**WEB PENGELOLAAN PENGAJUAN PROPOSAL PROGRAM**

**KREATIFITAS MAHASISWA INTERNAL**

**POLITEKNIK NEGERI INDRAMAYU**

**TUGAS AKHIR**

Sebagai salah satu syarat untuk menyelesaikan

Program Diploma III Teknik Informatika

Politeknik Negeri Indramayu



Oleh:

**RANTIKA**

**2003084**

**JURUSAN TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI INDRAMAYU**

**AGUSTUS 2023**

# **HALAMAN PENGESAHAN**

Tugas Akhir ini diajukan oleh:

Nama : RANTIKA

NIM : 2003084

Program Studi : Diploma III Teknik Informatia

Judul : Web Pengelolaan Pengajuan Program Kreatifitas Mahasiswa Internal Politeknik Negeri Indramayu

|  |
| --- |
| Pembimbing : Muhammad Anis Al Hilmi, S.Si., M.T.  NIP 199002282019031012 |

DEWAN PENGUJI

Ketua Penguji : Muhammad Anis Al Hilmi, S.Si., M.T.

NIP 199002282019031012

Anggota : Muhammad Anis Al Hilmi, S.Si., M.T.

Penguji I NIP 199002282019031012

Anggota : Muhammad Anis Al Hilmi, S.Si., M.T.

Penguji II NIP 199002282019031012

|  |  |
| --- | --- |
|  | Indramayu, Agustus 2022  Ketua Jurusan Teknik Informatika  Eka Ismantohadi, S.Kom., M.Eng  NIP 198107092021211005 |

# **MOTTO**

‘

# **PERNYATAAN**

Dengan ini saya menyatakan dengan sesungguhnya bahwa Tugas Akhir ini adalah asli hasil karya sendiri serta **Tugas Akhir** ini tidak pernah diajukan untuk memperoleh gelar Ahli Madya disuatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dirujuk dalam naskah ini dan disebutkan dalam daftar pustaka.

|  |
| --- |
| Indramayu, Agustus 2022  Yang Menyatakan,  **Rantika**  NIM 2003084 |

# **ABSTRAK**

# **ABSTRACT**

# **KATA PENGANTAR**

Puji Syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat dan hidayah-Nya penulis dapat menyelesaikan Tugas Akhir dengan judul “Web Pengelolaan Pengajuan Program Kreatifitas Mahasiswa Internal Polindra”. Tugas Akhir ini penulis susun sebagai persyaratan untuk menyelesaikan studi program Diploma III Program Studi Teknik Informatika, Jurusan Teknik Informatika, Politeknik Negeri Indramayu.

Penulis menyadari tanpa adanya dukungan dan kerja sama dari berbagai pihak, kegiatan Tugas Akhir ini tidak akan dapat berjalan baik. Untuk itu, penulis ingin menyampaikan rasa terima kasih kepada:

1. *Allah Subhanahu wa ta’ala*.
2. Bapak Rofan Aziz, S.T., M.T. selaku Direktur Politeknik Negeri Indramayu.
3. Bapak Eka Ismantohadi, S.Kom.,M.Eng. selaku ketua jurusan Teknik Informatika Politeknik Negeri Indramayu..
4. Muhammad Anis Al Hilmi, S.Si., M.T. selaku Dosen Pembimbing.
5. Bapak dan Ibu Dosen Jurusan Teknik Informatika Politeknik Negeri Indramayu yang telah memberikan ilmu yang bermanfaat.
6. Orang tua dan kelima kakak penulis serta keluarga yang selalu mendoakan dan memberikan semangat kepada penulis agar dapat menyelesaikan Tugas Akhir dengan lancar dan tepat waktu.

Dan seluruh pihak yang telah membantu dan mendukung lancarnya pembuatan Tugas Akhir dari awal hingga akhir yang tidak dapat disebutkan satu persatu.

Indramayu, Agustus 2023

Penulis

# **DAFTAR ISI**

[**HALAMAN PENGESAHAN** iii](#_Toc135657085)

[**MOTTO** iv](#_Toc135657086)

[**PERNYATAAN** v](#_Toc135657087)

[**ABSTRAK** vi](#_Toc135657088)

[**ABSTRACT** vii](#_Toc135657089)

[**KATA PENGANTAR** viii](#_Toc135657090)

[**DAFTAR ISI** ix](#_Toc135657091)

[**DAFTAR TABEL** xi](#_Toc135657092)

[**DAFTAR GAMBAR** ii](#_Toc135657093)

[**DAFTAR LAMPIRAN** iii](#_Toc135657094)

[**BAB I PENDAHULUAN** 1](#_Toc135657095)

[**1.1.** **Latar Belakang Masalah** 1](#_Toc135657096)

[**1.2.** **Rumusan Masalah** 2](#_Toc135657097)

[**1.3.** **Batasan Masalah** 2](#_Toc135657098)

[**1.4.** **Tujuan Penelitian** 2](#_Toc135657099)

[**1.5.** **Manfaat Penelitian** 2](#_Toc135657100)

[**1.6.** **Sistematika Penulisan** 2](#_Toc135657101)

[**BAB II LANDASAN TEORI** 4](#_Toc135657102)

[**2.1.** **Pengertian PKM** 4](#_Toc135657103)

[**2.2.** **PHP** 4](#_Toc135657104)

[**2.3.** **Framework Laravel** 4](#_Toc135657105)

[**2.4.** **Visual Studio Code** 4](#_Toc135657106)

[**2.5.** **Github** 5](#_Toc135657107)

[**2.6.** **Flowchart** 5](#_Toc135657108)

[**BAB III METODE PELAKSANAAN** 7](#_Toc135657109)

[**3.1.** **Orientasi Industri** 7](#_Toc135657113)

[**BAB IV HASIL DAN PEMBAHASAN** 8](#_Toc135657114)

[**4.1.** **Kesimpulan** 8](#_Toc135657119)

[**BAB V PENUTUP** 9](#_Toc135657120)

[**DAFTAR PUSTAKA** 10](#_Toc135657125)

[**LAMPIRAN** 11](#_Toc135657126)

# **DAFTAR TABEL**

[**Tabel 2. 1**. Simbol Flowchart 6](#_Toc135820388)

[**Tabel 2. 2**. Simbol Use Case Diagram 8](#_Toc135820389)

[**Tabel 2. 3**. *Simbol Activity Diagram* 9](#_Toc135820390)

[**Tabel 2. 4***. Simbol Class Diagram* 10](#_Toc135820391)

[***Tabel 2. 5****. Simbol Sequence Diagram* 11](#_Toc135820392)

[Tabel 3. 1 Kebutuhan Hardware 15](#_Toc135820515)

[Tabel 3. 2 Kebutuhan Software 15](#_Toc135820516)

# **DAFTAR GAMBAR**

[Gambar 3. 1 Metode Pelaksanaan 12](#_Toc135741198)

# **DAFTAR LAMPIRAN**

LAMPIRAN 1. *FORM* DAFTAR HADIR MAHASISWA

LAMPIRAN 2. *FORM* LAPORAN KEGIATAN PMI

LAMPIRAN 3. *FORM* PENILAIAN INDUSTRI

LAMPIRAN 4. *FORM* PENILAIAN SEMINAR PMI

LAMPIRAN 5. *FORM* PENILAIAN LAPORAN PMI

LAMPIRAN 6. *FORM* REKAPITULASI NILAI PMI

LAMPIRAN 7. *FORM* DAFTAR HADIR PESERTA SEMINAR PMI

# **BAB I PENDAHULUAN**

## **Latar Belakang Masalah**

Program Kreatifitas Mahasiswa (PKM) adalah forum yang dibentuk oleh Direktorat Kemahasiswaan dan studi di bawah Kementrian Riset, Teknologi, dan Pendidikan Tinggi (Kemenristek Dikti) Republik Indonesi untuk memfasilitasi calon mahasiswa Indonesia untuk belajar dan berkembang serta menerapkan ilmu pengetahuan dan teknologi yang dipelajari dalam perkuliahan.

Dalam upaya mengakomodasi perkembangan ide kreatif dan inofatif mahasiswa, PKM terus dikembangkan dan disempurnakan sehingga mampu mengantisipasi, memahami bahkan berkontribusi untuk mewujudkan tujuan kehidupan dunia dicanangkan PBB dalam 17 Tujuan Pembangunan Berkelanjutan (Sustainable Development Goals/SDGS) tahun 2015-2030. PKM juga dirancang untuk mengadopsi teknologi digital yang telah merasuki nyaris semua sendi kehidupan.

Politeknik Negeri Indramayu (POLINDRA) merupakan salah satu politeknik yang ikut serta dalam kegiatan PKM. Bagian yang menangani pelayanan dan administrasi PKM yaitu bidang kemahasaiswaan Politeknik Negeri Indramayu (POLINDRA). Permasalahan yang terjadi antara lain lamanya proes administrasi, penyebarluasan informasi yang kurang mendukung, serta rekap dan berkas peserta PKM yang masih manual dengan cara proposal dikirim melalui *google form* sehingga menyulitkan untuk menyaring jika terdapat judul yang sama dan tidak ada riwayat untuk pengeditan ulang.

Dalam konteks ini, pengembangan aplikasi web pengelolaan pengajuan proposal program kreatifitas mahasiswa internal Politeknik Negeri Indramayu menjadi sangat penting. Aplikasi web semacam itu akan memberikan kemudahan dan efisiensi dalam proses pengajuan proposal, sehingga mahasiswa dapat lebih fokus pada pengembangan ide-ide kreatif mereka dan mendapatkan tanggapan dengan cepat dari pihak terkait.

## **Rumusan Masalah**

Berdasarkan latar belakang yang dijabarkan, maka diperoleh suatu rumusan permasalahan yang menjadi dasar pembuataan aplikasi yaitu bagaimana membuat sebuah aplikasi pengelolaan pengajuan kegiatan PKM?

## **Batasan Masalah**

Untuk memfokuskan pembahasan, dapat diperoleh beberapa batasan masalah, di antaranya:

1. Lingkup penggunaan sistem hanya pada POLINDRA
2. Sistem hanya menangani proses pendataan tim, *review* proposal, dokumentasi proposal yang telah disetujui, dan pemberian informasi.
3. Aplikasi ini dirancang dan dibuat dengan bahasa pemograman PHP dan menggunakan framework laravel.

## **Tujuan Penelitian**

Tujuan dari pembuatan sistem ini adalah menghasilkan sebuah aplikasi pengelolaan pengajuan kegiatan program kreatifitas mahasiswa (PKM) Politeknik Negeri Indramayu untuk meningkatkan aspek efesiensi proses review proposal dan memudahkan pengeloaan PKM.

## **Manfaat Penelitian**

Manfaat yang dapat diperoleh dengan dibuatnya sistem ini adalah :

1. Mempermudah dalam pengelolaan PKM.
2. Mempermudah penyampaian informasi PKM sehingga lebih efesien
3. Mempercepat mendapatkan *feedbach review* proposal.
4. Memberikan pengetahuan terkait implementasi ilmu yang di dapat ketika kuliah saat membangun sistem

## **Sistematika Penulisan**

Dalam sistematika penulisan laporan tugas akhir ini dibagi menjadi beberapa bab yang dimana setiap bagian memiliki pembahasan yang berbeda-beda tetapi saling terkait antara satu dengan lainnya. Untuk memudahkan penulisan laporan tugas akhir ini, penulis mengurutkan serta menjabarkan setiap bagian secara sistematis. Adapun urutan penulisan laporannya sebagai berikut:

**BAB 1 PENDAHULUAN**

Pada bab ini berisi pemaparan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan laporan.

**BAB II LANDASAN TEORI**

Bab ini berisi teori dasar yang menguraikan tentang rancang bangun, aplikasi, Konseling, Diabetes Mellitus, Metode *Waterfall,* Laravel, PHP*,* MySQL, Balsamiq Mockups, Visual Studio Code, *Flowchart*, *Entity Relationship Diagram (ERD)*, Teknik Pengujian *Black box*, Pengujian Kuesioner, dan *Skala Likert*

**BAB III METODE PELAKSANAAN**

Bagian ini berisi penjelasan tentang metode pelaksanaan, metode pengumpulan data, analisis kebutuhan aplikasi, analisis kebutuhan sistem, perancangan sistem, *flowchart*, rancangan *Entity Relationship Diagram (erd)*, perancangan *database*, dan perancangan desain antarmuka.

**BAB IV HASIL DAN PEMBAHASAN**

Pada bab ini terdapat hasil penelitian serta implementasi aplikasi *website* berdasarkan tahapan implementasi *database*, implementasi desain antarmuka, hasil pengujian aplikasi menggunakan metode *black box testing* dan hasil pengujian kuesioner.

**BAB V PENUTUP**

Bab ini berisi kesimpulan dan saran yang berkaitan dengan analisis dan optimalisasi aplikasi berdasarkan yang telah diuraikan pada bab-bab sebelumnya.

# **BAB II LANDASAN TEORI**

## **Pengertian PKM**

PKM merupakan salah satu wujud implementasi Tridharma Perguruan Tinggi yang diluncurkan oleh Ditjen Diktiristek pada tahun 2022 di bawah pengelolaan Belmawa merupakan salah satu upaya untuk menumbuhkan, mewadai, dan mewujudkan ide kreatif serta inovatif mahasiswa. PKM memberikan dampak terhadap peningkatan prestasi mahasiswa dan prestasi Perguruan Tinggi dalam pemeringkatan Kementrian Pendidikan, Kebudayaan, Riset, dan Teknologi. Jenis-jenis PKM terdiri dari tujuh bagian PKM-M (Pengabdian Kepada Masayarakat), PKM-P (PKM Penelitian), PKM-K (PKM Kewirausahaan), PKM-T (PKM Teknologi), PKM-KC (PKM Karsa Cipta), PKM-GT (PKM Tertulis), PKM-AI (PKM Artikel Ilmiah).

Politeknik Negeri Indramayu merupakan salah satu perguruan tinggi negeri yang turut serta mengikuti pelaksanaan kegiatan Program Kreatifitas Mahasiswa (PKM). Untuk alur kegiatan Program Kreatifitas Mahasiswa (PKM) di Politeknik Negeri Indramayu diawali dengaan pendaftaran tim untuk didata ke sistem pusat yaitu simbelmawa, kemudian mahasiswa atau tim mulai membuat proposal dan berdiskusi dengan dosen pembimbing masing-masig mengenai isi dan pembahasan proposal tersebut. Sebelum proposal diunggah ke simbelmawa, mahasiswa atau tim harus mengirim proposal tersebut siap jadi dan diunggah ke simbelmawa untuk diseleksi. Jika proposal lolos seleksi, maka tahapselanjutnya adalah Pekan Ilmiah Kreatifitas Mahasiswa Nasional (Pimnas) dimana proposal tersebut akan didanai oleh Ditjen Dikti (Direktorat Jendral Pendidikan Tinggi, Riset, dan Teknologi).

## **Politeknik Negeri Indramayu**

## **PHP**

PHP adalah bahasa *scriptinh server-side,* bahasa pemograman yang digunakan untuk mengembangkan situs web statis atau situs web dinamis atau aplikasi WEB. PHP singkatan dari *Hypertext Pre-processor,* yang sebelumnnya disebut *Personal Home Pages.*

## **Framework Laravel**

Laravel adalah salah satu *framework* PHP yang dirancang untuk membangun situs *website* mulai dari proyek kecil hingga besar. Laravel mudah dipahami karena mengikuti pola MVC (*Model View Controller). Framework* PHP ini menjadi pilihan yang dapat diandalkan karena memiliki kinerja, fitur serta skabilitas yang baik. Laravel pertama kali diciptakan oleh Taylor Otweel di tahun 2011. Sampai saat ini, Laravel sudah mengalami banyak pengembangan sehingga kompatibel dengan teknologi *website* terbaru

## **Visual Studio Code**

Visual Studio Code (VS Code) adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform,* artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemograman JavaScript, TypeScript, dan Node.js, serta bahasa pemograman lainnya dengan bantuan *plugin* yang dapat dipasang melalui *marketplace* VisualStudio Code (seperti C++, C#, Pythin, Go, Java, dst).

## **Github**

Github adalah sebuah aplikasi berbasis *website* dengan *Version Control System (VCS)* yang menyediakan layanan untuk menyimpan *repository* dengan gratis. *Repository* merupakan tempat yang dapat digunakan untuk menyimpan file berupa *Source code.* Aplikasi ini termasuk sangat populer dan banyak digunakan termasuk oleh perusahaan-perusahaan besar skala dunia seperti Facebook, Google, dan Twiter.

## **Flowchart**

*Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urut-urutan prosedur dari suatu program. *Flowchart* menolong analis dalam memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut (Ridlo, 2017). Dalam penulisan *Flowchart* dikenal dua model, yaitu *Flowchart* Sistem dan *Flowchart* Program. Berikut penjelasannya:

1. **Fungsi Flowchart**

Fungsi utama dari flowchart adalah memberi gambaran jalannya sebuah program dari satu proses ke proses lainnya. Sehingga, alur program menjadi mudah dipahami oleh semua orang. Selain itu, fungsi lain dari flowchart adalah untuk menyederhanakan rangkaian prosedur agar memudahkan pemahaman terhadap informasi tersebut.

1. **Jenis Flowchart**
2. Flowchart Dokumen

Pertama ada flowchart dokumen (document flowchart) atau bisa juga disebut dengan paperwork flowchart. Flowchart dokumen berfungsi untuk menelusuri alur form dari satu bagian ke bagian yang lain, termasuk bagaimana laporan diproses, dicatat, dan disimpan

1. Flowchart Program

Flowchart ini menggambarkan secara rinci prosedur dari proses program. Flowchart program terdiri dari dua macam, antara lain: flowchart logika program (program logic flowchart) dan flowchart program komputer terinci (detailed computer program flowchart).

1. Flowchart Proses

Flowchart proses adalah cara penggambaran rekayasa industrial dengan cara merinci dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem.

1. Flowchart Sistem

Flowchart sistem adalah flowchart yang menampilkan tahapan atau proses kerja yang sedang berlangsung di dalam sistem secara menyeluruh.

1. **Simbol Flowchart**

Berikut adalah simbol-simbol yang sering digunakan dalam proses pembuatan flowchart

**Tabel 2. 1**. Simbol Flowchart

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Simbol** | | | | | **Keterangan** |
|  | | | | | Merepresentasikan *input* data atau *output* data yang diproses atau informasi. |
|  | | | | | Merepresentasikan operasi (proses). |
|  | | | | | Simbol *connector* berfungsi  menyatakan sambungan antar proses dalam halaman yang sama . |
|  | | | | | Simbol arus/*flow* yaitu menyatakan alur kerja. |
|  | | | | | Simbol *decision* berfungsi  menunjukkan keputusan dalam program. |
|  | | | | | |
|  | | | | | Rincian operasi berada di tempat lain (subprogram). |
|  |  |  |  |  |
|  | | | | |
|  | | | | | Simbol *preparation* berfungsi menyediakan tempat penyimpanan suatu pengolahan untuk memberi nilai  awal. |
|  | | | | | Simbol *terminal*  berfungsi menyatakan permulaan atau akhir suatu program. |
|  | | | | | Simbol *manual input* berfungsi memasukkan data secara manual menggunakan *keyboard.* |
|  | | | | | Simbol dokumen berfungsi input berasal dari dokumen dalam bentuk kertas atau *output* dalam bentuk kertas*.* |
|  | | | | | Simbol *magnetic drum* berfungsi menyimpan *input* ataupun *output* ke dalam *database*. |

## **UML (*Unified Modelling Language)***

UML (Unified Modelling Language) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. UML juga dapat didefinisikan sebagai suatu bahasa standar visualisasi, perancangan, dan pendokumentasian sistem, atau dikenal juga sebagai bahasa standar penulisan blueprint sebuah software. UML berfungsi sebagai bridge atau jembatan penerjemah antara pengembang sistem dengan pengguna. Fungsi UML juga Dapat memberikan bahasa pemodelan visual atau gambar kepada para pengguna dari berbagai macam pemrograman maupun proses umum rekayasa dan Menyatukan informasi-informasi terbaik yang ada dalam pemodelan.

UML terdiri atas pengelompokkan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Beberapa contoh diagram UML diantaranya yaitu Use Case Diagram, Activity diagram dan Class diagram. (Efendi, 2018)

1. **Use Case Diagram**

Use Case Diagram adalah satu jenis dari diagram UML (Unified Modelling Language) yang menggambarkan hubungan interaksi antara sistem dan actor.

**Tabel 2. 2**. Simbol Use Case Diagram

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | *Actor* | Berfungsi sebagai peran pengguna ketika berinteraksi dengan use case. |
|  | *Dependency* | Pada suatu elemen mandiri atau independent akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri atau  independent. |
|  | *Generalization* | Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor). |
|  | *Include* | Menspesifikasikan bahwa *Usecase* sumber secara eksplisit. |
|  | *Extend* | Menspesifikasikan bahwa *Usecase* target memperluas perilaku dari *Usecase* sumber pada suatu titik yang diberikan. |
|  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
|  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
|  | *Use Case* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor. |
|  | *Collaboration* | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |

1. **Activity Diagram**

Activity diagram atau dalam bahasa Indonesia berarti diagram aktivitas, merupakan sebuah diagram yang dapat memodelkan berbagai proses yang tejadi pada sistem

**Tabel 2. 3**. *Simbol Activity Diagram*

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | *Initial Node* | Status awal aktivitas sistem, sebuah diagram  aktivitas memiliki sebuah status awal. |
|  | *Activity* | Aktivitas dari sistem yang mencerminkan  eksekusi dari suatu aksi. |
|  | *Desicion* | Asosiasi percabangan dimana jika ada pilihan  aktivitas lebih dari satu. |
|  | *Join* | Asosiasi penggabungan dimana lebih dari  satu dari aktivitas digabungkan menjadi akhir. |
|  | *Activity Final Node* | Status akhir yang dilakukan sistem, sebuah  diagram aktivitas memiliki sebuah status akhir. |
|  | *Swimlane* | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

1. **Class Diagram**

Class diagram atau diagram kelas merupakan suatu diagram yang digunakan untuk menampilkan kelas-kelas berupa pake-paket untuk memenuhi salah satu kebutuhan paket yang akan digunakan nantinya

**Tabel 2. 4***. Simbol Class Diagram*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Simbol** | | | **Nama** | **Keterangan** |
|  | | | *Class* | Merupakan blok - blok pembangun pada pemrograman berorientasi obyek. Terdiri dari nama *class*, atribut dan *method* pada *class* |
|  | *Class* Name |  |
| + atribut  + atribut |
| + method |
|  | | |
|  | | | *Association* | Merupakan sebuah *relationship* paling umum antara dua *class*. *Association* melambangkan tipe-tipe *relationship* dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah *relationship*. |
|  | | | *Composition* | Jika sebuah *class* tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *composition* terhadap *class* tempat dia bergantung tersebut. |
|  | | | *Depedency* | Umumnya *dependency* digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain. |
|  | | | *Aggregation* | Mengindikasikan keseluruhan bagian relationship dan biasanya disebut sebagai relasi. |

1. **Sequence Diagram**

*Sequence diagram* merupakan diagram yang menggambarkan kolaborasi dinamis antara sejumlah *object*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. (Pratama, 2019).

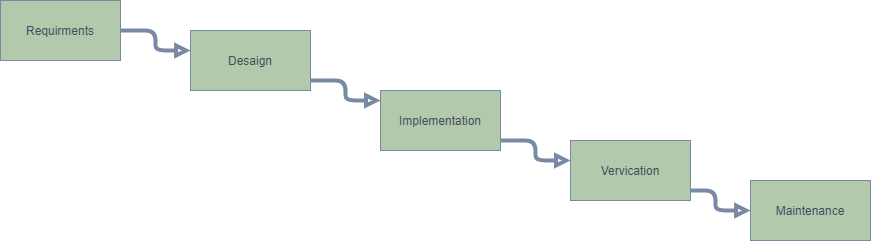
***Tabel 2. 5****. Simbol Sequence Diagram*

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | *Actor* | Menggambarkan seseorang yang berinteraksi dengan aplikasi. |
|  | *Life Line* | Objek *entity*, antarmuka yang saling berinteraksi. |
|  | *Object Message* | Menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi. |
|  | *Message to Self* | Menggambarkan pesan/hubungan objek itu sendiri yang menunjukkan urutan kejadian yang terjadi. |

# **BAB III METODE PELAKSANAAN**



## **Metode Pelaksanaan**

Metode pelaksanaan yang digunakan dalam menyelesaikan permasalahan ini atau membuat aplikasi ini yaitu menggunakan metode *waterfall,* metode *waterfall*  adalah pendekatan konsep pengembangan yang menekankan pada langkah sistematis. Sehingga, proses penciptaan sebuah system harus dilakukan secara berurutan, mulai dari tahapan identifikasi kebutuhan sampai ke proses perawatan. Tahapan dalam pelaksanaanya dapat dilihat pada gambar berikut: 

**Gambar 3. 1** Metode Pelaksanaan

Dalam membangun Aplikasi web pengelolaan program kreatifitas mahasiswa polindra internal terdapat tahapan-tahapan pada metode  *waterfall,* sebagai berikut:

1. *Requirement Gathering and analysis*

Mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun dengan mengumpulkan contoh *dataset* yang akan diolah kemudian masuk pada tahap klasifikasi.

1. *System Design*

Desain dikerjakan setelah kebutuhan selesai dikumpulkan secara lengkap. Merancang sistem dimulai dari *mock up interfaces*, *database*, gambar atau *icon* aplikasi.

1. *Implementation*

Desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Program yang dibangun langsung diuji baik secara unit. Mulai mengimplementasikan sistem yang sudah dirancang dan dianalisis ke dalam kode program. Setiap fungsi yang dibuat akan langsung dites apakah terjadi *error* atau tidak.

1. *Integration and Testing*

Penyatuan unit-unit program kemudian diuji secara keseluruhan (*system testing*). Melakukan tes semua fungsi atau fitur yang terdapat pada aplikasi apakah terjadi *error* atau tidak.

1. *Deployment of system*

Mengoperasikan program dilingkungannya dan melakukan pemeliharaan, seperti penyesuaian atau perubahan karena adaptasi dengan situasi sebenarnya dengan menerapkan aplikasi pada instansi atau perusahaan yang membutuhkan.

1. *Maintenance*

Proses pemeliharaan sistem yang sudah dibangun. Pemeliharaan dilakukan selama aplikasi masih digunakan pada institusi atau perusahaan terkait.

## **Analisis Kebutuhan Fungsional dan Non-Fungsional**

Tahapan selanjutnya adalah pengumpulan data yaitu melakukan analisis kebutuhan aplikasi. Tahapan ini bertujuan untuk memastikan bahwa aplikasi ini dapat digunakan tepat sasaran untuk mahasiswa polindra yang akan mengikuti kegiatan PKM. Analisis kebutuhan aplikasi ini terdiri dari kebutuhan fungsional dan kebutuhan non fungsional.

### Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan-kebutuhan yang memiliki keterkaitan langsung dengan aplikasi. Adapun kebutuhan fungsional Aplikasi Web Pengelolaan Program Kreatifitas Mahasiswa adapun kebutuhan fungsionalnya adalah sebagai berikut:

1. Aplikasi dapat menampilkan informasi kegiatan Program Kreatifitas Mahasiswa (PKM).
2. Aplikasi dapat melakukan proses CRUD(*Create-Read-Update-Delete)* data.
3. Aplikasi dapat melakukan proses *review.*
4. Aplikasi dapat meng-*upload* file proposal ke dalam sistem.
5. Aplikasi dapat melakukan login.
6. Aplikasi dapat melakukan register untuk mahasiswa.
7. Aplikasi dapat mengelola data *reviewer.*
8. Aplikasi dapat mengelola data dosen pembimbing.

### Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang berisi tentang proses-proses yang tidak harus atau tidak terkait secara langsung oleh sistem. Adapun kebutuhan non-fungsional dalam membangun aplikasi tersebut meliputi:

1. Aplikasi dapat dijalankan pada *web browser.*
2. Aplikasi memiliki tampilan *interface* yang mudah dipahami.
3. Aplikasi dapat digunakan apabila terhubung ke jaringan internet.

## **Analisis Kebutuhan Pengembangan Sistem**

Dalam proses pembangunan aplikasi “Web Pengelolaan Pengajuan Proposal Program Kreatifitas Mahasiswa Internal Politeknik Negeri Indramayu” terdapat kebutuhan *hardware* dan kebutuhan *software.* Berikut adalah kebutuhan-kebutuhan yang diperlukan:

### Kebutuhan Hardware

Kebutuhan *Hardware* atau perangkat keras merupakan komponen yang dapat disentuh secara fisik dan memiliki spesifik atau kriteria agar dapat menjalankan sistem dengan baik. Spesifikasi atau kriteria *hardware* yang digunakan untuk membangun sistem dapat dilihat pada Tabel 3.1

Tabel 3. 1 Kebutuhan Hardware

|  |  |  |
| --- | --- | --- |
| **No.** | **Jenis *Hardware*** | **Kebutuhan *Hardware*** |
| 1. | *Processor* | Intel(R) Core(TM) i5-5200U |
| 2. | RAM | 8GB atau lebih |
| 3. | Monitor | LCD dengan LED backlight 14 inci |

### Kebutuhan Software

Kebutuhan software atau perangkat lunak merupakan data-data yang terdapat dalam sebuah komputer. Kebutuhan ini wajib ada untuk menunjang *hardware* sehingga lebih mudah dalam membangun aplikasi ini. Spesifikasi atau kriteria dalam aplikasi ini dapat dilihat pada tablel 3.2.

Tabel 3. 2 Kebutuhan Software

|  |  |  |
| --- | --- | --- |
| **No.** | **Jenis *Software*** | **Kebutuhan *Software*** |
| 1. | Sistem Operasi | Windows 11 |
| 2. | *Framework* | Laravel versi 9 |
| 3. | Text Edotor | Visual Studio Code |
| 4. | Lokal *Server* | Laragon |
| 5. | Website *Browser* | Chrome, Microsoft Edge |

# **BAB IV HASIL DAN PEMBAHASAN**



## **Kesimpulan**

# **BAB V PENUTUP**



# **DAFTAR PUSTAKA**

Setiawan, R. (2021, August 3). *Flowchart Adalah: Fungsi, Jenis, Simbol, dan Contohnya - Dicoding Blog*. Dicoding Blog. <https://www.dicoding.com/blog/flowchart-adalah/>

Dicoding Intern. (2021, May 11). *Apa itu UML? Beserta Pengertian dan Contohnya - Dicoding Blog*. Dicoding Blog. https://www.dicoding.com/blog/apa-itu-uml/

‌ Redaksi Jagoan Hosting. (2022, February 3). *Metode Waterfall: Pengertian, Tahapan, Kelebihan & Kelemahan*. Blog Jagoan Hosting | Tutorial Website & Web Hosting Indonesia; Jagoan Hosting Indonesia. https://www.jagoanhosting.com/blog/metode-waterfall/

‌

‌

# **LAMPIRAN**