Praktikum Industri Kreatif Berbasis Outcome Based Education (OBE)* berfokus pada pengembangan sistem penilaian terstruktur berbasis CPL dan CPMK melalui pendekatan OBE dengan perancangan rubrik penilaian yang terstandar guna menghasilkan penilaian yang objektif dan konsisten [14].

3. METODE PENELITIAN

Metodologi penelitian yang digunakan pada penelitian rancang bangun *frontend* sistem informasi capaian pembelajaran lulusan (cpl) berbasis web menggunakan metode *Agile Kanban*. Adapun tahapan yang dilakukan pada penelitian ditunjukkan pada Gambar 5.



Gambar 4 Metodologi Penelitian

3.1. Studi Literatur

Tahapan ini bertujuan untuk memperoleh pemahaman yang mendalam mengenai topik yang diteliti. Peneliti melakukan penelusuran terhadap berbagai sumber ilmiah, seperti jurnal, laporan penelitian, dan dokumen kebijakan yang berkaitan dengan Capaian Pembelajaran

Lulusan (CPL), serta implementasi sistem informasi dalam konteks pendidikan tinggi.

3.2. Analisis Kebutuhan Sistem

Setelah memperoleh landasan teori yang kuat, penelitian dilanjutkan ke tahap analisis kebutuhan sistem. Pada tahap ini dilakukan identifikasi terhadap kebutuhan pengguna sistem, baik dari sisi *funksional* maupun *nonfunksional*. Analisis kebutuhan dilakukan melalui observasi terhadap proses bisnis yang berjalan, studi dokumen, serta wawancara informal dengan pihak-pihak yang berkepentingan, seperti dosen dan pengelola sistem akademik

3.3. Perancangan Sistem

Tahap perancangan sistem merupakan proses transformasi dari hasil analisis kebutuhan menjadi desain teknis sistem yang akan diimplementasikan. Perancangan ini bertujuan untuk memastikan bahwa sistem informasi yang dibangun mampu memenuhi kebutuhan pengguna secara efisien, terstruktur, dan mudah dikembangkan di masa mendatang.

3.4. Kanban Board

Setelah desain sistem selesai, peneliti memasuki tahap pembuatan *Kanban board*. *Kanban board* memetakan setiap tugas ke dalam status pengerjaan tertentu, sehingga memungkinkan pengaturan prioritas, pelacakan progres, serta pencegahan penumpukan pekerjaan.

Pada *Kanban board* terdapat empat kolom utama, yaitu *To Do*, *In Progress*, *Testing*, dan *Done*. Setiap fitur atau modul yang akan dikembangkan pertama kali ditempatkan pada kolom *To Do*.

3.5. Development

Setelah *Kanban board* disusun sebagai alat manajemen tugas, tahap pengembangan (*development*) dilaksanakan secara terstruktur melalui alur kerja yang telah ditetapkan pada papan tersebut. Setiap *backlog* fitur yang telah diidentifikasi dipindahkan secara bertahap dari kolom *To Do* ke *In Progress* sesuai dengan prioritas pengerjaan.

4. HASIL DAN PEMBAHASAN

4.1. Studi Literatur

Studi literatur memberikan landasan konseptual dalam pengembangan sistem informasi penghitungan Capaian Pembelajaran Lulusan (CPL) dengan mengacu pada Standar

Nasional Pendidikan Tinggi dan pendekatan *Outcome-Based Education (OBE)*, yang menekankan pengukuran ketercapaian kompetensi mahasiswa melalui keterkaitan antara CPL dan CPMK. Kajian ini menegaskan bahwa sistem perlu mampu memetakan, mengukur, dan menyajikan data capaian secara terstruktur guna mendukung evaluasi mutu pendidikan.

4.2. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem berfokus pada pemenuhan fungsi dan kualitas layanan untuk mendukung perhitungan Capaian Pembelajaran Lulusan (CPL) secara sistematis dan terukur di tingkat program studi. Proses ini meliputi identifikasi kebutuhan fungsional dan nonfungsional serta penentuan aktor sistem sebagai dasar perancangan dan implementasi sistem informasi. Adapun setelah dilakukannya identifikasi terhadap tujuan pengembangan aplikasi didapatkan kebutuhan *fungsional* terdapat pada tabel 1:

Tabel 1 Kebutuhan *Fungsional*

| Role | Kebutuhan |
|----------------------|--|
| Admin Universitas | 1. Dapat melakukan <i>login</i> . 2. Dapat mengelola data prodi. 3. Dapat mengelola akun admin universitas dan admin prodi. 4. Dapat melihat hasil perhitungan. 5. Dapat melakukan <i>logout</i> . |
| Admin Program Studi | 1. Dapat melakukan <i>login</i> . 2. Dapat mengelola akun kepala program studi dan dosen. 3. Dapat mengelola data mahasiswa. 4. Dapat mengelola data mata kuliah. 5. Dapat mengelola data kelas matakuliah. 6. Dapat mengelola data kelas mahasiswa. 7. Dapat mengelola data CPL. 8. Dapat mengelola data CPMK. 9. Dapat mengelola pemetaan CPL – CPMK. 10. Dapat mengelola penilaian. 11. Dapat melihat hasil perhitungan. 12. Dapat melakukan <i>logout</i> . |
| Kepala Program Studi | 1. Dapat melakukan <i>login</i> . 2. Dapat melihat data CPL dan CPMK. 3. Dapat melihat pemetaan. 4. Dapat melihat hasil perhitungan. 5. Dapat melakukan <i>logout</i> . |
| Dosen | 1. Dapat melakukan <i>login</i> . 2. Dapat mengelola penilaian. 3. Dapat mengelola data nilai mahasiswa. |

| |
|---|
| 4. Dapat melihat hasil perhitungan. 5. Dapat melakukan <i>logout</i> . |
|---|

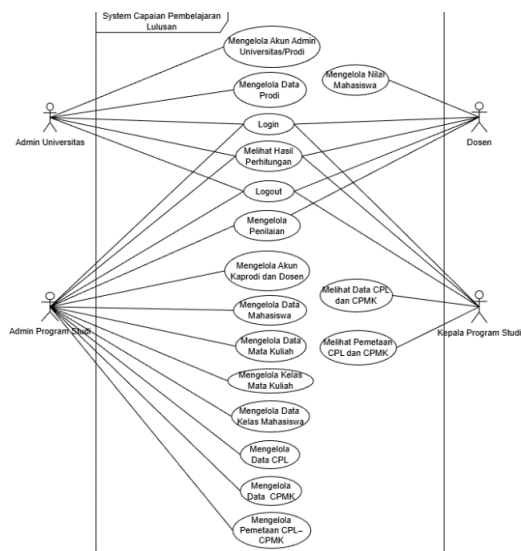
Kemudian, setelah dilakukannya identifikasi terhadap tujuan pengembangan aplikasi didapatkan kebutuhan non fungsional sistem yaitu:

1. *Usability*: Antarmuka pengguna dirancang agar mudah dipahami dan digunakan oleh pengguna sesuai perannya, dengan navigasi dan alur interaksi yang jelas.
2. *Performance*: Setiap halaman web harus dimuat dalam waktu kurang dari 5 detik dalam kondisi jaringan normal.
3. *Security*: Hanya pengguna yang berwenang dapat mengakses dan mengelola data sesuai hak aksesnya.
4. *Compatibility*: Sistem harus dapat berjalan dengan baik di berbagai browser modern (*Chrome, Firefox, Edge*).

4.3. Perancangan Sistem

Dalam tahap ini dilakukan pembuatan rancangan *visual* dan teknis yang meliputi diagram use case, diagram aktivitas (*activity diagram*), serta arsitektur sistem website secara keseluruhan. Dalam tahap ini terdapat beberapa diagram yang akan menjadi acuan selama pengembangan yaitu:

1. Use Case Diagram



Gambar 5 Use Case Diagram Sistem

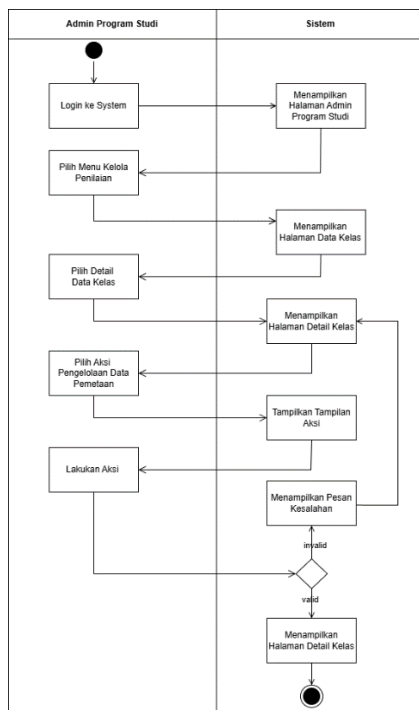
Gambar 5 merupakan diagram yang menggambarkan hubungan antara para aktor dan fungsi-fungsi utama dalam Sistem Capaian Pembelajaran Lulusan.

2. Activity Diagram

Berikut disajikan beberapa *activity diagram* yang merepresentasikan alur proses utama dalam sistem sehingga memberikan gambaran menyeluruh mengenai mekanisme kerja serta interaksi antara aktor dan sistem.

1. Activity Kelola Pemetaan

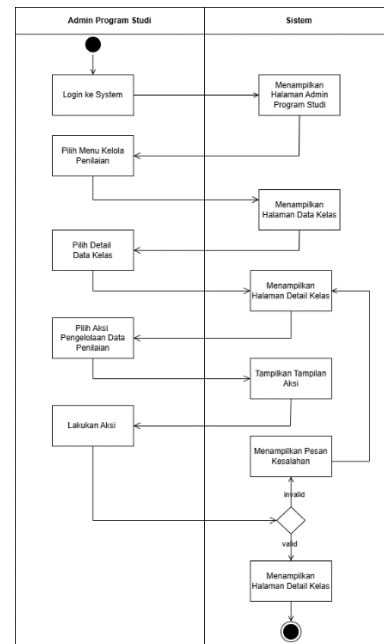
Gambar 6 memperlihatkan alur aktivitas fitur pengelolaan data pemetaan pada suatu kelas. Pada halaman detail kelas, Admin memilih aksi pengelolaan data pemetaan, dan sistem menampilkan tampilan sesuai dengan aksi tersebut.



Gambar 6 Activity Kelola Pemetaan

2. Activity Kelola Penilaian

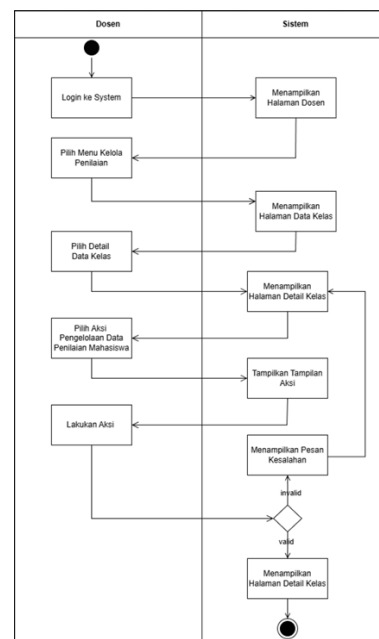
Gambar 7 memperlihatkan alur aktivitas fitur pengelolaan data penilaian (*Assessment*) pada halaman detail kelas. Sistem menampilkan daftar kelas, lalu Admin memilih salah satu kelas untuk melihat detailnya. Pada halaman detail kelas, Admin memilih aksi pengelolaan data penilaian, dan sistem menampilkan tampilan sesuai dengan aksi yang dipilih.



Gambar 7 Activity Kelola Penilaian

3. Activity Penilaian Mahasiswa

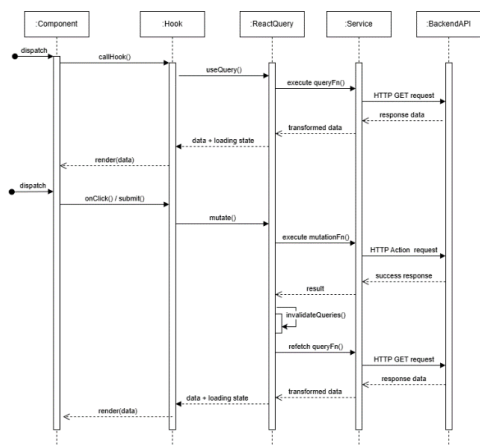
Gambar 8 menggambarkan alur proses input nilai mahasiswa ke dalam sistem berdasarkan komponen penilaian yang telah ditetapkan pada kelas atau mata kuliah. Proses ini meliputi pemilihan kelas dan komponen *assessment*, pengisian nilai untuk masing-masing mahasiswa, validasi data, serta penyimpanan hasil penilaian.



Gambar 8 Activity Penilaian Mahasiswa

3. Sequence Diagram Pengelolaan Data

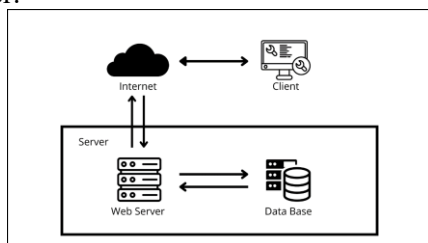
Gambar 9 menunjukkan *sequence diagram* alur pengambilan dan pembaruan data pada aplikasi frontend berbasis React Query. Berikut Sequence Diagram Pengelolaan Data pada Sistem Capaian Pembelajaran Lulusan.



Gambar 9 Sequence Diagram Aliran Data

4. Arsitektur Sistem Website

Gambar 10 menunjukkan arsitektur web yang memperlihatkan bagaimana sebuah aplikasi web berinteraksi antara klien dan server.



Gambar 10 Arsitektur Sistem Website

4.4. Perancangan Kanban Board

Gambar 11 memperlihatkan visualisasi hasil pengelolaan pengembangan sistem melalui *Kanban Board* di Trello.



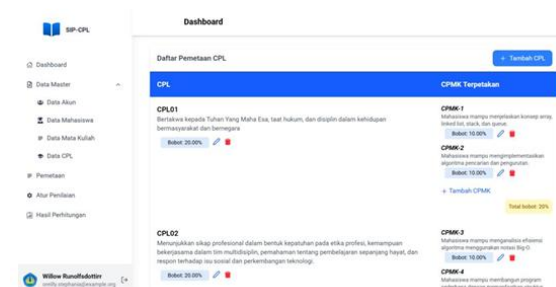
Gambar 11 Kanban Board

4.5. Development

Pada tahap ini, dilakukan proses pengembangan pada sisi *front-end* sistem mencakup pembuatan halaman-halaman sistem yang interaktif, pengelolaan data melalui *state management*, serta integrasi dengan layanan backend menggunakan API. Berikut beberapa tampilan yang sudah diimplementasikan:

1. Halaman Pemetaan Sistem

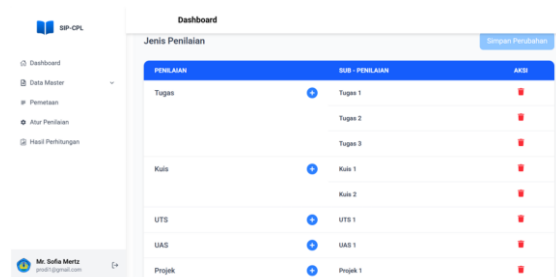
Gambar 12 merupakan Halaman Pemetaan yang digunakan untuk menghubungkan Capaian Pembelajaran Lulusan (CPL) dengan Capaian Pembelajaran Mata Kuliah (CPMK). Proses pemetaan ini bertujuan untuk memastikan bahwa setiap mata kuliah memiliki kontribusi yang terukur terhadap ketercapaian CPL program studi, sehingga perhitungan capaian pembelajaran dapat dilakukan secara sistematis dan terstruktur.



Gambar 12 Halaman Pemetaan Sistem

2. Halaman Penilaian Sistem

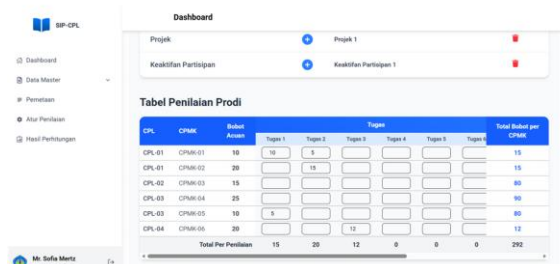
Gambar 13 merupakan Halaman Penilaian yang digunakan untuk mengatur dan mengelola komponen *assessment* yang diterapkan pada suatu kelas atau mata kuliah. Pengaturan ini menjadi dasar dalam proses input nilai mahasiswa dan perhitungan ketercapaian capaian pembelajaran secara terstruktur dan terukur.



Gambar 13 Halaman Penilaian Sistem

3. Halaman Atur Penilaian

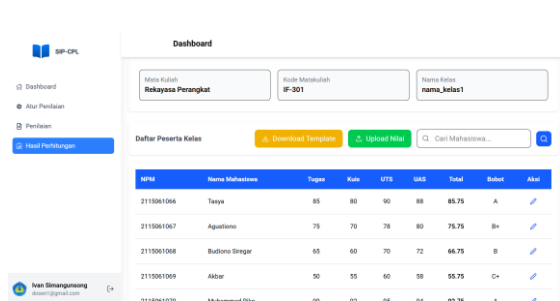
Gambar 14 merupakan Halaman Atur Penilaian merupakan fitur yang digunakan untuk menetapkan hubungan dan pembobotan antara komponen *assessment* dengan Capaian Pembelajaran Mata Kuliah (CPMK).



Gambar 14 Halaman Atur Penilaian

4. Halaman Penilaian Dosen

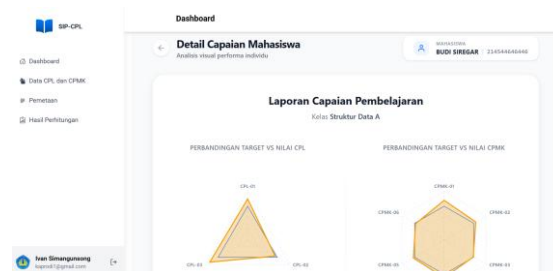
Gambar 15 memperlihatkan *Halaman Penilaian* yang menampilkan seluruh mahasiswa dalam suatu kelas beserta fitur pengisian nilai. Dosen dapat memilih mahasiswa lalu mengisi nilai untuk setiap komponen penilaian seperti tugas, kuis, praktikum, UTS, dan UAS sesuai struktur mata kuliah.



Gambar 15 Halaman Penilaian Dosen

5. Halaman Capaian Mahasiswa Kepala Program Studi

Gambar 16 merupakan Halaman Detail Capaian Mahasiswa yang menampilkan gambaran hasil belajar seorang mahasiswa di dalam kelas, diawali dengan grafik yang membandingkan target dengan nilai yang didapat pada tingkat CPL dan CPMK untuk melihat apakah standar pembelajaran sudah tercapai



Gambar 16 Halaman Capaian Mahasiswa Kepala Program Studi

4.6. Testing

Pada tahapan ini testing dilakukan dengan menggunakan metode *Blackbox Testing*. Dimana *testing* dilakukan untuk setiap kebutuhan fungsional sistem tanpa melihat kode programnya. Sehingga didapatkan hasil pengujian sebagai berikut:

Tabel 2 Total Pengujian

| Modul Pengujian | Total Pengujian | Jumlah Test Case Berhasil | Jumlah Test Case Gagal |
|----------------------|-----------------|---------------------------|------------------------|
| Authentikasi User | 10 | 7 | 3 |
| Admin Universitas | 25 | 19 | 6 |
| Admin Program Studi | 86 | 61 | 25 |
| Kepala Program Studi | 16 | 16 | 0 |
| Dosen | 20 | 20 | 0 |
| Total | 157 | 123 | 34 |

$$\text{Tingkat Keberhasilan} = \frac{123}{157} \times 100\% = 78.34\%$$

Dengan demikian, tingkat keberhasilan pengujian yang diperoleh adalah 78.34%.

4.7. Progres *Development* Kebutuhan Fungsional

Sistem belum dapat beroperasi secara penuh karena pengembangan *Application Programming Interface* (API) pada sisi *backend* masih berlangsung. Akibatnya, beberapa fungsi yang memerlukan integrasi data masih menggunakan *dummy data* pada tahap pengujian. Meskipun demikian, *semua*

UI/UX yang dibangun telah merepresentasikan alur fungsional sistem dan siap untuk integrasi lebih lanjut setelah API selesai dikembangkan.

Tabel 3 Progres *Development* Kebutuhan *Fungsional*

| Aktor | Total Fungsional | Jumlah Fungsional Selesai | Jumlah Fungsional Belum Selesai |
|----------------------|------------------|---------------------------|---------------------------------|
| Admin Universitas | 5 | 4 | 1 |
| Admin Program Studi | 12 | 10 | 2 |
| Kepala Program Studi | 5 | 2 | 3 |
| Dosen | 5 | 2 | 3 |
| Total | 27 | 18 | 9 |

$$\text{Persentase Selesai} = \frac{18}{27} \times 100\% = 66,67\%$$

Berdasarkan tabel di atas, dapat disimpulkan bahwa progres pengembangan sistem telah mencapai 66,7% dari total kebutuhan *fungsional*.

5. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa frontend Sistem Informasi Capaian Pembelajaran Lulusan (CPL) telah berhasil dianalisis, dirancang, dan dikembangkan sesuai dengan kebutuhan fungsional dan nonfungsional yang mencakup aspek usability, performance, security, dan compatibility. Arsitektur frontend berbasis web dirancang secara modular dengan pemisahan layer tampilan, hooks, service, dan API sehingga mendukung pengelolaan alur data secara terstruktur. Pengembangan dilakukan menggunakan pendekatan *Agile Kanban* secara bertahap dan terkontrol. Hasil pengujian *black box testing* menunjukkan tingkat keberhasilan fungsional sebesar 78,34% dengan progres penyelesaian pengembangan mencapai 66,7%, yang mengindikasikan bahwa sebagian besar fitur utama telah berjalan sesuai kebutuhan,

meskipun masih terdapat beberapa modul yang belum sepenuhnya terimplementasi dan memerlukan pengembangan lanjutan serta integrasi backend yang lebih komprehensif.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada para narasumber dari Program Studi Informatika yang telah memberikan informasi dan masukan dalam proses analisis kebutuhan sistem.

DAFTAR PUSTAKA

- [1] Direktorat Pembelajaran dan Kemahasiswaan, *Panduan Penyusunan Kurikulum Pendidikan Tinggi Mendukung Merdeka Belajar-Kampus Merdeka Menuju Indonesia Emas*, Edisi V. Jakarta: Direktorat Jenderal Pendidikan Tinggi, Riset, dan Teknologi, Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi, 2024.
- [2] "React – Library JavaScript untuk membangun antarmuka pengguna." Accessed: Mar. 24, 2025. [Online]. Available: <https://id.legacy.reactjs.org/>
- [3] I. Sommerville, *Software engineering*, 9th ed. Boston: Pearson, 2011.
- [4] N. S. Adilah, L. Hadjaratie, and R. Yusuf, "Pengembangan Sistem Informasi Rencana Pembelajaran Semester dan Evaluasi Capaian Pembelajaran Lulusan Berbasis Progressive Web App," *Diffusion: Journal of Systems and Information Technology*, vol. 2, no. 2, Art. no. 2, Jul. 2022, doi: 10.37031/diffusion.v2i2.15571.
- [5] M. Akbar and D. Irianto, "Usulan Sistem Penilaian Capaian Pembelajaran Mata Kuliah (CPMK) dalam Menunjang Penilaian Capaian Pembelajaran Lulusan (CPL) Program Studi," *Prosiding Seminar Nasional Teknik Industri (SENASTI)*, vol. 1, pp. 974–984, Oct. 2023.
- [6] F. N. Hasanah and R. S. Untari, *Buku Ajar Rekayasa Perangkat Lunak*. 2020. Accessed: Jul. 31, 2024. [Online]. Available: <https://press.umsida.ac.id/index.php/umsidapr ess/article/view/978-623-6833-89-6>
- [7] A. Stellman and J. Greene, *Learning agile: Understanding scrum, XP, lean, and kanban*, First edition. Beijing: O'Reilly, 2014.
- [8] A. G. Purwanto, R. Y. Wijaya, T. Timotius, and I. B. Trisno, "Website System Design Using Agile Kanban Based On QR Code," *JISA*, vol. 5, no. 1, pp. 19–27, Jun. 2022, doi: 10.31326/jisa.v5i1.1066.

- [9] Object Management Group, “Unified Modeling Language (UML), Version 2.5.1,” Object Management Group (OMG), formal/2017-12-05, Dec. 2017. [Online]. Available: <https://www.omg.org/spec/UML/2.5.1/PDF>
- [10] I. G. S. Aryandana, A. E. Permanasari, and T. B. Adji, “Comparing method equivalence class partitioning and boundary value analysis with study case add medicine module,” *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 732, no. 1, p. 012072, Jan. 2020, doi: 10.1088/1757-899X/732/1/012072.
- [11] Tailwind Labs Inc., “Tailwind CSS: Rapidly build modern websites without ever leaving your HTML,” Tailwind CSS. Accessed: Mar. 24, 2025. [Online]. Available: <https://tailwindcss.com>
- [12] D. Tanu Wijaya, I. Sumadikarta, and B. Panjaitan, “Analisa Dan Perancangan Aplikasi Evaluasi Capaian Pembelajaran Lulusan,” *pro*, vol. 4, pp. 137–147, Nov. 2023, doi: 10.59134/prosidng.v4i.558.
- [13] P. D. Cahyawardani, “Pengembangan Sistem Informasi Evaluasi Capaian Pembelajaran Lulusan Jurusan Informatika FTI UII,” *AUTOMATA*, vol. 1, no. 1, Art. no. 1, Jan. 2020, Accessed: Jul. 31, 2024. [Online]. Available: <https://journal.uui.ac.id/AUTOMATA/article/view/13873>
- [14] E. Driasetyawan and S. T. Ir. Ratnanto Fitriadi, “Perancangan Sistem Penilaian Praktikum Industri Kreatif Berbasis Outcome Based Education (OBE) (Studi Kasus: Praktikum Industri Kreatif Laboratorium Teknik Industri),” s1, Universitas Muhammadiyah Surakarta, 2023. Accessed: Jul. 31, 2024. [Online]. Available: <https://eprints.ums.ac.id/115253/>