

PEMBUATAN MODUL SKEMA PENELITIAN PADA SISTEM INFORMASI LPPM

TUGAS AKHIR

Diajukan untuk Memenuhi Persyaratan Akademik dalam
Menyelesaikan Pendidikan pada Program Studi
S1 Teknik Informatika Universitas Kristen Maranatha

Oleh

Juan Sterling Martua

2072009



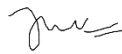
**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN MARANATHA
BANDUNG
2023**

LEMBAR PENGESAHAN

PEMBUATAN MODUL SKEMA PENELITIAN PADA SISTEM INFORMASI LPPM

**Dengan ini, saya menyatakan bahwa
isi CD ROM Laporan Penelitian sama dengan hasil revisi akhir**

Bandung, 6 Juni 2023



**Juan Sterling Martua
2072009**

**Menyetujui,
Pembimbing I**

**Dr. Hapnes Toba, M.Sc.
NIK: 710004
Penguji I**

**Erico Darmawan Handoyo, S.Kom., M.T.
NIK: 720247**

**Mengetahui,
Ketua Program Studi Teknik Informatika**

**Julianti Kasih, SE., M.Kom.
NIK: 720286**

PERNYATAAN ORISINALITAS LAPORAN PENELITIAN

Dengan ini, saya yang bertanda tangan di bawah ini:

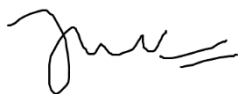
Nama : Juan Sterling Martua
NRP : 2072009
Fakultas/ Program Studi : Teknologi Informasi / Teknik Informatika

Menyatakan bahwa laporan penelitian ini adalah benar merupakan hasil karya saya sendiri dan bukan duplikasi dari orang lain.

Apabila pada masa mendatang diketahui bahwa pernyataan ini tidak benar adanya, saya bersedia menerima sanksi yang diberikan dengan segala konsekuensinya.

Demikian pernyataan ini saya buat.

Bandung, 6 Juni 2023



Juan Sterling Martua

NRP: 2072009

PERNYATAAN PUBLIKASI LAPORAN PENELITIAN

Saya yang bertanda tangan di bawah ini:

Nama : Juan Sterling Martua

NRP : 2072009

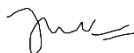
Fakultas/ Program Studi : Teknologi Informasi / Teknik Informatika

Dengan ini, saya menyatakan bahwa:

1. Demi perkembangan ilmu pengetahuan, saya menyetujui untuk memberikan kepada Universitas Kristen Maranatha Hak Bebas Royalti non eksklusif (*Non Exclusive Royalty Free Right*) atas laporan penelitian saya yang berjudul “Perancangan dan Implementasi Master Data Modul Penelitian pada Sistem Informasi LPPM”.
2. Universitas Kristen Maranatha Bandung berhak menyimpan, mengalihmediakan/ mengalihformatkan, mengelola dalam bentuk pangkalan data (*database*), mendistribusikannya, serta menampilkan dalam bentuk *softcopy* untuk kepentingan akademis tanpa perlu meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta.
3. Saya bersedia dan menjamin untuk menanggung secara pribadi tanpa melibatkan pihak Universitas Kristen Maranatha Bandung, segala bentuk tuntutan hukum yang timbul atas pelanggaran hak cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya dan untuk dapat dipergunakan sebagaimana mestinya.

Bandung, 6 Juni 2023



Juan Sterling Martua

NRP: 2072009

PRAKATA

Puji dan syukur penulis sampaikan kepada Tuhan Yang Maha Esa karena atas rahmat dan berkat-Nya penulis dapat mengikuti kegiatan MBKM magang dan dapat menyusun laporan berjudul “Perancangan dan Implementasi Master Data Modul Penelitian pada Sistem Informasi LPPM”. Laporan ini dibuat sebagai salah satu persyaratan dalam kelulusan S1 di Fakultas Teknologi Informasi Universitas Kristen Maranatha program studi Teknik Informatika serta melaporkan hasil pekerjaan dari kegiatan magang Merdeka Belajar Kampus Merdeka atau MBKM di perusahaan *Innovative, Intelligent, & Creative Studio* (i2c Studio).

Penulis ingin berterima kasih kepada pihak-pihak yang telah memberikan arahan, bimbingan, serta keterkaitannya dalam membantu penyusunan laporan ini, diantaranya:

1. Bapak Ir. Teddy Marcus Zakaria, M.T. selaku Dekan Fakultas Teknologi Informasi.
2. Ibu Julianti Kasih, SE., M.Kom. selaku Ketua Program Studi Teknik Informatika.
3. Bapak Sulaeman Santoso, S.Kom., M.T. selaku Koordinator KP/TA.
4. Bapak Dr. Hapnes Toba, M.Sc. selaku pembimbing.
5. Bapak Tjatur Kandaga Gautama, M.T., selaku pembimbing lapangan dari i2c Studio
6. Rekan – rekan satu tim dalam *Innovative, Intelligent, & Creative Studio* (i2c Studio).

Akhir kata, penulis menyadari bahwa hasil dalam laporan ini tentunya memiliki kekurangan. Penulis terbuka pada kritik, saran, dan masukan dari pembaca untuk kemajuan laporan penulis di masa depan. Penulis berharap laporan ini dapat memberikan manfaat kepada para pembaca.

Bandung, 6 Juni 2023



Juan Sterling Martua

ABSTRAK

Program MBKM adalah program yang dicanangkan oleh Menteri Pendidikan dan Kebudayaan yang bertujuan mendorong mahasiswa untuk menguasai berbagai keilmuan untuk bekal memasuki duniakerja. Salah satu bentuk kegiatan dari MBKM adalah Magang. Penulis mengikuti program Magang MBKM di i2c Studio. i2c Studio memiliki proyek untuk membuat sistem informasi untuk LPPM Maranatha yang bernama SIPPM. LPPM Maranatha membutuhkan teknologi untuk membantu pengelolaan penelitian, pengelolaan pengabdian masyarakat, pengelolaan pembuatan buku, dan pengelolaan pengajuan insentif atas hasil karya ilmiah yang berada dalam lingkup Universitas Kristen Maranatha. Proyek SIPPM ini menggunakan *framework* Laravel dan bahasa pemrograman PHP. Sebelum mengerjakan proyek, penulis terlebih dahulu mempelajari *framework* Laravel dan bahasa pemrograman PHP serta komponen pendukung lainnya, kemudian mengikuti tes yang diberikan oleh pihak i2c Studio. Setelah tes tersebut dinyatakan lulus, penulis mulai mengerjakan proyek SIPPM. Pada proyek SIPPM ini, penulis membuat migrasi dan model untuk modul Penelitian serta membuat *create*, *read*, dan *delete* (CRD) untuk Prasyarat Program Penelitian. Penulis juga melakukan pengujian terlebih dahulu untuk *master* data modul penelitian ini. Diharapkan dengan adanya *master* data modul penelitian dapat menjadi dasar pengelolaan penelitian oleh LPPM di kemudian hari.

Kata kunci: master data, sistem migrasi, modul penelitian, pengujian

ABSTRACT

The MBKM Program is an initiative launched by Minister of Education and Culture with the goal of encouraging students to acquire diverse knowledge for their preparation to enter the workforce. One of the activities offered under the MBKM Program is an internship. Author participated in the MBKM Internship Program at i2c Studio. i2c Studio had a project to develop an information system for LPPM Maranatha called SIPPM. LPPM Maranatha required technology to assist in research management, community service management, book production management, and management of incentive for scientific works in Maranatha Christian University. SIPPM project utilized Laravel framework and the PHP programming language. Prior to commencing the project, the author studied Laravel framework, PHP programming language, and other supporting components, then the author participated in a test by i2c Studio. After successfully passing the test, the author began working on SIPPM project. In SIPPM project, the author created migrations and models for Research module, such as implemented create, read, and delete (CRD) functionalities for the Research Program Prerequisites. Furthermore, the author does a testing for master data of research module. Hopefully the master data for research module will serve as a foundation for research management by LPPM in the future.

Keywords: master data, migration system, research module, testing

DAFTAR ISI

LEMBAR PENGESAHAN	1
PERNYATAAN ORISINALITAS LAPORAN PENELITIAN	2
PERNYATAAN PUBLIKASI LAPORAN PENELITIAN	3
PRAKATA	4
ABSTRAK	5
ABSTRACT	6
DAFTAR ISI	7
DAFTAR GAMBAR	10
DAFTAR TABEL	11
DAFTAR SINGKATAN	12
BAB 1 PENDAHULUAN	13
1.1 Latar Belakang	13
1.2 Rumusan Masalah	14
1.3 Tujuan Pembahasan	14
1.4 Ruang Lingkup	14
1.5 Sumber Data	15
1.6 Sistematika Penyajian	15
BAB 2 Instansi dan Deskripsi Pekerjaan	17
2.1 Profil Perusahaan	17
2.2 Struktur Organisasi Perusahaan	18
2.3 Deskripsi Pekerjaan dan Tanggung Jawab	19
2.4 Keterkaitan Hasil Studi/ <i>Training</i> dengan Pekerjaan	20
2.5 Timeline Proyek	21
BAB 3 LANDASAN TEORI	22

3.1	PHP	22
3.2	Laravel	22
3.3	Visual Studio Code	22
3.4	Github Desktop	23
3.5	PostgreSQL	23
3.6	HTML	24
3.7	Composer	24
3.8	Node.js	25
3.9	Bootstrap	25
3.10	CoreUI	25
3.11	SWAL	25
3.12	Select2	26
3.13	JavaScript	26
3.14	jQuery	26
3.15	Laravel Collective	27
3.16	CDN	27
3.17	Trello	27
3.18	DataTables	27
BAB 4	Hasil Pekerjaan	28
4.1	Migrasi dan Model Modul Penelitian	28
4.1.1	Migrasi dan Model Skema Penelitian	29
4.1.2	Migrasi dan Model Prasyarat Dosen Penelitian	32
4.1.3	Migrasi dan Model Prasyarat Detail Penelitian	34
4.1.4	Migrasi dan Model Prasyarat Anggaran Penelitian	36
4.1.5	Migrasi dan Model Prasyarat Program Penelitian	38
4.1.6	Migrasi dan Model Prasyarat Luaran Penelitian	41

4.2.1	Tahap Proposal	43
4.2.1	Pengajuan Proposal Penelitian	43
4.2.2	Review Proposal Penelitian	45
4.3	Tahap Pencairan Dana Tahap 1	47
4.4	Tahap Laporan Kemajuan	47
4.4.1	Pengajuan Laporan Kemajuan Penelitian	47
4.4.2	Review Laporan Kemajuan Penelitian	48
4.5	Tahap Laporan Akhir	49
BAB 5 SIMPULAN DAN SARAN		50
5.1.1	Simpulan	50
5.1.2	Saran	51
DAFTAR PUSTAKA		52
LAMPIRAN A LOG KERJA		1

DAFTAR GAMBAR

Gambar 2. 1 Struktur Organisasi I2C Studio	18
--	----

DAFTAR TABEL

Tabel 2. 1 <i>Timeline</i> Proyek Magang di I2C Studio.....	21
---	----

DAFTAR SINGKATAN

CDN	<i>Content Delivery Network</i>
CRD	<i>Create, Read, Delete</i>
CRUD	<i>Create, Read, Update, Delete</i>
DSTI	Direktorat Sistem dan Teknologi Infomasi
ERP	<i>Enterprise Resource Planning</i>
i2c	<i>Innovative, Intelligent & Creative</i>
IT	<i>Information Technology</i>
LPPM	Lembaga Pelayanan dan Pengabdian Masyarakat
MBKM	Merdeka Belajar Kampus Merdeka
MVC	<i>Model View Controller</i>
PHP	<i>Hypertext Preprocessor</i>
RDBMS	<i>Relational DataBase Management System</i>
SIPPM	Sistem Informasi Pelayanan dan Pengabdian Masyarakat
SWAL	<i>Sweet Alert</i>

BAB 1

PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, tujuan pembahasan, ruang lingkup, sumber data, dan sistematika penyajian yang disediakan oleh penulis agar pembaca dapat memahami laporan ini.

1.1 Latar Belakang

Seiring dengan berjalannya waktu, teknologi juga telah berkembang sangat pesat. Dengan pesatnya perkembangan teknologi tersebut membuat teknologi kini memiliki peranan yang penting dalam kehidupan manusia. Teknologi dapat membantu manusia dalam memenuhi kebutuhannya, bahkan perusahaan dan pihak lainnya juga menggunakan teknologi untuk membantu memenuhi kebutuhannya. Salah satu pihak yang akan menggunakan teknologi untuk membantu memenuhi kebutuhannya adalah LPPM Maranatha. LPPM Maranatha membutuhkan teknologi untuk membantu pengelolaan penelitian, pengelolaan pengabdian masyarakat, pengelolaan pembuatan buku, dan pengelolaan pengajuan insentif atas hasil karya ilmiah yang berada dalam lingkup Universitas Kristen Maranatha.

LPPM Maranatha memiliki cakupan tugas yang luas, sehingga membutuhkan sistem informasi yang terintegrasi dan mensinergikan seluruh pihak yang terkait. Sistem Informasi untuk LPPM saat ini sedang dikembangkan juga oleh Direktorat Sistem dan Teknologi Informasi (DSTI), namun DSTI memiliki kendala karena banyaknya tugas dan prioritas pengembangan lain yang sedang digarap oleh DSTI.

Penulis bersama dengan tim i2c Studio membantu DSTI dalam melakukan analisa, desain dan implementasi sistem informasi untuk LPPM Maranatha yang bernama SIPPM atau Sistem Informasi Pelayanan dan Pengabdian Masyarakat. Penulis mendapat beberapa tugas yang mencakup semua modul tersebut. Pada laporan ini, penulis ingin membahas mengenai skema penelitian pada SIPPM Maranatha. Pada skema penelitian ini, penulis akan membahas mengenai alur mulai

dari pengajuan proposal penelitian yang dilakukan oleh dosen hingga mencapai luaran.

Diharapkan dengan adanya SIPPM ini, terutama untuk skema penelitian, dapat membantu dan menunjang kegiatan oleh dosen, admin, pihak LPPM, serta pihak lainnya dalam pengelolaan skema penelitian.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dibahas di atas, maka dapat dirumuskan masalah yang terdapat dalam pelaksanaan magang sebagai berikut:

1. Bagaimana pihak i2c Studio dapat membantu dan menunjang kegiatan LPPM Maranatha terutama dalam pengelolaan skema penelitian?
2. Bagaimana proses yang terjadi pada skema penelitian SIPPM Maranatha?

1.3 Tujuan Pembahasan

Berdasarkan rumusan masalah yang telah dibahas di atas, maka dapat disimpulkan tujuan pembahasan pada laporan ini sebagai berikut:

1. Membuat sistem informasi yang dapat membantu dan menunjang kegiatan LPPM Maranatha terutama dalam pengelolaan skema penelitian.
2. Menjelaskan mengenai alur dan proses yang terjadi dalam skema penelitian SIPPM Maranatha.

1.4 Ruang Lingkup

Ruang lingkup yang akan dibahas pada laporan ini berdasarkan pekerjaan yang telah penulis selesaikan sebagai berikut:

1. Menjelaskan alur tahap proposal skema penelitian pada SIPPM.
2. Menjelaskan alur tahap pencairan dana tahap 1 skema penelitian pada SIPPM.
3. Menjelaskan alur tahap laporan kemajuan skema penelitian pada SIPPM.
4. Menjelaskan alur tahap laporan akhir skema penelitian pada SIPPM.

5. Menjelaskan alur tahap LOA dan luaran skema penelitian pada SIPPM.

1.5 Sumber Data

Data Primer

Data primer didapatkan dari dokumen mengenai proposal SIPPM LPPM serta *e-book* berisi materi tentang *framework* Laravel yang diberikan oleh pihak i2c Studio.

Data Sekunder

Data sekunder didapatkan dari video-video YouTube serta website yang membahas mengenai tutorial penggunaan Laravel, PostgreSQL, Bootstrap, jQuery, Laravel Collective, SWAL, Select2 dan DataTables.

1.6 Sistematika Penyajian

Sistematika penyajian laporan pada kegiatan magang i2c Studio adalah sebagai berikut:

BAB 1: PENDAHULUAN

Pada bab ini membahas mengenai latar belakang, rumusan masalah, tujuan pembahasan, ruang lingkup, sumber data, dan sistematika penyajian.

BAB 2: INSTANSI DAN DEKSRIPI PEKERJAAN

Pada bab ini membahas mengenai profil perusahaan, struktur organisasi dari perusahaan, deskripsi mengenai pekerjaan-pekerjaan yang diberikan kepada penulis, serta hasil studi dan *timeline* proyek dari kegiatan yang dilakukan penulis selama magang di i2c Studio.

BAB 3: LANDASAN TEORI

Pada bab ini membahas mengenai landasan-landasan teori yang penulis pakai untuk menjadi dasar selama penulis mengerjakan proyek di i2c Studio.

BAB 4: HASIL PEKERJAAN

Pada bab ini membahas tentang hasil pekerjaan yang telah dikerjakan oleh penulis selama melakukan kegiatan magang di i2c Studio.

BAB 5: SIMPULAN DAN SARAN

Pada bab ini berisi mengenai simpulan dan saran dari kegiatan magang di i2c Studio agar i2c Studio dapat lebih baik lagi di masa depan.

BAB 2

INSTANSI DAN DESKRIPSI PEKERJAAN

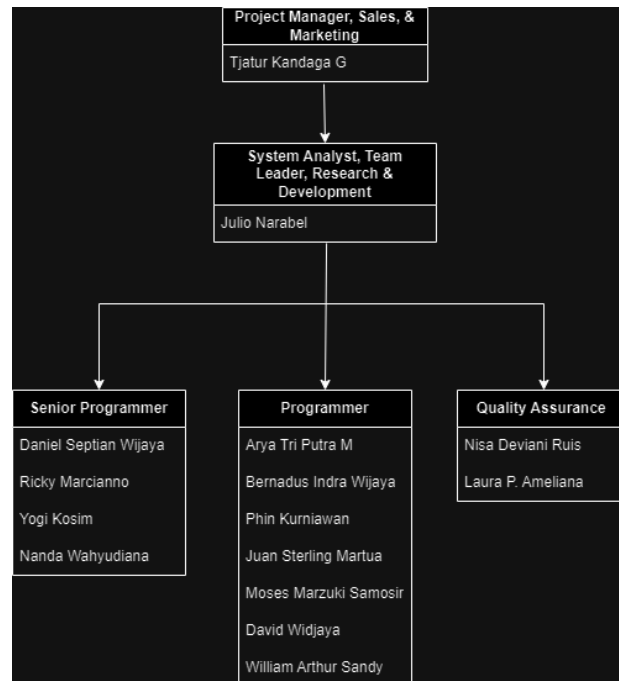
2.1 Profil Perusahaan

i2c Studio (*Innovative, Intelligent, & Creative Studio*) adalah perusahaan di bidang IT yang fokus pada pembuatan dan pengembangan perangkat lunak sebagai solusi untuk pelanggan. Perusahaan ini menawarkan solusi inovatif, cerdas, dan kreatif. i2c Studio mulai beroperasi pada Juni 2015 dan telah menjalin kemitraan dengan Fakultas Teknologi Informasi Universitas Kristen Maranatha yang berlokasi di lantai 8 gedung Graha Widya Maranatha (GWM).

Proyek awal i2c Studio adalah mengembangkan sistem Enterprise Resource Planning (ERP) untuk PT. Savero Indonesia di Jakarta. Tim i2c Studio terdiri dari dosen dan alumni dari berbagai disiplin ilmu seperti Teknik Informatika, Sistem Informasi, Teknik Elektro, dan Magister Ilmu Komputer dari Universitas Kristen Maranatha.

Perusahaan ini menerapkan proses rekayasa perangkat lunak modern dengan pendekatan yang fleksibel dan berfokus pada kebutuhan pelanggan. Pemilihan kata "Studio" dalam namanya mencerminkan dukungan mereka terhadap pendekatan kreatif, inovatif, dan *out-of-the-box* dalam menyelesaikan masalah pelanggan.

2.2 Struktur Organisasi Perusahaan



Gambar 2. 1 Struktur Organisasi I2C Studio

Gambar 2.1 merupakan diagram struktur organisasi yang ada pada perusahaan i2c Studio. Terdapat 5 jabatan/posisi yang terdapat pada i2c Studio. Jabatan pertama yaitu *Project Manager, Sales, & Marketing*. Jabatan tersebut dijabat oleh Bapa Tjatur Kandaga G. Tugas dan tanggung jawab untuk jabatan tersebut yaitu melaksanakan berbagai perencanaan seperti penetapan tujuan utama proyek dilaksanakan, melakukan penjadwalan tugas agar sesuai target yang dibutuhkan, memastikan proyek berjalan lancar serta sesuai dengan tujuan yang diiharapkan oleh klien, dan juga mempromosikan serta memasarkan i2c Studio kepada perusahaan-perusahaan lain yang membutuhkan sebuah sistem untuk membantu mereka dalam menjalankan proses bisnis.

Jabatan kedua yaitu *System Analyst, Team Leader, Research & Development*. Jabatan tersebut dijabat oleh Bapa Julio Narabel. Tugas dan tanggung jawab untuk jabatan tersebut yaitu menganalisa, menyusun, menerapkan, serta meningkatkan kinerja sistem yang sedang dirancang maupun yang sudah selesai dirancang, lalu menjadi ketua dalam tim, kemudian melakukan research terhadap proyek yang akan dirancang sebelum diserahkan dan dikerjakan agar berfungsi

dengan maksimal, serta melakukan pengembangan terhadap hasil proyek yang sudah dikerjakan oleh tim.

Jabatan ketiga yaitu *Senior Programmer*. Jabatan tersebut dijabat oleh Bapa Daniel Septian W, Bapa Ricky Marcianno, Bapa Yogi Kosim, dan Bapak Nanda Wahyudiana. Tugas dan tanggung jawab dari jabatan tersebut yaitu mengerjakan proyek yang diberikan oleh jabatan tahap ke-1 (*Project Manager, Sales, & Marketing*) & ke-2 (*System Analyst, Team Leader, Research & Development*). Tugas yang dikerjakan oleh *Senior Programmer* berada pada tingkat kesulitan yang lebih tinggi dibandingkan tugas dari jabatan tahap ke-4 (*Programmer*).

Jabatan keempat yaitu *Programmer*. Jabatan keempat ini dijabat oleh penulis sendiri Juan Sterling Martua, Arya Tri Putra M, Bernadus Indra Wijaya, Phin Kurniawan, Moses Marzuki Samosir, David Widjaya, dan William Arthur Sandy. Tugas dan tanggung jawabnya yaitu mengerjakan proyek yang diberikan oleh jabatan tahap ke-1 (*Project Manager, Sales, & Marketing*) & ke-2 (*System Analyst, Team Leader, Research & Development*). Tugas yang dikerjakan oleh *Programmer* berada pada tingkat kesulitan yang rendah hingga menengah.

Jabatan kelima yaitu *Quality Assurance*. Jabatan kelima dijabat oleh Nisa Deviani Ruis dan Laura P. Ameliana. Tugas dan tanggung jawab dari jawaban tersebut yaitu melakukan pengujian terhadap hasil proyek/sistem yang sudah dikerjakan oleh team i2c Studio untuk memastikan hasil proyek/sistem telah memenuhi syarat yang telah ditentukan berdasarkan kesepakatan pihak i2c Studio dengan klien.

2.3 Deskripsi Pekerjaan dan Tanggung Jawab

Pekerjaan dan tanggung jawab dari i2c Studio kepada penulis adalah membantu mengerjakan proyek sesuai dengan deskripsi pekerjaan yang telah diberikan oleh *Product Owner* kepada penulis melalui Trello. Sebelum memasuki pengerjaan proyek, penulis terlebih dahulu mempelajari penggunaan *framework* Laravel dan bahasa pemrograman PHP agar dapat mengerjakan proyek yang diberikan oleh I2C Studio. Selain *framework* Laravel dan bahasa pemrograman PHP, penulis juga mempelajari komponen lainnya seperti PostgreSQL, Bootstrap, CoreUI, SWAL, jQuery, JavaScript, Laravel Collective, Select2, dan DataTables.

Kemudian penulis harus menyelesaikan tes yang diberikan oleh pihak i2c Studio. Tes yang diberikan tentunya mencakup materi yang terdapat dalam proyek. Setelah diperiksa dan dinyatakan lulus, penulis baru dapat menerima dan mengerjakan proyek sesuai dengan deskripsi pekerjaan yang telah diberikan kepada penulis. Proyek tersebut dikerjakan secara bersama-sama dengan tim i2c Studio, sehingga penulis pun mengerjakan tugas tidak hanya untuk modul skema penelitian, tetapi juga mengerjakan modul skema lainnya sesuai dengan yang ditugaskan oleh *Product Owner*.

2.4 Keterkaitan Hasil Studi/ *Training* dengan Pekerjaan

Selama berkuliah tentunya penulis sudah pernah mempelajari mengenai penggunaan bahasa pemrograman PHP, DataTables, dan Bootstrap pada mata kuliah Pemrograman Web Lanjut. Lalu pada mata kuliah Pemrograman Web Dasar penulis mempelajari penggunaan Visual Studio Code, Github Desktop, HTML, dan JavaScript. Selain itu, pada mata kuliah Proyek Perangkat Lunak, penulis mempelajari *Agile* serta penggunaan Trello. Untuk database, penulis juga sudah pernah mempelajarinya pada mata kuliah Basis Data. Oleh karena itu, pada saat penulis magang di i2c Studio, materi-materi dari mata kuliah yang telah disebutkan tentunya dapat membantu penulis dalam mempelajari dan menyelesaikan pekerjaan yang diberikan oleh i2c Studio. Selain itu, pihak i2c Studio pun memberikan *link website* dan YouTube sebagai referensi kepada penulis agar dapat mempermudah penulis dalam mempelajari mengenai *framework* Laravel. Penulis juga mengakses secara mandiri di internet untuk mempelajari mengenai penggunaan PostgreSQL, CoreUI, SWAL, jQuery, Laravel Collective dan Select2.

2.5 Timeline Proyek

Berikut ini adalah *timeline* proyek yang penulis kerjakan selama magang di i2c Studio:

Tabel 2. 1 *Timeline* Proyek Magang di i2c Studio

Bulan: Februari		Bulan: Maret		Bulan: April		Bulan: Mei	
Minggu ke-	Kegiatan	Minggu ke-	Kegiatan	Minggu ke-	Kegiatan	Minggu ke-	Kegiatan
3	Mempelajari MVC dan CRUD pada Laravel	1	Mengerjakan mini proyek 1 dan 2 dari Bapak Julio Narabel	1	Mempelajari API pada Laravel	1	Mengerjakan Prasyarat Program Penelitian
4	Mempelajari ajax, Select2, middleware, dan storage pada Laravel	2	Mengerjakan mini proyek 3 dari Bapak Julio Narabel	2	Mengerjakan [Master Data I] Master Program Mahasiswa	2	Mengerjakan Prasyarat Skema Buku dan Prasyarat Program Pelayanan Publik
5	Mempelajari Laravel Collective	3	Mempelajari CoreUI	3	Mengerjakan Master Bidang Keahlian	3	Mengerjakan Prasyarat Detail Karya Ilmiah
		4	Mempelajari CoreUI	4	Mempelajari komponen bootstrap	4	Mengerjakan [Modul Questioner TKT] Migrasi & Model Part 1
		5	Mengerjakan [Migrasi & Model] Master Data 3	5	Mempelajari komponen bootstrap	5	Questioner TKT - Private Function checkUserAccess

BAB 3

LANDASAN TEORI

3.1 PHP

PHP merupakan singkatan dari *Hypertext Preprocessor*. PHP adalah bahasa penulisan skrip *open source* yang banyak digunakan dalam pemrograman atau pengembangan *website*. Bahasa ini umumnya dijalankan dalam komunikasi sisi server, dan saat ini didukung oleh hampir semua sistem. PHP juga merupakan salah satu bahasa pemrograman yang dapat berinteraksi langsung dengan database. PHP dapat diintegrasikan dengan berbagai database populer seperti MySQL, PostgreSQL, Oracle, Sybase, Informix, dan Microsoft SQL Server [1].

3.2 Laravel

Laravel merupakan *framework* PHP yang *open source* dan berisi banyak modul dasar untuk mengoptimalkan kinerja PHP dalam pengembangan aplikasi web. Laravel dapat membuat *web development* lebih cepat, lebih aman, dan lebih mudah. Laravel memberikan seperangkat alat dan sumber daya untuk membangun aplikasi berbasis PHP. Laravel memiliki ekosistem yang lengkap didukung oleh package dan ekstensi yang kompatibel [2]. *Framework* Laravel menggunakan struktur MVC (*Model View Controller*). MVC merupakan model aplikasi yang memisahkan antara data dan tampilan berdasarkan komponen aplikasi. Dengan adanya model MVC, pengguna Laravel menjadi lebih mudah dalam mempelajari Laravel. Serta menjadikan proses pembuatan aplikasi berbasis *website* menjadi lebih cepat [3].

3.3 Visual Studio Code

Visual Studio Code adalah kode editor sumber yang dikembangkan oleh Microsoft untuk Windows, Linux dan macOS. Ini termasuk dukungan untuk *debugging*, kontrol git yang tertanam dan GitHub, penyorotan sintaksis, penyelesaian kode cerdas, snippet, dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan *keyboard*, preferensi, dan meng-*install* ekstensi yang menambah fungsionalitas tambahan [4].

3.4 Github Desktop

GitHub Desktop adalah aplikasi sumber terbuka yang memungkinkan Anda berinteraksi dengan GitHub melalui antarmuka pengguna grafis (GUI) alih-alih mengandalkan baris perintah atau *browser web*. GitHub Desktop memberikan insentif kepada Anda dan tim Anda untuk bekerja sama sambil menerapkan praktik terbaik dengan Git dan GitHub.

GitHub Desktop memungkinkan pengembang untuk mengaktifkan perintah seperti pembuatan repositori, menarik permintaan, dan melakukan hanya dengan satu klik sederhana. Kenyamanan ekstra ini menambah unsur fleksibilitas ekstra untuk bekerja dengan Git dan berkolaborasi dengan pengembang lain.

Singkatnya, Git adalah sistem kontrol versi yang membantu Anda mengelola kode dan melacaknya, dan GitHub adalah platform hosting berbasis cloud yang memungkinkan pengembang mengelola repositori Git mereka. GitHub Desktop adalah aplikasi yang memungkinkan pengguna berinteraksi lebih baik dengan GitHub melalui GUI [5].

3.5 PostgreSQL

PostgreSQL adalah *relational database management system* atau RDBMS yang bersifat *open source*. Sistem manajemen basis data satu ini menggunakan bahasa query utama SQL, sama seperti MySQL. PostgreSQL dikembangkan oleh Berkeley Computer Science Department dan sudah dipercaya sebagai *database* andalan kurang lebih selama 30 tahun terakhir. *Database* PostgreSQL banyak digunakan pada berbagai aplikasi *mobile*, *web app*, hingga aplikasi *analytics*. PostgreSQL dapat mengolah data dalam tabel yang mempunyai hubungan satu sama lain. Umumnya, manajemen *database* ini banyak digunakan untuk berbagai aplikasi dengan kebutuhan pengolahan data yang kompleks. PostgreSQL adalah manajemen *database* yang mendukung berbagai bahasa pemrograman populer. Beberapa di antaranya seperti JavaScript, Java, PHP, Python, Tcl, C/C++, Perl, dan masih banyak lainnya [6].

3.6 HTML

HTML atau *Hypertext Markup Language* adalah bahasa *markup* yang digunakan untuk membuat struktur halaman *website* agar dapat ditampilkan pada *web browser*. Jadi, HTML dapat dianalogikan sebagai pondasi awal dalam menyusun kerangka halaman web secara terstruktur sebelum membahas terkait tampilan desain dan sisi fungsionalitas.

Fungsi utama penggunaan HTML sendiri adalah membangun tampilan *website* yang telah menerapkan metode semantik untuk memudahkan setiap pengembang dalam proses *development* dan *maintenance*. Kemudian, HTML juga dapat dikolaborasikan dengan penggunaan bahasa CSS (*Cascade Style Sheet*) serta JavaScript.

Peran dari HTML adalah untuk menyusun kerangka dan struktur halaman *website*. Kemudian, CSS membantu dalam memberikan tampilan desain meliputi warna, *font*, *outline*, dan lain sebagainya. Dan Tugas dari bahasa pemrograman JavaScript adalah memberikan sentuhan interaksi untuk memberikan pengalaman yang berbeda kepada *user* [7].

3.7 Composer

Composer adalah satu program aplikasi *package manager* yang digunakan manajemen proyek PHP. Composer memiliki fungsi menjalankan sebuah *script*, meng-*install file* PHP, meng-*install library* yang dibutuhkan sesuai dengan *framework* yang dirancang dengan PHP. Tidak hanya itu, composer juga melakukan *testing*.

Composer juga merupakan *package manager* yang berfungsi di bahasa pemrograman PHP yang juga sama seperti npm di JS (JavaScript), juga Maven and Gradle di Java. Composer akan menjadi bantuan untuk *provers* dalam mendeklarasikan dan mengelola dependensi proyek di PHP [8].

3.8 Node.js

Node.js adalah sebuah *software* yang digunakan untuk pengembangan aplikasi berbasis web dan ditulis dalam sintaks bahasa pemrograman JavaScript yang bersifat *open source* dan *cross platform*.

Dengan Node.js, kita bisa menjalankan kode JavaScript di mana saja, tidak hanya terbatas pada lingkungan *browser*. Dengan fitur *cross platform*, Node.js bisa berjalan di sistem operasi Windows, Mac OS X dan Linux dan tidak perlu ada perubahan kode program. Node.js adalah *software* yang mempunyai *library* sendiri sehingga bisa berjalan tanpa menggunakan program *web server* seperti Apache [9].

3.9 Bootstrap

Bootstrap adalah *framework web development* berbasis HTML, CSS, dan JavaScript yang dirancang untuk mempercepat proses pengembangan *web responsive* dan *mobile-first* (memprioritaskan perangkat seluler). Selain bisa digunakan untuk mengembangkan *website* dengan lebih cepat, Bootstrap adalah *framework* gratis yang bersifat *open source*. Skrip dan sintaks yang disediakan Bootstrap bisa diterapkan untuk berbagai komponen dalam desain web [10].

3.10 CoreUI

Core UI merupakan salah satu admin *template* bootstrap, *template* yang mempermudah *user* dalam membangun sebuah *dashboard* modern responsif yang dapat menyesuaikan di berbagai macam *device*. CoreUI menggunakan serangkaian wadah, baris, dan kolom untuk menata dan menyelaraskan konten [11].

3.11 SWAL

Sweet Alert (SWAL) merupakan *library* JavaScript yang menyediakan berbagai pilihan bagi kita untuk membuat berbagai jenis *pop-up alert* seperti *alert* biasa, *alert error*, *alert success*, *alert* konfirmasi, dan sebagainya. Penulisan kode *Sweet Alert* sangatlah singkat seperti menulis menggunakan fungsi bawaan JavaScript. Hal tersebut membuat *Sweet Alert* banyak diminati oleh para *web developer* dalam membuat sebuah *pop-up alert* pada *website* mereka [12].

3.12 Select2

Select2 adalah alternatif pengganti penggunaan *tag select* berbasis jQuery. Select2 mendukung pencarian, kumpulan data jarak jauh, dan hasil *pagination* (*scrolling* tak terbatas). Select2 memberikan kemudahan kepada *user* yaitu dapat memilih data dengan memasukkan pencarian pada dropdown tersebut. Fitur ini sangatlah membantu karena *user* tidak perlu *men-scroll* dan memilih data ketika sumber data jumlahnya cukup banyak [13].

3.13 JavaScript

JavaScript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokument HTML yang ditampilkan pada sebuah *Browser* menjadi lebih interaktif, tidak sekedar indah saja. JavaScript memberikan beberapa fungsionalitas ke dalam halaman web, sehingga dapat menjadi sebuah program yang disajikan dengan menggunakan antar muka web.

JavaScript merupakan bahasa script, yaitu bahasa yang tidak memerlukan kompiler untuk dapat menjalankannya, tetapi cukup dengan Interpreter. Tidak perlu ada proses kompilasi terlebih dahulu agar program dapat dijalankan. Browser web Netscape Navigator dan Internet Explorer adalah salah satu contoh dari salah banyak interpreter, karena kedua browser ini telah dilengkapi dengan Interpreter JavaScript. Tetapi tidak semua *browser web* dapat menjadi interpreter javascript karena belum tentu *browser* tersebut dilengkapi dengan interpreter JavaScript [14].

3.14 jQuery

jQuery adalah *library* JavaScript yang cukup andal, ringkas, dan mempunyai fitur yang cukup lengkap. *Library* ini membuat pemrosesan di HTML seperti perubahan dan manipulasi dokumen, *event handling*, animasi, dan AJAX JavaScript menjadi lebih sederhana. Hal ini didukung dengan API yang mudah digunakan dan dapat bekerja di berbagai macam *browser* [15].

3.15 Laravel Collective

Laravel collective adalah sebuah *package* yang memudahkan *developer* untuk membuat Form Dan HTML, maksudnya sintaks untuk membuat kedua jenis elemen ini menjadi lebih singkat dan lebih mudah digunakan untuk keperluan tertentu [16].

3.16 CDN

CDN kepanjangan dari *Content Delivery Network*. CDN adalah sebuah kumpulan server global yang terletak di beberapa data *center* di berbagai belahan dunia. Jaringan global ini bertugas untuk mengirimkan konten dari server ke suatu *website*. CDN sangat berperan dalam percepatan pengiriman konten *website* di mana server dan *visitor* terpaut jarak lokasi yang jauh [17].

3.17 Trello

Trello adalah sebuah aplikasi kolaboratif yang memungkinkan penggunaanya untuk mengatur berbagai macam pekerjaan dalam satu tempat. Menggunakan ini, semua orang yang terlibat dengan pekerjaan bisa tahu siapa saja yang mengerjakan apa, dan sejauh mana pekerjaan tersebut sudah berjalan. Trello banyak dicari karena dapat memudahkan pembagian tugas, *update progress*, dan lain sebagainya. Jadi tak ada lagi cerita *tasking* yang lewat dari *deadline* dan tidak dikerjakan dengan baik memiliki tampilan antarmuka yang unik. Sebagai sebuah aplikasi kolaborasi, Trello lebih mirip *sticky notes* yang simpel. Dengan tampilan yang *clear*, pengguna lebih mudah ketika menggunakannya [18].

3.18 DataTables

DataTables adalah *plugin* yang di bangun dari jQuery untuk menampilkan data dalam bentuk tabel dan sudah terintegasi dengan tombol *search* dan *pagination* untuk penomoran data pada tabel sehingga kita tidak perlu lagi membuat form pencarian pada data [19].

BAB 4

HASIL PEKERJAAN

Pada skema penelitian, terdapat proses-proses yang harus dilewati oleh dosen peneliti, dosen reviewer, dan admin.

4.1 Migrasi dan Model Modul Penelitian

Ketika hasil tes sudah diumumkan dan dinyatakan lulus, penulis diundang untuk bergabung dalam Trello untuk pengerjaan proyek SIPPM. Pada awal ketika masuk ke dalam pengerjaan proyek, penulis membuat kode program untuk migrasi dan model modul penelitian. Terdapat beberapa kode program yang telah penulis kerjakan untuk migrasi dan model pada modul penelitian. Kode program tersebut dibuat berdasarkan *task* yang diberikan Bapak Daniel Septian kepada penulis melalui Trello. Pertama-tama hal yang harus dilakukan adalah membuat database baru pada PostgreSQL kemudian menghubungkannya dengan proyek SIPPM.

```
9   DB_CONNECTION=pgsql
10  DB_HOST=127.0.0.1
11  DB_PORT=5432
12  DB_DATABASE=lppm
13  DB_USERNAME=postgres
14  DB_PASSWORD=
```

Gambar 4.5.1 adalah kode program yang penulis kerjakan untuk menghubungkan PostgreSQL dengan proyek SIPPM

4.1.1 Migrasi dan Model Skema Penelitian

Tabel mst_research_schemes

- Nama Migrasi: create_mst_research_schemes_table.
 - Nama Tabel: mst_research_schemes.
 - Rincian Kolom:
 - id
 - research_scheme => string => unique
 - status => boolean => default false
 - timestamps()
-
- softDeletes()

Gambar 4.5.1 adalah *task* untuk membuat migrasi Skema Penelitian. Pada *task* ini, penulis membuat migrasi dan untuk tabel mst_research_schemes.

```
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreateMstResearchSchemesTable extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14     public function up()
15     {
16         Schema::create('mst_research_schemes', function (Blueprint $table) {
17             $table->id();
18             $table->string('research_scheme')->unique();
19             $table->boolean('status')->default(false);
20             $table->timestamps();
21             $table->softDeletes();
22         });
23     }
24
25     /**
26      * Reverse the migrations.
27      *
28      * @return void
29      */
30     public function down()
31     {
32         Schema::dropIfExists('mst_research_schemes');
33     }
34 }
35
```

Gambar 4.5.3 adalah kode program yang penulis kerjakan untuk migrasi tabel mst_research_schemes. Terdapat kode program yang berisi kolom-kolom tabel yang diperlukan sesuai dengan *task*.

- Rincian Model:
 - Model: /Master/ResearchScheme

Gambar 4.5.4 adalah *task* untuk membuat model Skema Penelitian bernama ResearchScheme.

- Relasi di Model Master/ResearchScheme:
 - hasOne:
 - Nama Relasi: researchscheme_rrlecturer.
 - Kolom Terkait: research_scheme_id.
 - Model Target: /Master/Requirement/ResearchReqLecturer.

Gambar 4.5.5 adalah *task* untuk membuat relasi antara Skema Penelitian dengan Prasyarat Dosen Penelitian.

- Relasi di Model /Master/ResearchScheme:
 - hasOne:
 - Nama Relasi: researchscheme_rrdetail.
 - Kolom Terkait: research_scheme_id.
 - Model Target: /Master/Requirement/ResearchReqDetail.

Gambar 4.5.6 adalah *task* untuk membuat relasi antara Skema Penelitian dengan Prasyarat Detail Penelitian.

- Rincian Model:
 - Relasi di Model Master/ResearchScheme:
 - belongsToMany:
 - Nama Relasi: researchscheme_studentprogramme.
 - Kolom Terkait: research_scheme_id.
 - Model Target: Master/StudentProgramme.

Gambar 4.5.7 adalah *task* untuk membuat relasi antara Skema Penelitian dengan Prasyarat Program Penelitian.

◦ Relasi di Model Master/ResearchScheme:

- hasMany:
 - Nama Relasi: researchscheme_rroutput.
 - Kolom Terkait: research_scheme_id.
 - Model Target: /Master/Requirement/ResearchReqOutput.

Gambar 4.5.8 adalah *task* untuk membuat relasi antara Skema Penelitian dengan Prasyarat Luaran Penelitian.

◦ Relasi di Model /Master/ResearchScheme:

- hasOne:
 - Nama Relasi: researchscheme_rrbudget.
 - Kolom Terkait: research_scheme_id.
 - Model Target: /Master/Requirement/ResearchReqBudget.

Gambar 4.5.9 adalah *task* untuk membuat relasi antara Skema Penelitian dengan Prasyarat Luaran Penelitian.

```

1  <?php
2
3  namespace App\Models\Master;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\SoftDeletes;
8  use App\Models\Master\Requirement\ResearchReqLecturer;
9  use App\Models\Master\Requirement\ResearchReqDetail;
10 use App\Models\Master\Requirement\ResearchReqBudget;
11 use App\Models\Master\StudentProgramme;
12
13 class ResearchScheme extends Model
14 {
15     use HasFactory;
16     use SoftDeletes;
17     protected $fillable = [
18         'research_scheme',
19         'status'
20     ];
21     protected $table = 'mst_research_schemes';
22     public function researchscheme_rrlecturer(){
23         return $this->hasOne(ResearchReqLecturer::class, "research_scheme_id");
24     }
25     public function researchscheme_rrdetail(){
26         return $this->hasOne(ResearchReqDetail::class, "research_scheme_id");
27     }
28     public function researchscheme_rrbudget(){
29         return $this->hasOne(ResearchReqBudget::class, "research_scheme_id");
30     }
31     public function researchscheme_studentprogramme(){
32         return $this->belongsToMany(StudentProgramme::class, "mst_research_req_programmes", "research_scheme_id", "student_programme_id");
33     }
34     public function researchscheme_rroutput(){
35         return $this->hasMany(ResearchReqOutput::class, "research_scheme_id");
36     }
37 }

```

Gambar 4.5.10 adalah kode program yang penulis kerjakan untuk model ResearchScheme. Terdapat kode program untuk kolom tabel yang dapat diisi secara manual serta relasi tabel yang diperlukan.

4.1.2 Migrasi dan Model Prasyarat Dosen Penelitian

Tabel mst_research_req_lecturers

- Nama Migrasi: create_mst_research_req_lecturers_table.
- Nama Tabel: mst_research_req_lecturers.
- Rincian Kolom:
 - research_scheme_id => unsigned integer => Primary Key => FK ke tabel mst_research_schemes.
 - lecturer_reqs => string => nullable
 - min_lecturer => integer => default 0
 - max_lecturer => integer => nullable
 - min_lead_position_id => unsigned integer => nullable
 - max_lead_position_id => unsigned integer => nullable
 - min_lead_position_strata => string => nullable
 - max_lead_position_strata => string => nullable
 - maximum_age => integer => nullable
 - min_member => integer => default 0
 - min_member_position_id => unsigned integer => nullable
 - min_member_position_strata => string => nullable
 - min_partner => integer => default 0
 - min_partner_position_id => unsigned integer => nullable
 - min_partner_position_strata => string => nullable
 - timestamps()
 - softDeletes()

Gambar 4.5.11 adalah *task* untuk membuat migrasi Prasyarat Dosen Penelitian. Pada *task* ini, penulis membuat migrasi untuk tabel mst_research_req_lecturer.

```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 class CreateMstResearchReqLecturersTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('mst_research_req_lecturers', function (Blueprint $table) {
17             $table->unsignedInteger('research_scheme_id')->primary();
18             $table->foreign('research_scheme_id')->references('id')->on('mst_research_schemes');
19             $table->string('lecturer_reqs')->nullable();
20             $table->integer('min_lecturer')->default(0);
21             $table->integer('max_lecturer')->nullable();
22             $table->unsignedInteger('min_lead_position_id')->nullable();
23             $table->unsignedInteger('max_lead_position_id')->nullable();
24             $table->string('min_lead_position_strata')->nullable();
25             $table->string('max_lead_position_strata')->nullable();
26             $table->integer('maximum_age')->nullable();
27             $table->integer('min_member')->default(0);
28             $table->unsignedInteger('min_member_position_id')->nullable();
29             $table->string('min_member_position_strata')->nullable();
30             $table->integer('min_partner')->default(0);
31             $table->unsignedInteger('min_partner_position_id')->nullable();
32             $table->string('min_partner_position_strata')->nullable();
33             $table->timestamps();
34             $table->softDeletes();
35         });
36     }
37
38     /**
39      * Reverse the migrations.
40      *
41      * @return void
42      */
43     public function down()
44     {
45         Schema::dropIfExists('mst_research_req_lecturers');
46     }
47 }
```

Gambar 4.5.12 adalah kode program yang penulis kerjakan untuk migrasi tabel `mst_research_req_lecturers`. Terdapat kode program yang berisi kolom-kolom tabel yang diperlukan sesuai dengan *task*.

```

o Relasi di Model Master/Requirement/ResearchReqLecturer:
  ■ belongsTo:
    ■ Relasi 1:
      ■ Nama Relasi: rrl lecturer_researchscheme.
      ■ Kolom Terkait: research_scheme_id.
      ■ Model Target: /Master/ResearchScheme.
    ■ Relasi 2:
      ■ Nama Relasi: rrl lecturer_leadposmin.
      ■ Kolom Terkait: min_lead_position_id.
      ■ Model Target: /Master/AcademicPosition.
    ■ Relasi 3:
      ■ Nama Relasi: rrl lecturer_leadposmax.
      ■ Kolom Terkait: max_lead_position_id.
      ■ Model Target: /Master/AcademicPosition.
  ■ Relasi 4:
    ■ Nama Relasi: rrl lecturer_memberposmin.
    ■ Kolom Terkait: min_member_position_id.
    ■ Model Target: /Master/AcademicPosition.
  ■ Relasi 5:
    ■ Nama Relasi: rrl lecturer_memberposmax.
    ■ Kolom Terkait: max_member_position_id.
    ■ Model Target: /Master/AcademicPosition.
  ■ Relasi 6:
    ■ Nama Relasi: rrl lecturer_partnerposmin.
    ■ Kolom Terkait: min_partner_position_id.
    ■ Model Target: /Master/AcademicPosition.
  ■ Relasi 7:
    ■ Nama Relasi: rrl lecturer_partnerposmax.
    ■ Kolom Terkait: max_partner_position_id.
    ■ Model Target: /Master/AcademicPosition.

```

Gambar 4.5.13 adalah *task* untuk membuat model Prasyarat Dosen Penelitian. Pada *task* ini, penulis membuat model untuk tabel `mst_research_req_lecturers` bernama `ResearchReqLecturer`.

```

1  <?php
2
3  namespace App\Models\Master\Requirement;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\SoftDeletes;
8  use App\Models\Master\ResearchScheme;
9  use App\Models\Master\AcademicPosition;
10
11 class ResearchReqLecturer extends Model
12 {
13     use HasFactory;
14     use SoftDeletes;
15     protected $fillable = [
16         'research_scheme_id',
17         'lecturer_reqs',
18         'min_lecturer',
19         'max_lecturer',
20         'min_lead_position_id',
21         'max_lead_position_id',
22         'min_lead_position_strata',
23         'max_lead_position_strata',
24         'maximum_age',
25         'min_member',
26         'min_member_position_id',
27         'min_member_position_strata',
28         'min_partner',
29         'min_partner_position_id',
30         'min_partner_position_strata'
31     ];
32     protected $table = "mst_research_req_lecturers";
33     protected $primaryKey = "research_scheme_id";
34     const LECTURER = ['NIDN', 'NIDK', 'NIDN/NIDK'];
35     const STRATA = ['D-III', 'S-1', 'S-2', 'S-3', 'PROFESI'];
36     public function rrlecturer_researchscheme(){
37         return $this->belongsTo(ResearchScheme::class,"research_scheme_id");
38     }
39     public function rrlecturer_leadposmin(){
40         return $this->belongsTo(AcademicPosition::class,"min_lead_position_id");
41     }
42     public function rrlecturer_leadposmax(){
43         return $this->belongsTo(AcademicPosition::class,"max_lead_position_id");
44     }
45     public function rrlecturer_memberposmin(){
46         return $this->belongsTo(AcademicPosition::class,"min_member_position_id");
47     }
48     public function rrlecturer_memberposmax(){
49         return $this->belongsTo(AcademicPosition::class,"max_member_position_id");
50     }
51     public function rrlecturer_partnerposmin(){
52         return $this->belongsTo(AcademicPosition::class,"min_partner_position_id");
53     }
54     public function rrlecturer_partnerposmax(){
55         return $this->belongsTo(AcademicPosition::class,"max_partner_position_id");
56     }
57 }
58

```

Gambar 4.5.14 adalah kode program yang penulis kerjakan untuk model ResearchReqLecturer. Terdapat kode program untuk kolom tabel yang dapat diisi secara manual serta relasi tabel yang diperlukan.

4.1.3 Migrasi dan Model Prasyarat Detail Penelitian

Tabel mst_research_req_details

- Nama Migrasi: create_mst_research_req_details_table.
- Nama Tabel: mst_research_req_details.
- Rincian Kolom:
 - research_scheme_id => unsigned integer => Primary Key => FK ke tabel mst_research_schemes.
 - min_mandatory_output => integer => default 0.
 - min_student => integer => default 0.
 - max_student => integer => nullable.
 - disciplines => string => nullable.
 - student_transport_cost => float => nullable.
 - student_transport_period => string => nullable.
 - waiting_period => integer => nullable.
 - waiting_span => string => nullable.
 - timestamps()
 - softdeletes()

Gambar 4.5.15 adalah *task* untuk membuat migrasi Prasyarat Detail Penelitian. Pada *task* ini, penulis membuat migrasi untuk tabel `mst_research_req_details`.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreateMstBookReqDetailsTable extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14      public function up()
15      {
16          Schema::create('mst_book_req_details', function (Blueprint $table) {
17              $table->unsignedInteger('book_scheme_id')->primary();
18              $table->foreign('book_scheme_id')->references('id')->on('mst_book_schemes');
19              $table->integer('min_lecturer')->default(0);
20              $table->integer('max_lecturer')->nullable();
21              $table->integer('min_mandatory_output')->default(0);
22              $table->integer('max_mandatory_output')->nullable();
23              $table->string('duration_type');
24              $table->integer('max_duration');
25              $table->string('discipline')->nullable();
26              $table->string('book_size');
27              $table->integer('collected_print');
28              $table->boolean('revision')->default(false);
29              $table->timestamps();
30              $table->softDeletes();
31          });
32      }
33
34      /**
35       * Reverse the migrations.
36       *
37       * @return void
38       */
39      public function down()
40      {
41          Schema::dropIfExists('mst_book_req_details');
42      }
43  }
44

```

Gambar 4.5.16 adalah kode program yang penulis kerjakan untuk migrasi tabel `mst_research_req_details`. Terdapat kode program yang berisi kolom-kolom tabel yang diperlukan sesuai dengan *task*.

- Rincian Model:
 - Model: `/Master/Requirement/ResearchSchemeDetail`.
 - Sertakan `const DISCIPLINES = ['MONO', 'MULTI', 'MONO / MULTI']`.
 - Sertakan `const TRANSPORT_PERIOD = ['hari', 'minggu', 'bulan', 'semester', 'tahun', 'penelitian']`.
 - Sertakan `const WAITING_SPAN = ['hari', 'minggu', 'bulan', 'tahun']`.
 - Relasi di Model `/Master/Requirement/ResearchReqDetail`:
 - belongsTo:
 - Nama Relasi: `rrdetail_researchscheme`.
 - Kolom Terkait: `research_scheme_id`.
 - Model Target: `/Master/ResearchScheme`.

Gambar 4.5.17 adalah *task* untuk membuat model Prasyarat Detail Penelitian. Pada *task* ini, penulis membuat model untuk tabel `mst_research_req_details` bernama `ResearchSchemeDetail`.

```

1  <?php
2
3  namespace App\Models\Master\Requirement;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\SoftDeletes;
8  use App\Models\Master\ResearchScheme;
9
10 class ResearchReqDetail extends Model
11 {
12     use HasFactory;
13     use SoftDeletes;
14     protected $fillable = [
15         'research_scheme_id',
16         'min_mandatory_output',
17         'min_student',
18         'max_student',
19         'disciplines',
20         'student_transport_cost',
21         'student_transport_period',
22         'waiting_period',
23         'waiting_span'
24     ];
25     protected $table = "mst_research_req_details";
26     protected $primaryKey = "research_scheme_id";
27     const DISCIPLINES = ['MONO', 'MULTI', 'MONO / MULTI'];
28     const TRANSPORT_PERIOD = ['hari', 'minggu', 'bulan', 'semester', 'tahun', 'penelitian'];
29     const WAITING_SPAN = ['hari', 'minggu', 'bulan', 'tahun'];
30     public function rrdetail_researchscheme()
31     {
32         return $this->belongsTo(ResearchScheme::class, "research_scheme_id");
33     }
34 }
35

```

Gambar 4.5.18 adalah kode program yang penulis kerjakan untuk model `ResearchSchemeDetail`. Terdapat kode program untuk kolom tabel yang dapat diisi secara manual serta relasi tabel yang diperlukan.

4.1.4 Migrasi dan Model Prasyarat Anggaran Penelitian

Tabel `mst_research_req_budgets`

- Nama Migrasi: `create_mst_research_req_budgets_table`.
- Nama Tabel: `mst_research_req_mandatory_output`.
- Rincian Kolom:
 - `research_scheme_id` => unsigned integer => Primary Key => FK ke tabel `mst_research_schemes`.
 - `min_budget` => float => default 0
 - `max_budget` => float => nullable
 - `lead_ejm` => integer => nullable
 - `member_ejm` => integer => nullable
 - `ejm_period` => integer => nullable
 - `ejm_period_type` => string => nullable
 - `timestamps()`
 - `softdeletes()`

Gambar 4.5.19 adalah *task* untuk membuat migrasi Prasyarat Anggaran Penelitian. Pada *task* ini, penulis membuat migrasi untuk tabel `mst_research_req_mandatory_output`.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreateMstResearchReqBudgetsTable extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14     public function up()
15     {
16         Schema::create('mst_research_req_mandatory_output', function (Blueprint $table) {
17             $table->unsignedInteger('research_scheme_id')->primary();
18             $table->foreign('research_scheme_id')->references('id')->on('mst_research_schemes');
19             $table->float('min_budget')->default(0);
20             $table->float('max_budget')->nullable();
21             $table->integer('lead_ejm')->nullable();
22             $table->integer('member_ejm')->nullable();
23             $table->integer('ejm_period')->nullable();
24             $table->string('ejm_period_type')->nullable();
25             $table->timestamps();
26             $table->softDeletes();
27         });
28     }
29
30     /**
31      * Reverse the migrations.
32      *
33      * @return void
34      */
35     public function down()
36     {
37         Schema::dropIfExists('mst_research_req_mandatory_output');
38     }
39 }
40

```

Gambar 4.5.20 adalah kode program yang penulis kerjakan untuk migrasi tabel `mst_research_mandatory_output`. Terdapat kode program yang berisi kolom-kolom tabel yang diperlukan sesuai dengan *task*.

- Rincian Model:
 - Model: Master/Requirement/ResearchReqBudget.
 - Sertakan const TYPE = ['semester', 'tahun', 'periode'].
 - Relasi di Model Master/Requirement/ResearchReqBudget:
 - belongsTo:
 - Nama Relasi: rrbudget_researchscheme.
 - Kolom Terkait: research_scheme_id.
 - Model Target: /Master/ResearchScheme.

Gambar 4.5.21 adalah *task* untuk membuat model Prasyarat Anggaran Penelitian. Pada *task* ini, penulis membuat model untuk tabel `mst_research_req_mandatory_output` bernama `ResearchReqBudget`.

```

1  <?php
2
3  namespace App\Models\Master\Requirement;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\SoftDeletes;
8  use App\Models\Master\ResearchScheme;
9
10 class ResearchReqBudget extends Model
11 {
12     use HasFactory;
13     protected $fillable = [
14         'research_scheme_id',
15         'min_budget',
16         'max_budget',
17         'lead_ejm',
18         'member_ejm',
19         'ejm_period',
20         'ejm_period_type',
21     ];
22     protected $table = "mst_research_req_mandatory_output";
23     protected $primaryKey = "research_scheme_id";
24     const TYPE = ['semester', 'tahun', 'periode'];
25     public function rrbudget_researchscheme(){
26         return $this->belongsTo(ResearchScheme::class,"research_scheme_id");
27     }
28 }
29

```

Gambar 4.5.22 adalah kode program yang penulis kerjakan untuk model ResearchReqBudget. Terdapat kode program untuk kolom tabel yang dapat diisi secara manual serta relasi tabel yang diperlukan.

4.1.5 Migrasi dan Model Prasyarat Program Penelitian

Tabel mst_research_req_programmes

- Nama Migrasi: create_mst_research_req_programmes_table.
- Nama Tabel: mst_research_req_programmes.
- Rincian Kolom:
 - research_scheme_id => unsigned integer => FK ke tabel mst_research_schemes.
 - student_programme_id => unsigned integer => FK ke tabel mst_student_programmes.

- timestamps()
- softdeletes()
- Set kombinasi antara kolom research_scheme_id dgn kolom student_programme_id sebagai Composite Primary Key (utk relasi many to many).

Gambar 4.5.23 adalah *task* untuk membuat migrasi Prasyarat Program Penelitian. Pada *task* ini, penulis membuat migrasi untuk tabel `mst_research_req_programmes`. Tabel ini sebenarnya adalah tabel yang terbentuk karena tabel Skema Penelitian memiliki relasi *many to many* dengan tabel Program Mahasiswa.

```

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreateMstResearchReqProgrammesTable extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14     public function up()
15     {
16         Schema::create('mst_research_req_programmes', function (Blueprint $table) {
17             $table->unsignedInteger('research_scheme_id')->index();
18             $table->foreign('research_scheme_id')->references('id')->on('mst_research_schemes');
19             $table->unsignedInteger('student_programme_id')->index();
20             $table->foreign('student_programme_id')->references('id')->on('mst_student_programmes');
21             $table->timestamps();
22             $table->softDeletes();
23             $table->primary(['research_scheme_id', 'student_programme_id']);
24         });
25     }
26
27     /**
28      * Reverse the migrations.
29      *
30      * @return void
31      */
32     public function down()
33     {
34         Schema::dropIfExists('mst_research_req_programmes');
35     }
36 }
37

```

Gambar 4.5.24 adalah kode program yang penulis kerjakan untuk migrasi tabel `mst_research_req_programmes`. Terdapat kode program yang berisi kolom-kolom tabel yang diperlukan sesuai dengan *task*. Untuk kolom `research_scheme_id` dan `student_programme_id` dibuat menjadi *composite primary key* karena tabel `mst_research_req_programmes` dan tabel `mst_student_programmes` memiliki relasi *many to many*.

- Rincian Model:
 - Relasi di Model Master/ResearchScheme:
 - belongsToMany:
 - Nama Relasi: researchscheme_studentprogramme.
 - Kolom Terkait: research_scheme_id.
 - Model Target: Master/StudentProgramme.
 - Relasi di Model Master/StudentProgramme:
 - belongsToMany:
 - Nama Relasi: studentprogramme_researchscheme.
 - Kolom Terkait: student_programme_id.
 - Model Target: Master/ResearchScheme.

Gambar 4.5.25 adalah *task* untuk membuat relasi tabel Prasyarat Program Penelitian. Pada *task* ini, penulis membuat relasi pada model ResearchScheme dan StudentProgramme.

```

17 public function studentprogramme_researchscheme(){
18     return $this->belongsToMany(ResearchScheme::class,"mst_research_req_programmes","student_programme_id","research_scheme_id");
19 }

```

Gambar 4.5.26 adalah kode program yang telah penulis kerjakan untuk membuat relasi antara tabel mst_student_programme dengan tabel mst_research_scheme di model StudentProgramme.

```

31 public function researchscheme_studentprogramme(){
32     return $this->belongsToMany(StudentProgramme::class,"mst_research_req_programmes","research_scheme_id","student_programme_id");
33 }

```

Gambar 4.5.27 adalah kode program yang telah penulis kerjakan untuk membuat relasi antara tabel mst_research_scheme dengan tabel mst_student_programme di model ResearchScheme.

4.1.6 Migrasi dan Model Prasyarat Luaran Penelitian

Tabel mst_research_req_outputs

- Nama Migrasi: create_mst_research_req_outputs_table.
- Nama Tabel: mst_research_req_outputs.
- Rincian Kolom:
 - id
 - research_scheme_id => unsigned integer => FK ke tabel mst_research_schemes.
 - output_id => unsigned integer => FK ke tabel mst_outputs.
 - category => string
 - output_title => string
 - min_grade => string => nullable
 - max_grade => string => nullable
 - timestamps()
 - softDeletes()

Gambar 4.5.28 adalah *task* untuk membuat migrasi Prasyarat Luaran Penelitian. Pada *task* ini, penulis membuat migrasi untuk tabel mst_research_req_outputs.

```
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  class CreateMstResearchReqOutputsTable extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14      public function up()
15      {
16          Schema::create('mst_research_req_outputs', function (Blueprint $table) {
17              $table->id();
18              $table->unsignedInteger("research_scheme_id");
19              $table->foreign("research_scheme_id")->references("id")->on("mst_research_schemes");
20              $table->unsignedInteger(["output_id"]);
21              $table->foreign("output_id")->references("id")->on("mst_outputs");
22              $table->string("category");
23              $table->string("output_title");
24              $table->string("min_grade")->nullable();
25              $table->string("max_grade")->nullable();
26              $table->timestamps();
27              $table->softDeletes();
28          });
29      }
30
31      /**
32       * Reverse the migrations.
33       *
34       * @return void
35       */
36      public function down()
37      {
38          Schema::dropIfExists('mst_research_req_outputs');
39      }
40  }
```

Gambar 4.5.29 adalah kode program yang penulis kerjakan untuk migrasi tabel `mst_research_req_outputs`. Terdapat kode program yang berisi kolom-kolom tabel yang diperlukan sesuai dengan *task*.

- Rincian Model:
 - Model: Master/Requirement/ResearchReqOutput.
 - Sertakan `const CATEGORY = ['wajib', 'tambahan']`.
 - Relasi di Model Master/Requirement/ResearchReqOutput:
 - belongsTo:
 - Relasi 1:
 - Nama Relasi: `rroutput_researchscheme`.
 - Kolom Terkait: `research_scheme_id`.
 - Model Target: `/Master/ResearchScheme`.
 - Relasi 2:
 - Nama Relasi: `rroutput_output`.
 - Kolom Terkait: `output_id`.
 - Model Target: `/Master/Output`.

Gambar 4.5.30 adalah *task* untuk membuat model Prasyarat Luaran Penelitian. Pada task ini, penulis membuat model untuk tabel `mst_research_req_outputs` bernama `ResearchReqOutput`.

```

1  <?php
2
3  namespace App\Models\Master\Requirement;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\SoftDeletes;
7  use Illuminate\Database\Eloquent\Model;
8  use App\Models\Master\ResearchScheme;
9  use App\Models\Master\Output;
10
11 class ResearchReqOutput extends Model
12 {
13     use HasFactory;
14     use SoftDeletes;
15     protected $fillable = [
16         'research_scheme_id',
17         'output_id',
18         'category',
19         'output_title',
20         'min_grade',
21         'max_grade'
22     ];
23     protected $table = "mst_research_req_outputs";
24     const CATEGORY = ['wajib', 'tambahan'];
25     public function rroutput_researchscheme(){
26         return $this->belongsTo(ResearchScheme::class,"research_scheme_id");
27     }
28     public function rroutput_output(){
29         return $this->belongsTo(Output::class,"output_id");
30     }
31 }
32

```

Gambar 4.5.29 adalah kode program yang penulis kerjakan untuk model ResearchReqOutput. Terdapat kode program untuk kolom tabel yang dapat diisi secara manual serta relasi tabel yang diperlukan.

4.2.1 Tahap Proposal

Tahap Proposal adalah tahap pertama pada skema penelitian. Tahapan ini diawal dengan dosen peneliti melakukan pengajuan proposal penelitian hingga proposal tersebut nantinya direview oleh dosen reviewer.

4.2.1 Pengajuan Proposal Penelitian

Pada halaman *dashboard*, terdapat sebuah *sidebar* di mana pada *sidebar* tersebut berisi menu-menu. Dosen terlebih dahulu klik Menu “Skema Penelitian”, kemudian klik SubMenu “Daftar Penelitian (Peneliti)”. Setelah itu akan muncul halaman “Daftar Penelitian”, di mana pada halaman tersebut terdapat filter yang nantinya dapat memunculkan daftar penelitian berdasarkan filter yang dipilih ke dalam bentuk table. Pada bagian kanan atas terdapat tombol “Ajukan Proposal Penelitian”. Ketika diklik, maka akan muncul sebuah modal yang berisi form pengajuan proposal penelitian. Dosen harus mengisi form tersebut. Jika sudah, klik tombol “Ajukan” pada bagian bawah modal. Lalu dosen akan diarahkan ke halaman “Pengelolaan Draft Proposal Penelitian”. Pada halaman tersebut, terdapat beberapa panel yang berisi form untuk melengkapi data untuk pengajuan proposal. Dosen harus mengisi terlebih dahulu data-data sesuai dengan form tersebut. Dosen harus memiliki minimal 3 bidang keilmuan yang sesuai dengan penelitian yang akan dilakukan pada Panel Kanan 2. Kemudian klik Tombol "Ajukan Proposal" pada Panel Kanan 1 sebelum masa pengajuan proposal berakhir. Maka akan muncul Pop-Up, klik "Ajukan". Jika proses pengajuan berhasil (tidak ada prasyarat yang terlewat / belum diisi), maka dosen akan diarahkan ke Halaman "Detail Penelitian". Saat ini, dosen peneliti akan menunggu hasil pengecekan proposal penelitian yang akan dilakukan oleh pihak LPPM. Permintaan revisi atau persetujuan proposal akan diinformasikan melalui notifikasi email ke email

terdaftar (yang digunakan untuk login ke sistem). Status penelitian saat ini adalah: "proposal - pengecekan kelengkapan".

Admin LPPM kemudian akan mengecek kelengkapan proposal penelitian tersebut. Terdapat 2 cara untuk melakukan pengecekan kelengkapan proposal penelitian. Cara pertama adalah Klik Tombol "Cek Kelengkapan Proposal" pada Halaman "Daftar Penelitian", setelah menerapkan filter Fase = 'proposal'. Cara kedua adalah Klik langsung dari Notifikasi pada Header (cara ini hanya dapat dilakukan 1 kali saja). Kemudian admin LPPM dapat melakukan pengecekan kelengkapan proposal penelitian yang terdiri atas 6 panel. Jika pengajuan proposal penelitian tersebut membutuhkan revisi, maka admin perlu Klik Tombol "Perlu Revisi" pada Panel Kanan 1, kemudian akan muncul sebuah form. Admin perlu menulis secara detail mengenai bagian-bagian yang memerlukan revisi, kemudian klik Tombol "Ajukan Permintaan untuk Revisi". Admin LPPM harus menunggu hingga revisi selesai dilakukan oleh dosen peneliti yang bersangkutan agar dapat mengambil keputusan berikutnya. Jika pengajuan proposal penelitian tersebut diterima, maka klik Tombol "Terima" pada Panel Kanan 1. Kemudian akan muncul form berikut. Klik Tombol "Terima Proposal". Kemudian admin LPPM akan diarahkan untuk mengalokasikan / menugaskan Reviewer. Jika pengajuan proposal penelitian tersebut ditolak, klik tombol "Tolak" pada Panel Kanan 1. Kemudian akan muncul form berikut. Admin perlu menuliskan alasan penolakan, kemudian klik Tombol "Tolak Proposal".

Jika admin LPPM meminta dosen peneliti untuk melakukan revisi terhadap proposal penelitian yang telah diajukan, maka dosen peneliti harus melakukan revisi terhadap proposal penelitian tersebut. Agar dapat melakukan revisi, dosen peneliti harus mengakses ke halaman "Proposal Penelitian". Terdapat 2 cara untuk mengakses halaman tersebut. Cara pertama adalah dosen peneliti mengakses halaman "Daftar Penelitian (Sebagai Peneliti)". Setelah diterapkan filter, lalu klik Tombol "Kelola Revisi" pada salah satu penelitian yang akan direvisi proposalnya. Cara kedua adalah klik langsung dari Notifikasi pada Header (cara ini hanya dapat dilakukan 1 kali saja). Jika sudah melakukan salah satu cara tersebut, maka dosen peneliti diarahkan ke halaman "Proposal Penelitian". Pada halaman ini, terdapat beberapa panel yang berisi form. Kemudian dosen peneliti harus melakukan revisi

terhadap bagian yang diminta oleh admin LPPM, kemudian klik Tombol "Konfirmasi Revisi". Checklist pertanggungjawaban pada form, kemudian klik Tombol "Konfirmasi". Kemudian dosen peneliti akan diarahkan ke halaman "Rincian Penelitian". Dosen peneliti harus menunggu hasil pengecekan proposal penelitian yang akan dilakukan oleh pihak LPPM. Permintaan revisi atau persetujuan proposal akan diinformasikan melalui notifikasi email ke email terdaftar (yang digunakan untuk login ke sistem). Status penelitian saat ini adalah: "proposal - pengecekan kelengkapan".

Jika admin LPPM menerima proposal penelitian tersebut, maka dosen peneliti akan menerima notifikasi baik melalui email maupun di web yang mengabarkan bahwa proposal penelitian telah diterima. Saat ini, dosen peneliti harap menunggu hasil review proposal penelitian yang akan dilakukan oleh pihak reviewer. Dosen peneliti akan diberi kabar via notifikasi email maupun notifikasi di web setelah proses review selesai. Status penelitian saat ini adalah: "proposal - belum direview".

4.2.2 Review Proposal Penelitian

Proposal Penelitian yang telah diterima/disetujui oleh admin LPPM selanjutnya akan direview oleh 2 dosen *reviewer*. Dosen *reviewer* dipilih oleh admin LPPM. Agar dapat memilih *reviewer*, admin LPPM perlu mengakses halaman "Alokasi Reviewer Skema Penelitian". Halaman tersebut dapat diakses ketika admin LPPM mengklik tombol "Terima" pada Panel Kanan 1 pada halaman "Pengecekan Kelengkapan Proposal Penelitian". Halaman "Alokasi Reviewer Skema Penelitian" juga dapat diakses dengan cara mengakses halaman "Daftar Penelitian", kemudian klik Tombol "Cek Kelengkapan Proposal" pada salah satu item yang memiliki status 'proposal - belum direview'. Kemudian admin LPPM akan diarahkan ke halaman berikut. Klik Tombol "Kelola Reviewer". Kemudian admin LPPM akan diarahkan ke Halaman "Alokasi Reviewer". Lalu admin LPPM mengklik Tombol "Tambah Reviewer" untuk memunculkan modal berisi form "Tambah Reviewer". Pada form tersebut, admin LPPM akan menerapkan filter berdasarkan Fakultas dan Program Studi, lalu mengklik Tombol "Cari Reviewer".

Jika terdapat *reviewer* yang cocok, maka nama *reviewer* terpilih akan muncul di Input "Reviewer". Admin LPPM kemudian mengklik tombol "Simpan" untuk memilih *reviewer* tersebut. Dosen yang ditunjuk sebagai *reviewer* akan mendapat notifikasi.

Dosen *reviewer* yang mendapat notifikasi dapat mengklik langsung dari Notifikasi pada Header (cara ini hanya dapat dilakukan 1 kali saja) untuk melakukan review proposal penelitian. Selain cara tersebut, dapat juga dilakukan dengan cara mengklik Tombol "Lakukan Review" pada salah satu penelitian di Halaman "Daftar Penelitian (Sebagai Reviewer)". Sebelumnya, untuk memunculkan daftar penelitian, dosen *reviewer* terlebih dahulu menerapkan *filter*. Jika sudah melakukan salah satu cara tersebut, dosen *reviewer* akan diarahkan ke halaman "Review Proposal Penelitian". Halaman "Review Proposal Penelitian" (sebelum generate formulir review) terdiri atas 3 panel. Untuk melakukan review terhadap proposal penelitian, dosen *reviewer* perlu untuk mengklik tombol "Generate Formulir Review" pada Panel Kiri 2. Kemudian akan muncul Formulir Review. Pada formulir ini, dosen *reviewer* dapat melakukan penilaian. Setelah penilaian dilakukan, dosen *reviewer* mengklik Tombol "Kalkulasi Skor" untuk melihat total skor hasil penilaiannya. Setelah itu, dosen *reviewer* dapat menuliskan rekomendasi berdasarkan hasil review. Jika memerlukan perbaikan, maka dosen *reviewer* dapat mengklik tombol "Perbaiki (Tolak)". Apabila proposal diterima, maka dosen *reviewer* dapat mengklik tombol "Lanjutkan (Terima)". Selanjutnya, dosen *reviewer* akan diarahkan ke halaman "Hasil Review Proposal Penelitian".

Jika kedua dosen *reviewer* memberikan keputusan "Terima", maka admin LPPM menerima notifikasi untuk melanjutkan ke tahap Alokasi Dana Tahap 1. Namun apabila terdapat dosen *reviewer* yang memberikan keputusan "Tolak", maka admin LPPM akan mengalokasikan 1 orang *reviewer* lagi untuk melakukan *review* Proposal Penelitian. Jika *reviewer* tersebut memberikan keputusan "Terima", maka admin LPPM menerima notifikasi untuk melanjutkan ke tahap Alokasi Dana Tahap 1. Namun jika *reviewer* tersebut yang memberikan keputusan "Tolak", maka admin LPPM harus memutuskan proposal penelitian akan dilanjutkan atau tidak. Proposal penelitian yang telah diterima oleh *reviewer* selanjutnya akan masuk ke tahap di mana admin LPPM melakukan proses Alokasi Dana Tahap 1.

Setelah proses *review* selesai dilakukan, maka dosen peneliti akan mendapat notifikasi mengenai hasil *review* (via *email* ataupun notifikasi di web). Dosen peneliti dapat mengklik notifikasi untuk melihat hasil *review* proposal penelitian (diarahkan ke Halaman "Rincian Penelitian"). Selain cara tersebut, dosen peneliti juga dapat mengetahui hasil *review* dengan mengklik Tombol "Lihat Detail" pada Halaman "Daftar Penelitian (Sebagai Peneliti)". Pada halaman "Rincian Penelitian", pada bagian panel kanan 1, terdapat hasil *review* proposal.

4.3 Tahap Pencairan Dana Tahap 1

Pencairan Dana Tahap 1 dilakukan oleh admin LPPM. Untuk dapat melakukan pencairan dana tahap 1, pada Sidebar Menu, admin LPPM dapat mengklik Menu "Skema Penelitian", kemudian klik Submenu "Alokasi Dana". Kemudian admin LPPM men-*set filter*, kemudian mengklik Tombol "Set Alokasi Dana" pada penelitian yang akan diset alokasi dananya. Kemudian akan muncul modal berisi form "Alokasi Dana Penelitian". Admin LPPM akan melengkapi data pada form tersebut, kemudian mengklik tombol "Simpan". Kemudian akan terkirim notifikasi kepada dosen peneliti melalui email dan notifikasi pada web. Selanjutnya admin LPPM beserta *reviewer* akan menunggu hingga Laporan Kemajuan Penelitian diunggah oleh dosen peneliti.

4.4 Tahap Laporan Kemajuan

Tahap selanjutnya adalah tahap laporan kemajuan. Tahap ini diawali dengan dosen peneliti mengajukan laporan kemajuan penelitian hingga laporan kemajuan penelitian tersebut nantinya di-*review* oleh dosen *reviewer*.

4.4.1 Pengajuan Laporan Kemajuan Penelitian

Dosen peneliti pertama-tama membuka halaman "Daftar Penelitian (Sebagai Peneliti)", kemudian menerapkan filter dengan Fase = "proposal", kemudian klik Tombol "Lihat Detail" pada penelitian yang memiliki status yaitu "proposal - alokasi dana" atau "proposal - pencairan dana tahap 1". Kemudian, pada Halaman "Rincian Penelitian", pada Panel Kanan 1, dosen peneliti mengklik Tombol "Unggah & Kirim Laporan Kemajuan". Kemudian dosen peneliti akan

mengunggah File Laporan Kemajuan Penelitian dan mencentang pernyataan pada form. Dosen peneliti masih bisa mengedit (unggah ulang) File Laporan Kemajuan Penelitian selama periode pengumpulan Laporan Kemajuan Penelitian belum berakhir.

4.4.2 Review Laporan Kemajuan Penelitian

Laporan Kemajuan Penelitian yang telah diunggah oleh dosen peneliti kemudian akan di-review oleh dosen *reviewer*. Untuk dapat me-review laporan kemajuan penelitian, dosen *reviewer* dapat masuk ke halaman "Daftar Penelitian (Sebagai Reviewer)", kemudian menerapkan filter dengan Fase = 'laporan kemajuan'. Kemudian mengklik tombol "Review Laporan Kemajuan" pada salah satu penelitian yang memiliki Status = 'laporan kemajuan - belum monev'. Kemudian dosen *reviewer* akan diarahkan ke halaman "Review Laporan Kemajuan Penelitian". Halaman "Review Laporan Kemajuan Penelitian" (sebelum generate formulir review) terdiri atas 3 panel, seperti pada Halaman "Review Proposal Penelitian". Kemudian dosen *reviewer* dapat mengklik tombol "Generate Formulir Review Laporan Kemajuan" pada Panel Kiri 2. Kemudian akan muncul formulir *review* dan dosen *reviewer* dapat segera melakukan penilaian. Setelah penilaian selesai dilakukan, dosen *reviewer* dapat mengklik tombol "Kalkulasi Skor" untuk melihat total skor hasil penilaiannya. Selanjutnya dosen *reviewer* akan menuliskan rekomendasi berdasarkan hasil review, kemudian mengklik Tombol "Perbaiki (Tolak)" jika ingin menolak atau tombol "Lanjutkan (Terima)" jika ingin menerima laporan kemajuan penelitian tersebut. Kemudian dosen *reviewer* akan diarahkan ke halaman "Hasil Review Laporan Kemajuan Penelitian".

Jika kedua dosen *reviewer* menerima/menyetujui laporan kemajuan penelitian tersebut, maka dosen peneliti akan menerima notifikasi bahwa review Laporan Kemajuan Penelitian telah selesai dilakukan, dan dosen (peneliti) sudah dapat menyerahkan Laporan Akhir Penelitian dan/atau Draft Luaran Penelitian. Dosen peneliti dapat mengklik notifikasi untuk melihat rincian hasil review Laporan Kemajuan Penelitian, kemudian akan diarahkan ke Halaman "Rincian Penelitian". Dosen peneliti bisa juga melihat rincian hasil review Laporan Kemajuan Penelitian dengan mengklik Tombol "Lihat Detail" pada Halaman

"Daftar Penelitian (Sebagai Peneliti)". Pada Halaman "Rincian Penelitian", klik Tombol "Lihat" pada Panel Kanan 4 (Panel "Hasil Review Laporan Kemajuan"). Kemudian dosen peneliti akan diarahkan ke halaman "Rincian Hasil Review Laporan Kemajuan Penelitian".

Namun jika terdapat dosen *reviewer* yang menolak/memberikan hasil negatif (tidak lulus) terhadap laporan kemajuan penelitian tersebut, maka admin LPPM harus mengambil keputusan apakah penelitian layak untuk dilanjutkan atau harus di-*takedown*.

4.5 Tahap Laporan Akhir

Jika laporan kemajuan penelitian tersebut telah diterima/disetujui oleh semua dosen reviewer, maka selanjutnya dosen peneliti dapat mengunggah Laporan Akhir Penelitian dan/atau Draft Kemajuan Penelitian.

BAB 5

SIMPULAN DAN SARAN

Pada bab ini akan membahas mengenai simpulan dan saran dari kegiatan MBKM Magang di i2c Studio yang sudah penulis lalui.

5.1.1 Simpulan

Dari kegiatan magang MBKM di i2c Studio yang penulis lalui, ada beberapa hal yang penulis dapat simpulkan, di antaranya:

1. Dengan membuat sistem informasi yang dapat membantu dan menunjang kegiatan LPPM Maranatha terutama dalam pengelolaan skema penelitian, maka dapat memudahkan pekerjaan baik itu pihak LPPM dan pihak lainnya yang berkaitan dengan skema penelitian, sehingga membuat kinerja menjadi lebih efektif dan efisien terutama pada skema penelitian.
2. Dengan menjelaskan mengenai alur dan proses yang terjadi dalam skema penelitian pada SIPPM, maka baik itu pihak LPPM, dosen, dan pihak lainnya yang terkait dapat memahami alur dan proses yang terjadi dalam skema penelitian pada SIPPM, sehingga dapat mengetahui penyebab dan akibat yang terjadi pada skema penelitian.

5.1.2 Saran

Saran yang dapat penulis berikan khususnya untuk pihak i2c Studio berdasarkan pengalaman penulis ketika mengikuti magang di i2c Studio adalah pada saat melakukan review pada task yang telah dikerjakan, mohon untuk Senior Programmer melakukan review dengan lebih teliti lagi. Seringkali penulis masih menemukan error pada kode program yang telah di-review. Selain itu, pada saat Senior Programmer memberikan task kepada penulis, mohon agar task yang diberikan tidak ada kesalahan penulisan agar ke depannya tidak terjadi kesalahpahaman dan mempermudah pekerjaan. Penulis juga menyarankan agar hasil pekerjaan untuk master data modul penelitian dapat menjadi dasar pengelolaan penelitian oleh LPPM di kemudian hari.

DAFTAR PUSTAKA

- [1] A. Faradilla, "Apa Itu PHP? Pengertian PHP untuk Pemula," 18 Januari 2023. [Online]. Available: <https://www.hostinger.co.id/tutorial/apa-itu-php/>. [Diakses 10 Mei 2023].
- [2] PT Biznet GIO Nusantara, "Mengenal Laravel, Framework PHP untuk membuat Aplikasi Web!," [Online]. Available: <https://www.biznetgio.com/news/apa-itu-laravel>. [Diakses 10 Mei 2023].
- [3] M. A. Maksum, "Apa itu Laravel? Pengertian, Fitur dan Kelebihannya," 10 Juni 2022. [Online]. Available: <https://www.dewaweb.com/blog/apa-itu-laravel/>. [Diakses 2023 Mei 11].
- [4] Agustini and W. J. Kurniawan, "Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi," *Sistem E-Learning Do'a dan Iqro' dalam Peningkatan Proses*, p. 155, 2019.
- [5] J. Terra, "Cara Menggunakan Desktop GitHub: Tutorial Desktop GitHub," 24 Februari 2023. [Online]. Available: <https://www.simplilearn.com/how-to-use-github-desktop-tutorial-article>. [Accessed 21 Mei 2023].
- [6] E. Santi, "PostgreSQL Adalah: Pengertian, Fungsi, Kelebihannya," 19 Maret 2023. [Online]. Available: <https://idwebhost.com/blog/postgresql-adalah/>. [Accessed 15 Mei 2023].
- [7] M. R. Adani, "Mengulas Penggunaan HTML dan Cara Kerjanya pada Web Development," 15 Februari 2021. [Online]. Available: <https://www.sekawanmedia.co.id/blog/pengertian-html/>. [Diakses 30 Mei 2023].
- [8] G. A. Sinaga, "Apa itu composer, cara menginstall composer dan laravel," 15 September 2022. [Online]. Available: <https://itcc.itpln.ac.id/apa-itu-composer-cara-menginstall-composer-dan-laravel/>. [Diakses 2 Juni 2023].
- [9] Redaksi Jagoan Hosting, "Apa itu Node.js? Contoh Penggunaan, Fitur & Cara Install," 7 Maret 2022. [Online]. Available:

- <https://www.jagoanhosting.com/blog/node-js-adalah/>. [Diakses 2 Juni 2023].
- [10] F. A., “Apa Itu Bootstrap? Pengertian, Fungsi, dan Kelebihannya,” 21 February 2023. [Online]. Available: <https://www.hostinger.co.id/tutorial/apa-itu-bootstrap>. [Diakses 10 Mei 2023].
- [11] R. F. Rahman, “Tutorial mengintegrasikan Laravel dengan Core Ui menggunakan fitur Laravel Mix,” 21 February 2020. [Online]. Available: <https://medium.com/@cahrizki.rf/tutorial-mengintegrasikan-laravel-dengan-core-ui-menggunakan-fitur-laravel-mix-46fa98b2ca1f>. [Diakses 10 Mei 2023].
- [12] B. Kristian, “Cara Membuat Tampilan Pop-up Alert Menggunakan SweetAlert,” 23 Desember 2021. [Online]. Available: <https://www.gamelab.id/news/1131-cara-membuat-tampilan-pop-up-alert-menggunakan-sweetalert#:~:text=SweetAlert%20merupakan%20library%20JavaScript%20yang,menulis%20menggunakan%20fungsi%20bawaan%20JavaScript..> [Diakses 10 Mei 2023].
- [13] Sutriman, “Cara membuat autocomplete Select2 dengan Ajax,” 12 Juli 2019. [Online]. Available: <https://sutriman.com/cara-membuat-autocomplete-select2-dengan-ajax/>. [Diakses 10 Mei 2023].
- [14] A. Yani dan B. Saputra, “RANCANG BANGUN SISTEM INFORMASI EVALUASI SISWA DAN KEHADIRAN GURU BERBASIS WEB,” *Studi Kasus di SMK Nusa Putra Kota Tangerang*, vol. 11, p. 111, 2018.
- [15] Y. K., “Pengertian jQuery Serta Fungsi dan Contohnya,” 25 Mei 2019. [Online]. Available: <https://www.niagahoster.co.id/blog/jquery-adalah/>. [Diakses 10 Mei 2023].
- [16] N. Akbar, “CARA INSTALL LARAVEL COLLECTIVE/HTML PADA LARAVEL 5,” 9 Maret 2017. [Online]. Available: <https://belajarphp.net/cara-install-laravelcollectivehtml-pada-laravel-5/#:~:text=Laravelcollective%2Fhtml%20adalah%20sebuah%20package,>

mudah%20digunakan%20untuk%20keperluan%20tertentu.. [Diakses 10 Mei 2023].

- [17] Dewaweb Team, “Penjelasan Lengkap Apa Itu CDN,” 28 Januari 2021. [Online]. Available: <https://www.dewaweb.com/blog/penjelasan-lengkap-apa-itu-cdn/>. [Diakses 30 Mei 2023].
- [18] LP2MP Universitas Medan Area , “Apa Itu Trello? Fitur dan Manfaat Trello,” 23 Juni 2022. [Online]. Available: <http://lp2mp.uma.ac.id/apa-itu-trello-fitur-dan-manfaat-trello/>. [Diakses 1 Juni 2023].
- [19] B. Telkom, "Pengertian Data Table," 23 Maret 2017. [Online]. Available: <http://blogblctelkom.blogspot.com/2017/03/pengertian-data-table.html>. [Accessed 17 Mei 2023].

LAMPIRAN A LOG KERJA

Februari			
Hari	Tanggal	Jumlah Jam	Pekerjaan
Kamis	9-Feb-23	2:00	mempelajari instalasi dan konfigurasi PHP
Sabtu	11-Feb-23	1:30	menonton dan mempelajari tutorial penggunaan laravel di youtube
Selasa	14-Feb-23	8:00	mempelajari model,view,controller,database
Rabu	15-Feb-23	8:00	mempelajari input database dan memunculkan database dan datatable serta delete
Kamis	16-Feb-23	9:00	mempelajari delete,update,dan master
Jumat	17-Feb-23	9:00	mempelajari factory dan sweetalert
Sabtu	18-Feb-23	2:00	mereview kembali materi yang telah dipelajari
Senin	20-Feb-23	9:00	mempelajari ajax untuk edit dan delete serta navbar
Selasa	21-Feb-23	9:00	mempelajari return with message,relasi antar tabel,dan select2 ,serta fetch,update,delete dari relasi tabel tersebut
Rabu	22-Feb-23	10:00	mempelajari autentikasi login dan signup dengan modal,instalasi postgre dan membuat database dengan postgre
Kamis	23-Feb-23	8:30	mempelajari storage untuk upload gambar
Jumat	24-Feb-23	8:00	mempelajari penggunaan middleware
Sabtu	25-Feb-23	3:00	Mereview kembali materi yang telah dipelajari
Minggu	26-Feb-23	3:00	Melanjutkan review materi
Senin	27-Feb-23	8:00	mempelajari penggunaan collective form pada laravel
Selasa	28-Feb-23	9:00	mengulang materi collective laravel,mencoba mengerjakan kuis laravel

Maret			
Hari	Tanggal	Jumlah Jam	Pekerjaan
Rabu	1-Mar-23	9:00	mengerjakan soal mini projek1 dari Ko Julio
Kamis	2-Mar-23	9:00	mengerjakan soal mini projek2 dari Ko Julio
Jumat	3-Mar-23	9:00	melanjutkan mengerjakan soal mini projek2 dari Ko Julio
Sabtu	4-Mar-23	5:00	mereview materi yang telah dipelajari
Minggu	5-Mar-23	5:00	mereview materi yang telah dipelajari
Senin	6-Mar-23	9:00	memperbaiki soal mini projek1, mengerjakan mini projek3 dari Ko Julio
Selasa	7-Mar-23	9:00	mengerjakan soal mini projek3 dari Ko Julio
Rabu	8-Mar-23	9:00	mengerjakan soal mini projek3 dari Ko Julio
Kamis	9-Mar-23	9:00	mempelajari JQuery
Jumat	10-Mar-23	9:00	mempelajari JQuery
Sabtu	11-Mar-23	5:00	mereview materi yang telah dipelajari
Minggu	12-Mar-23	5:00	mereview materi yang telah dipelajari
Senin	13-Mar-23	9:00	install node.js,coreui,visual studio comunity
Selasa	14-Mar-23	9:00	mempelajari coreui
Rabu	15-Mar-23	9:00	mempelajari coreui
Kamis	16-Mar-23	9:00	mempelajari coreui
Jumat	17-Mar-23	9:00	mempelajari coreui
Sabtu	18-Mar-23	5:00	mempelajari coreui
Minggu	19-Mar-23	5:00	mempelajari coreui
Senin	20-Mar-23	8:00	mempelajari coreui
Selasa	21-Mar-23	8:00	mempelajari coreui
Kamis	23-Mar-23	8:00	mempelajari coreui
Sabtu	25-Mar-23	5:00	mempelajari coreui
Minggu	26-Mar-23	5:00	mempelajari coreui
Senin	27-Mar-23	9:00	melakukan instalasi projek LPPM,composer,dan npm,Membuat migrasi dan model master data 3
Selasa	28-Mar-23	9:00	memperbaiki kesalahan pembuatan migrasi dan model master data 3 berdasarkan hasil review
Rabu	29-Mar-23	9:00	mempelajari kembali CRUD pada laravel

Maret			
Hari	Tanggal	Jumlah Jam	Pekerjaan
Kamis	30-Mar-23	6:00	mempelajari API pada laravel
Jumat	31-Mar-23	8:00	mempelajari API pada laravel

April			
Hari	Tanggal	Jumlah Jam	Pekerjaan
Sabtu	1-Apr-23	4:00	mempelajari API pada laravel
Senin	3-Apr-23	8:00	mempelajari testing pada laravel
Selasa	4-Apr-23	8:00	mempelajari testing pada laravel
Rabu	5-Apr-23	8:00	mengerjakan [Master Data I] Master Program Mahasiswa
Kamis	6-Apr-23	8:00	mengerjakan [Master Data I] Master Program Mahasiswa
Sabtu	8-Apr-23	4:00	Mereview projek yang telah dikerjakan
Senin	10-Apr-23	8:00	Mengerjakan [Master Data II] Master Bidang Keahlian
Selasa	11-Apr-23	8:00	Mengerjakan [Master Data II] Master Bidang Keahlian
Rabu	12-Apr-23	8:00	Memperbaiki [Master Data II] Master Bidang Keahlian
Kamis	13-Apr-23	8:00	Mempelajari API token pada laravel
Jumat	14-Apr-23	8:00	Mempelajari API token pada laravel
Sabtu	15-Apr-23	4:00	Mereview projek yang telah dikerjakan
Senin	17-Apr-23	8:00	Mempelajari validasi request pada laravel
Selasa	18-Apr-23	8:00	Mempelajari layouting bootstrap
Rabu	19-Apr-23	8:00	Mempelajari bootstrap form
Rabu	26-Apr-23	8:00	Mempelajari komponen bootstrap
Kamis	27-Apr-23	8:00	Mempelajari komponen bootstrap

Mei			
Hari	Tanggal	Jumlah Jam	Pekerjaan
Senin	1-May-23	11:00	28 Apr Mengerjakan [Master Data - Prasyarat Skema Penelitian] Prasyarat Program Penelitian 8 jam, 29 Apr Mengerjakan [Master Data - Prasyarat Skema Penelitian] Prasyarat Program Penelitian 3 jam
Selasa	2-May-23	8:00	Mengerjakan [Master Data - Prasyarat Skema Penelitian] Prasyarat Program Penelitian
Rabu	3-May-23	8:00	Mempelajari multi select2 dan SWAL
Kamis	4-May-23	8:00	Mempelajari penggunaan attach dan compact serta SWAL pada laravel
Jumat	5-May-23	8:00	Mempelajari penggunaan detach compact serta SWAL pada laravel
Sabtu	6-May-23	4:00	Mereview proyek yang telah dikerjakan
Senin	8-May-23	8:00	Menyusun laporan KP
Selasa	9-May-23	8:00	Menyusun laporan KP, mengerjakan [Master Data - Prasyarat Skema Buku] Prasyarat Skema Buku
Rabu	10-May-23	8:00	Menyusun laporan KP, mengerjakan [Master Data - Prasyarat Skema Buku] Prasyarat Skema Buku
Kamis	11-May-23	8:00	Mengerjakan [Master Data - Prasyarat Skema Buku] Prasyarat Detail Buku
Jumat	12-May-23	8:00	Mengerjakan [Master Data - Prasyarat Skema Pelayanan Publik] Prasyarat Program Pelayanan Publik
Sabtu	13-May-23	4:00	Menyusun laporan KP
Senin	15-May-23	8:00	Mengerjakan [Master Data - Prasyarat Skema Karya Ilmiah] Prasyarat Detail Karya Ilmiah
Selasa	16-May-23	8:00	Mengerjakan [Master Data - Prasyarat Skema Karya Ilmiah] Prasyarat Detail Karya Ilmiah
Rabu	17-May-23	6:00	Menyusun laporan KP
Jumat	19-May-23	6:00	Menyusun laporan KP
Sabtu	20-May-23	6:00	Menyusun laporan KP
Senin	22-May-23	6:00	Revisi laporan KP
Selasa	23-May-23	6:00	Menyusun laporan KP dan mempelajari desain dan analisis proyek LPPM

Mei			
Hari	Tanggal	Jumlah Jam	Pekerjaan
Rabu	24-May-23	6:00	Revisi laporan KP
Kamis	25-May-23	8:00	Mengerjakan [Modul Questioner TKT] Migrasi & Model Part 1 dan rapat
Jumat	26-May-23	8:00	Melanjutkan mengerjakan [Modul Questioner TKT] Migrasi & Model Part 1
Sabtu	27-May-23	6:00	Revisi laporan KP
Senin	29-May-23	8:00	Mengerjakan [Modul Questioner TKT] Questioner TKT - Private Function checkUserAccess dan revisi laporan
Selasa	30-May-23	6:00	Revisi Laporan KP dan Menyusun PowerPoint KP
Rabu	31-May-23	6:00	Menyusun Powerpoint KP

RIWAYAT HIDUP PENULIS

1. Identitas Diri

Nama : Juan Sterling Martua
NRP : 2072009
Tempat, Tanggal Lahir : Bandung, 16 Februari 2002
Alamat : Komp. Mitra Dago Parahyangan
Jalan Anjung blok L no 5
Antapani, Bandung
No HP : 081809835640
Email : juansterling1409@gmail.com



2. Riwayat Pendidikan

SD : SD Santo Agustinus (2008 – 2014)
SMP : SMP Santa Maria (2014 – 2017)
SMA : SMA Santa Maria 1 (2017 – 2020)
S1 : Universitas Kristen Maranatha (2020 – sekarang)

3. Prestasi

- Penghargaan Dean's List Semester Ganjil tahun akademik 2020/2021
- Penghargaan Dean's List Semester Genap tahun akademik 2020/2021
- Penghargaan Dean's List Semester Ganjil tahun akademik 2021/2022
- Penghargaan Dean's List Semester Genap tahun akademik 2021/2022
- Penghargaan Dean's List Semester Ganjil tahun akademik 2022/2023

4. Riwayat Pekerjaan

- Staff laboratorium Fakultas IT Maranatha: Januari 2022 – Juli 2022
- Koordinator tim staff laboratorium Fakultas IT Maranatha: Juli 2022 – Januari 2023
- Staff laboratorium Fakultas IT Maranatha: Januari 2023 – Sekarang