

**SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI**

**RANCANG BANGUN APLIKASI LINK-MATCH STT-NF MODUL DOSEN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL**

**PROPOSAL TUGAS AKHIR**

**MUHAMMAD AZHAR RASYAD**

**0110217029**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**DEPOK**

**MARET 2020**



**SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI**

**RANCANG BANGUN APLIKASI LINK-MATCH STT-NF MODUL DOSEN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL**

**PROPOSAL TUGAS AKHIR**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar**

**Sarjana**

**MUHAMMAD AZHAR RASYAD**

**0110217029**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**DEPOK**

**MARET 2020**

# HALAMAN PERNYATAAN ORISINALITAS

**Tugas Akhir ini adalah hasil karya penulis,**

**dan semua sumber baik yang dikutip maupun dirujuk**

**telah saya nyatakan dengan benar.**

**Nama : Muhammad Azhar Rasyad**

**NIM : 0110217029**

**Tanda Tangan : ...............................**

**Tanggal : ...............................**

# HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh :

Nama : Muhammad Azhar Rasyad

NIM : 0110217029

Program Studi : Teknik Informatika

Judul : RANCANG BANGUN APLIKASI LINK-MATCH STT-NF MODUL DOSEN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL

**Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri**

**DEWAN PENGUJI**

|  |  |
| --- | --- |
| Pembimbing I  (Nama, gelar) | Pembimbing II  (Nama, gelar) |
| Penguji I  (Nama, gelar) | Penguji II  (Nama, gelar) |

Ditetapkan di : Depok

Tanggal : 21 Maret 2020

# KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah *Subhanahu wa Ta’ala*, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir ini. Tak lupa *Shalawat* serta salam penulis sampaikan kepada Nabi Muhammad *Shallallahu ‘Alaihi wa Sallam*.Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tugas akhir ini, sangatlah sulit bagi penulis untuk menyelesaikannya. Oleh karena itu, penulis mengucapkan terima kasih kepada :

1. Allah *Subhanahu wa Ta’ala*.
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Lukman Rosyidi, ST. MM. MT., selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Bapak Ahmad Rio Adriansyah, S.Si M.Si., selaku Ketua Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Bapak Sirojul Munir, S.Si. M.Kom., selaku Dosen Pembimbing Akademik dan selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
6. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
7. Karyawan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah meluangkan waktunya untuk memberikan data yang diperlukan bagi penulisan ilmiah ini.
8. Teman-teman Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah mendukung penulis dalam menyelesaikan penulisan ilmiah ini.
9. Seluruh pihak yang telah membantu secara langsung maupun tidak langsung, yang tidak dapat penulis sertakan satu persatu namun tidak mengurangi rasa terima kasih penulis.

Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah *Subhanahu wa Ta’ala* berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga Tugas Akhir ini membawa manfaat bagi pengembangan ilmu.

Depok, 21 Maret 2020

Penulis

# HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI

**TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Muhammad Azhar Rasyad

NIM : 0110217029

Program Studi : Teknik Informatika

Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Non-eksklusif (*Non-exclusive Royalty – Free Right*)** atas karya ilmiah saya yang berjudul :

RANCANG BANGUN APLIKASI LINK-MATCH STT-NF MODUL DOSEN BERBASIS WEB MENGGUNAKAN FRAMEWORK LARAVEL

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-eksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 21 Maret 2020

Yang Menyatakan

( Muhammad Azhar Rasyad )

# ABSTRAK

Nama : Muhammad Azhar Rasyad

NIM : 0110217029

Program Studi : Teknik Informatika

Judul :

Tugas Akhir ini membahas tentang

Kata kunci :

# ABSTRACT

Name : Muhammad Azhar Rasyad

NIM : 0110217029

Study Program : Teknik Informatika

Title :

The focus of final research is about

Key words :

# DAFTAR ISI

[HALAMAN PERNYATAAN ORISINALITAS i](#__RefHeading___Toc682_2304900861)

[HALAMAN PENGESAHAN ii](#__RefHeading___Toc1828_1702832820)

[KATA PENGANTAR iii](#__RefHeading___Toc1830_1702832820)

[HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI v](#__RefHeading___Toc593_4044772922)

[ABSTRAK vi](#__RefHeading___Toc595_4044772922)

[ABSTRACT vii](#__RefHeading___Toc597_4044772922)

[DAFTAR ISI viii](#__RefHeading___Toc599_4044772922)

[DAFTAR GAMBAR x](#__RefHeading___Toc603_4044772922)

[DAFTAR TABEL xi](#__RefHeading___Toc684_2304900861)

[BAB I 1](#__RefHeading___Toc740_2304900861)

[PENDAHULUAN 1](#__RefHeading___Toc1514_2193851436)

[1.1 Latar Belakang 1](#__RefHeading___Toc723_2304900861)

[1.2 Perumusan Masalah 3](#__RefHeading___Toc760_2304900861)

[1.3 Tujuan Penelitian 3](#__RefHeading___Toc762_2304900861)

[1.4 Manfaat Penelitian 4](#__RefHeading___Toc5804_3720861844)

[1.5 Batasan Masalah 4](#__RefHeading___Toc764_2304900861)

[1.6 Sistematika Penulisan 4](#__RefHeading___Toc768_2304900861)

[BAB II 6](#__RefHeading___Toc1209_2304900861)

[LANDASAN TEORI 6](#__RefHeading___Toc4085_2193851436)

[2.1 Tinjauan Pustaka 6](#__RefHeading___Toc1302_2304900861)

[2.1.1 *Link and Match* 6](#__RefHeading___Toc2005_946457605)

[2.1.2 Model Pengembangan 7](#__RefHeading___Toc2007_946457605)

[2.1.2.1 Scrum 7](#__RefHeading___Toc5245_2595941674)

[2.1.2.2 *Unified Modelling Language* 9](#__RefHeading___Toc5284_2595941674)

[2.1.3 *Tools* Pengembangan 11](#__RefHeading___Toc2009_946457605)

[2.1.3.1 PHP *Framework* Laravel 11](#__RefHeading___Toc5286_2595941674)

[2.1.3.2 PostgreSQL 14](#__RefHeading___Toc5288_2595941674)

[2.1.4 *Web Service* 15](#__RefHeading___Toc2011_946457605)

[2.1.5 Pengujian Sistem 16](#__RefHeading___Toc2013_946457605)

[2.1.5.1 *Black Box Testing* 16](#__RefHeading___Toc5290_2595941674)

[2.1.5.2 *User Acceptance Testing* 17](#__RefHeading___Toc5292_2595941674)

[2.1.5.4 Kuesioner 18](#__RefHeading___Toc5294_2595941674)

[2.2 Penelitian Terkait 19](#__RefHeading___Toc760_23049008613)

[2.2.1 Tabel Penelitian Terkait 19](#__RefHeading___Toc3286_2291303207)

[2.2.2 Posisi Penelitian 20](#__RefHeading___Toc3288_2291303207)

[BAB III 22](#__RefHeading___Toc1211_2304900861)

[METODE PENELITIAN 22](#__RefHeading___Toc4087_2193851436)

[3.1 Tahapan Penelitian 22](#__RefHeading___Toc1304_2304900861)

[3.1.1 Analisa Kebutuhan Sistem 22](#__RefHeading___Toc15548_3789637748)

[3.1.2 Perancangan Sistem 23](#__RefHeading___Toc15550_3789637748)

[3.1.3 Implementasi 23](#__RefHeading___Toc15552_3789637748)

[3.1.4 Pengujian dan Hasil 23](#__RefHeading___Toc15554_3789637748)

[3.2 Rancangan Penelitian 24](#__RefHeading___Toc760_23049008614)

[3.2.1 Jenis Penelitian 24](#__RefHeading___Toc762_23049008614)

[3.2.2 Lingkungan Pengembangan 24](#__RefHeading___Toc764_23049008614)

[3.2.3 Bahan dan Alat 24](#__RefHeading___Toc766_23049008614)

[3.3 Jadwal Penelitian 25](#__RefHeading___Toc15556_3789637748)

[BAB IV 26](#__RefHeading___Toc1213_2304900861)

[ANALISIS DAN PERANCANGAN 26](#__RefHeading___Toc4089_2193851436)

[4.1 Analisis Sistem 26](#__RefHeading___Toc1331_2304900861)

[4.1.1 *End User* 26](#__RefHeading___Toc5806_3720861844)

[4.1.2 *User Requirement* 26](#__RefHeading___Toc5808_3720861844)

[4.2 Perancangan Sistem 26](#__RefHeading___Toc760_230490086141)

[4.2.1 *Entity Relationship Diagram* 26](#__RefHeading___Toc5810_3720861844)

[4.2.2 *Use Case Diagram* 26](#__RefHeading___Toc5812_3720861844)

[4.2.3 *Activity Diagram* 26](#__RefHeading___Toc5814_3720861844)

[4.2.4 *User Interface* 26](#__RefHeading___Toc5816_3720861844)

[4.3 Rancangan Pengujian 26](#__RefHeading___Toc1335_2304900861)

[4.3.1 *Black Box Testing* 27](#__RefHeading___Toc5818_3720861844)

[4.3.2 *User Acceptance Testing* 27](#__RefHeading___Toc5820_3720861844)

[4.3.3 Postman 27](#__RefHeading___Toc5822_3720861844)

[4.3.4 Kuesioner 27](#__RefHeading___Toc5824_3720861844)

[BAB V 28](#__RefHeading___Toc1215_2304900861)

[KESIMPULAN DAN SARAN 28](#__RefHeading___Toc4091_2193851436)

[5.1 Kesimpulan 28](#__RefHeading___Toc1339_2304900861)

[5.2 Saran 28](#__RefHeading___Toc760_2304900861411)

[DAFTAR PUSTAKA 29](#__RefHeading___Toc1217_2304900861)

[LAMPIRAN 32](#__RefHeading___Toc1219_2304900861)

# DAFTAR GAMBAR

[Logo 1 1](#_toc111)

[Logo 2 2](#_toc146)

[Gambar 1 : Struktur Aplikasi Link-Match STT-NF 3](#_toc468)

[Gambar 2 : Link and Match STT-NF 6](#_toc21)

[Gambar 3 : Alur Kerja Metode Scrum 9](#_toc539)

[Gambar 4 : UML Diagram 10](#_toc546)

[Gambar 5 : Arsitektur Laravel 13](#_toc580)

[Gambar 6 : Konsep MVC 14](#_toc588)

[Gambar 7 : Arsitektur PostgreSQL 14](#_toc595)

[Gambar 8 : Arsitektur Web Service 16](#_toc604)

[Gambar 9 : Alur Kerja Black Box Testing Link-Match STT-NF 17](#_toc631)

[Gambar 10 : Alur Kerja User Acceptance Testing Link-Match STT-NF 18](#_toc637)

[Gambar 11 : Tahapan Penelitian 22](#_toc800)

[Gambar 12 : Jadwal Penelitian 25](#_toc845)

[Gambar 13 : Entity Relationship Diagram 27](#_toc871)

[Gambar 14 : Use Case Diagram 28](#_toc875)

[Gambar 15 : Use Case Diagram User Management 29](#_toc886)

[Gambar 16 : Use Case Diagram Team Management 30](#_toc901)

[Gambar 17 : Use Case Diagram Project Management 31](#_toc913)

[Gambar 18 : Activity Diagram 32](#_toc928)

[Gambar 19 : Activity Diagram Role Administrator 33](#_toc939)

[Gambar 20 : Activity Diagram Role Dosen 34](#_toc949)

[Gambar 21 : Activity Diagram Role Scrum Master 35](#_toc959)

[Gambar 22 : Activity Diagram Role Product Owner 36](#_toc963)

[Gambar 23 : Activity Diagram Role Mahasiswa 37](#_toc981)

[Gambar 24 : User Interface Login 38](#_toc993)

[Gambar 25 : User Interface Role Administrator 38](#_toc996)

[Gambar 26 : User Interface Role Dosen 39](#_toc1000)

[Gambar 27 : User Interface Role Scrum Master 39](#_toc1003)

[Gambar 28 : User Interface Role Product Owner 40](#_toc1006)

[Gambar 29 : User Interface Role Mahasiswa 40](#_toc1009)

# DAFTAR TABEL

Table1 ii

Tabel1:SkalaLikert 19

Tabel2:PenelitianTerkait 19

Tabel3:PosisiPenelitian 20

Tabel4:RancanganPengujianBlackBoxTesting 41

Tabel5:RancanganPengujianUserAcceptanceTesting 44

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang

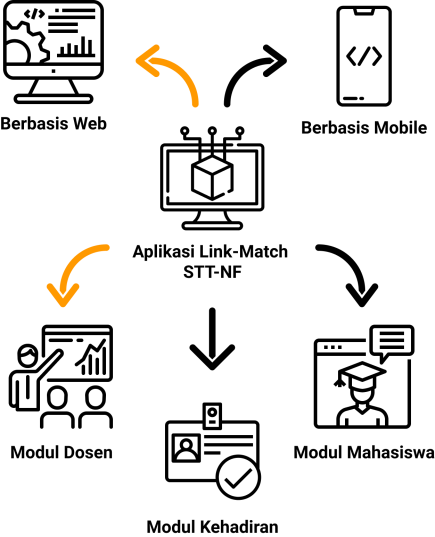
Menurut (Badan Pusat Statistik, 2019) pada Agustus 2019 di Indonesia, terdapat 133,56 juta orang merupakan angkatan kerja, dengan 7,05 juta orang atau sekitar 5% dinyatakan sebagai pengangguran. Jumlah tersebut memang sedikit dibandingkan dengan jumlah orang yang bekerja yaitu 95%, akan tetapi 5% tersebut tetaplah orang-orang yang membutuhkan penghasilan dari pekerjaan untuk memenuhi kebutuhan hidup sehari-hari. Berdasarkan Tingkat Pengangguran Terbuka (TPT) menurut jenjang pendidikan, persentase TPT tertinggi sebesar 10,42% pada jenjang pendidikan Sekolah Menengah Kejuruan (SMK).

Salah satu faktor tingginya pengangguran karena ketidakseimbangan antara permintaan dan penawaran tenaga kerja (Sholeh, 2007). Padahal tujuan dari pendidikan menengah kejuruan yaitu mengutamakan siswa untuk memasuki lapangan kerja berdasarkan Peraturan Pemerintah Republik Indonesia Nomor 29 Tahun 1990 Tentang Pendidikan Menengah. Di antara penyebab tingginya pengangguran pada jenjang pendidikan SMK yaitu karena adanya *mismatch*. *Mismatch* merupakan ketidaksesuaian kurikulum yang ada pada SMK dengan industri pekerjaan, sehingga industri pekerjaan menjadi kesulitan dalam menerima lulusan SMK yang memiliki kompetensi berbeda dari industrinya.

Departemen Pendidikan dan Kebudayaan Republik Indonesia membuat sebuah kebijakan program *Link and Match*. *Link and Match* merupakan program untuk mempelajari kompetensi-kompetensi apa saja yang dibutuhkan di industri pekerjaan. Target *Link and Match* sebenarnya tidak hanya untuk lulusan SMK melainkan juga untuk lulusan perguruan tinggi supaya dapat mengurangi jumlah pengangguran yang terus bertambah. Menurut Soemarso selaku Ketua Dewan Pembina Politeknik, *Link and match* pada perguruan tinggi diharapkan dapat menyesuaikan kurikulum dengan kebutuhan yang ada di industri pekerjaan (Disas, 2018).

Mulai September 2019 Sekolah Tinggi Teknologi Terpadu Nurul Fikri (STT-NF) sudah melaksanakan kebijakan *Link and Match* yang difasilitasi oleh *Student Learning and Incubator Business Center* (SLIBC) STT-NF. Konsep dari *Link and Match* STT-NF yaitu mengerjakan sebuah proyek yang diberikan oleh *product owner* selaku *client* dari industri pekerjaan (Sekolah Tinggi Teknologi Terpadu Nurul Fikri, 2019). Namun hal tersebut belum terealisasi dengan baik dikarenakan program *Link and Match* STT-NF baru berjalan dan membutuhkan beberapa sistem demi kelancaran program ini, salah satunya yaitu tersedianya aplikasiuntuk mendokumentasikan proyek-proyek apa saja yang telah dikerjakan mahasiswa STT-NF serta penghubung antara *product owner* dengan pihak STT-NF.

Oleh karena itu untuk kelancaran program tersebut maka diadakan sebuah penelitian tentang aplikasiLink-MatchSTT-NF, sehingga pihak perguruan tinggi ataupun pihak luar dapat dengan mudah melihat hasil dari *Link and Match* yang ada di STT-NF. Terlepas dari itu semua, pengembangan aplikasiini tidaklah mudah dikarenakan banyak pihak yang terlibat mulai dari administrator, dosen, asisten dosen, mahasiswa, dan *product owner*, sehingga diperlukan pengembangan yang kompleksitas agar dapat berjalan dengan baik. Dikarenakan hal tersebut pada aplikasi ini dibagi menjadi tiga bagian yaitu modul dosen, mahasiswa, dan kehadiran dengan dua platform diantaranya berbasis web dan berbasis *mobile*. Khusus pada penelitian ini akan dikembangkan aplikasiLink-Match STT-NF modul dosen yang berbasiskan web dengan fitur *user management, team management,* dan *project management* seperti yang dijelaskan pada gambar 1 berikut.



Gambar 1 : Struktur Aplikasi Link-Match STT-NF

## 1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah pada tugas akhir ini adalah “Bagaimana mengembangkan aplikasi berbasis web untuk mendukung program *Link and Match* di STT-NF pada modul dosen?”. Demi menjawab hal tersebut, maka ada beberapa hal yang perlu dikaji yaitu sebagai berikut :

1. Apa saja fitur yang tersedia pada aplikasi Link-Match STT-NF modul dosen?
2. Bagaimana proses pengembangan aplikasi Link-MatchSTT-NF modul dosen berbasis web menggunakan php framework laravel?
3. Apakah aplikasi Link-Match STT-NF modul dosen dapat mendukung program *Link and Match* di STT-NF?

## 1.3 Tujuan Penelitian

Penelitian ini memiliki tujuan untuk mendukung program *Link and Match* STT-NF, yaitu :

1. Menyediakan fitur pada aplikasi Link-Match STT-NF modul dosen.
2. Memberikan tahapan pengembangan aplikasi Link-Match STT-NF modul dosen.
3. Menjadi solusi dalam program *Link and Match* di STT-NF bagian modul dosen.

## 1.4 Manfaat Penelitian

Penelitian ini memiliki manfaat untuk mendukung program *Link and Match* STT-NF pada modul dosen, yaitu :

1. Mendukung program *Link and Match* STT-NF khusus modul dosen.
2. Memonitor proyek *Link and Match* STT-NF yang dikerjakan mahasiswa.
3. Mendokumentasikan proyek yang dikerjakan mahasiswa dalam program *Link and Match* STT-NF.
4. Mempermudah *user* admin dalam manajemen proyek *Link and Match* STT-NF.

## 1.5 Batasan Masalah

Pada tugas akhir ini terdapat batasan-batasan masalah yang perlu diperhatikan yaitu :

1. Penelitian ini mengembangkan aplikasi *Link and Match* di STT-NF.
2. Penelitian ini khusus dikembangkan dengan berbasis *website*.
3. *Website* *Link and Match* di STT-NF ini berfokus pada pengembangan modul dosen.
4. Modul dosen pada *website Link and Match* di STT-NF hanya digunakan oleh *user* administrator, dosen, asisten dosen dan *product owner*.
5. Implementasi pada penelitian ini hanya terbatas fitur manajemen *user*, manajemen tim, manajemen skor, dan memonitor proyek.

## 1.6 Sistematika Penulisan

Pada penelitian ini untuk memudahkan dalam memahami tugas akhir ini secara keseluruhan maka diperlukan sistematika penulisan yaitu:

1. BAB I PENDAHULUAN merupakan bab yang berisi pembuka dari penelitian ini yaitu latar belakang, perumusan masalah, tujuan dan manfaat, batasan masalah, metode penelitian, dan sistematika penulisan.
2. BAB II LANDASAN TEORI merupakan bab yang akan menjadi pedoman dasar dari penelitian ini yaitu tinjauan pustaka dan penelitian terkait.
3. BAB III METODE PENELITIAN merupakan bab yang berisi tahapan-tahapan dari penelitian ini yaitu tahapan penelitian, rancangan penelitian, solusi pemecahan masalah, jenis penelitian, metode pengumpulan data, lingkungan pengembangan, dan waktu penelitian.
4. BAB IV….
5. BAB V…

# BAB II

# LANDASAN TEORI

## 2.1 Tinjauan Pustaka

Tinjauan pustaka ini akan menjelaskan teori yang dijadikan sebagai acuan dari penelitian ini meliputi *Link and Match*, model pengembangan, *tools* pengembangan, *web service*, pengujian sistem.

### 2.1.1 *Link and Match*

Penelitian ini mempunyai fokus utama yaitu dalam mengembangkan sebuah program bernama *Link and Match.* Menurut(Disas, 2018) *Link and Match* merupakan program guna meningkatan relevansi antara kurikulum pendidikan di sekolah dengan kebutuhan di dunia kerja. Program ini merupakan salah satu kebijakan Departemen Pendidikan dan Kebudayaan Republik Indonesia dalam menekan angka pengangguran di Indonesia. Penekanan pengangguran yang dimaksud khusus sektor pendidikan karena kurikulum pendidikan sebelumnya belum sesuai dengan kondisi dunia kerja saat ini, hal ini yang disebut *education mismatch*.

Oleh karena itu salah satu perguruan tinggi swasta di Indonesia yaitu Sekolah Tinggi Teknologi Terpadu Nurul Fikri (STT-NF) di tahun 2019 mulai menerapkan program tersebut supaya lulusannya mendapatkan pembelajaran yang sesuai dengan dunia kerja saat ini, terlihat suasana program *Link and Match* di STT-NF pada gambar 2.

Gambar 2 : Link and Match STT-NF

Program *Link and Match* di STT-NF saat ini menggunakan metode scrum, harapannya setelah mengikuti program tersebut dapat menambah portofolio, mendapatkan pengalaman di dunia kerja sebelum lulus, dan memiliki kompetensi yang baik (Sekolah Tinggi Teknologi Terpadu Nurul Fikri, 2019). Dalam mendukung program tersebut dibuatlah sebuah aplikasi Link-Match STT-NF yang terdiri dari dua platform yaitu berbasis web dan mobile, serta dibagi menjadi tiga modul yaitu modul dosen, mahasiswa, dan kehadiran. Pada penelitian ini hanya berfokus pada pengembangan aplikasi Link-Match STT-NF yang berbasiskan web dengan modul dosen.

### 2.1.2 Model Pengembangan

Penelitian ini menggunakan dua model pengembangan yaitu scrum dan UML.

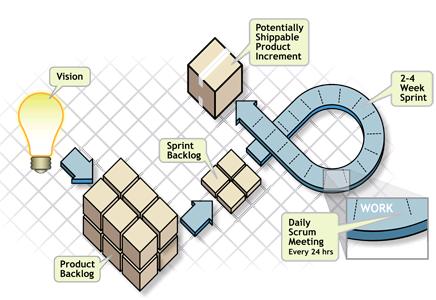
#### 2.1.2.1 Scrum

Menurut (Azdy & Azhari, 2012) scrum merupakan model pengembangan yang bersifat fleksibel dalam mengatasi permasalahan sebuah produk atau dalam penelitian ini disebut sebagai aplikasi yang sedang dikembangkan secara kompleks. Model pengembangan scrum ini sering digunakan karena permasalahan-permasalahan yang dialami ketika mengembangkan suatu aplikasi seperti berikut :

1. Adanya ketidakpastian dalam proses pengembangan aplikasi.
2. Kebutuhan-kebutuhan pada aplikasi tidak dapat diketahui secara detail sampai aplikasi tersebut digunakan secara langsung.
3. Menspesifikasikan suatu aplikasi secara menyeluruh diperlukan adanya interaksi antara pengguna dan aplikasi tersebut.
4. Kebutuhan aplikasi yang tidak jelas, sering berubah-ubah, dan teknologi yang terus berkembang pesat dapat menjadikan pengembangan aplikasi tidak dapat diprediksi dengan tepat.

Permasalahan di atas dapat ditanggulangi dengan adanya model pengembangan yaitu scrum, karena setiap waktunya terus menganalisa kebutuhan aplikasi secara detail dan menyeluruh. Scrum dapat dikatakan suatu metode dalam mengembangkan aplikasi yang di dalamnya terdapat proses pengelolaan agar dapat menghasilkan aplikasi yang tepat. Pengelolaan metode scrum tersebut terdiri dari beberapa komponen yaitu :

1. *Role*, dalam scrum terdapat peran-peran bagi para pengembang diantaranya *product owner*, *scrum master*, dan tim pengembang, berikut masing-masing penjelasannya :
   * *Product owner* berperan sebagai penanggung jawab terhadap aplikasi yang sedang dikembangkan.
   * *Scrum master* berperan sebagai pengontrol dari tim pengembang aplikasi agar setiap individu dapat berkontribusi secara maksimal.
   * Tim pengembang berperan secara langsung dalam implementasi pengembangan sebuah aplikasi.
2. *Artifact*, merupakan bahan-bahan yang perlu disiapkan dalam pengembangan aplikasi dengan metode scrum dan terdiri dari *product backlog, sprint backlog,* dan *burndown chart*, berikut masing-masing penjelasannya :
   * *Product backlog* yaitu daftar keseluruhan tugas yang harus dikerjakan dalam pengembangan aplikasi.
   * *Sprint* yaitu rentang waktu yang telah ditentukan dalam menyelesaikan tugas yang ada pada pengembangan aplikasi. *Sprint backlog* yaitu sebagian daftar tugas yang telah ditentukan dari *product backlog* untuk dikerjakan dalam waktu satu *sprint*.
   * *Burndown chart* yaitu menampilkan pekerjaan yang belum terselesaikan dalam *product backlog*.
3. *Event*, dalam pengelolaan suatu aplikasi dengan metode scrum terdapat banyak kegiatan berupa *sprint planning*, *daily scrum*, *sprint review*, dan *sprint retrospective*, berikut masing-masing penjelasannya :
   * *Sprint planning* yaitu merencanakan tugas-tugas yang akan dikerjakan selama satu *sprint*.
   * *Daily scrum* yaitu kegiatan yang setiap harinya melaporkan progres tugas yang sedang dikerjakan.
   * *Sprint review* yaitu melaporkan tugas yang telah dikerjakan selama satu *sprint*.
   * *Sprint restrospective* yaitu kegiatan khusus dimana tim pengembang membahas evaluasi apa saja yang telah dikerjakan selama *sprint* sebelumnya.

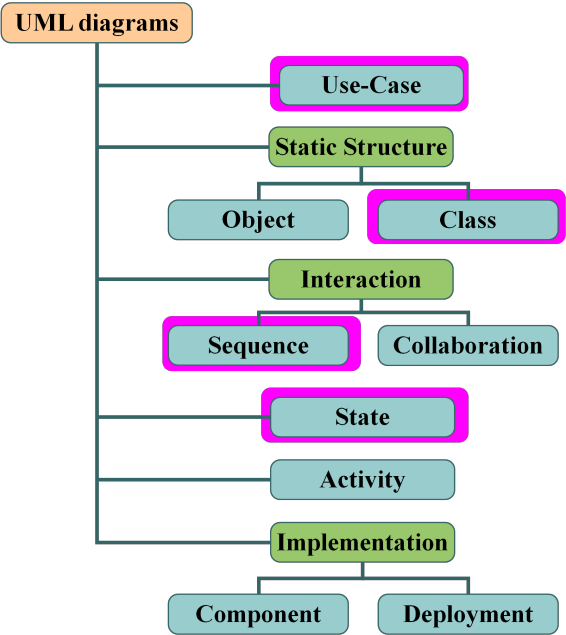


Gambar 3 : Alur Kerja Metode Scrum

Pada gambar 3 di atas merupakan alur kerja dari metode scrum, akan tetapi metode scrum di atas perlu dispesifikasikan untuk menentukan tugas apa saja yang harus dikerjakan dalam mengembangkan suatu aplikasi. Oleh karena itu dibutuhkan suatu pemodelan aplikasi salah satunya menggunakan *Unified Modelling Language*.

#### 2.1.2.2 Unified Modelling Language

Menurut (Haviluddin, 2011) *Unified Modelling Language* atau UML merupakan aplikasi untuk menggambarkan sebuah program ke dalam sebuah model yang berbentuk diagram. Tujuan UML ini supaya memudahkan developer dalam membuat sistem yang sudah dimodelkan dengan diagram secara visual. Berikut gambar UML diagram-diagram apa saja yang dapat membantu dalam sebuah sistem:



Gambar 4 : UML Diagram

Gambar 4 di atas menunjukkan banyaknya diagram dalam UML akan tetapi dalam teori ini hanya dibahas beberapa yang peneliti gunakan dalam penelitian ini diantaranya :

1. *Use Case* Diagram yaitu diagram yang menggambarkan aktor, fungsi, serta relasinya sebagai visualisasi yang memberikan peran untuk aktor. Dalam diagram ini terbagi menjadi dua bagian yaitu *System Use Case* merupakan interaksi dengan sistem dan *Business Use Case* merupakan interaksi bisnis dengan konsumen. Pada diagram ini juga disebut sebagai *behavior* diagramkarena menggambarkan bisnis proses dari sebuah sistem.
2. *Class* Diagram yaitu visualisasi struktur kelas dari suatu sistem, kelas yang dimaksud merupakan suatu bagian yang berupa tabel. Diagram ini juga memiliki peran dalam membentuk arsitektur sistem yang akan dibuat. Terdapat tiga bagian dalam *class* diagramyaitu nama, atribut, dan metoda. Pada diagram ini juga disebut sebagai strukturdiagramkarena menggambarkan spesifikasi arsitektur suatu sistem mulai dari kelas, objek, dan *relationship*.
3. *Sequence* Diagram yaitu objek yang berinteraksi secara tersusun berdasarkan urutan kejadian atau dapat dikatakan sebagai langkah demi langkah suatu sistem. Pada diagram ini disebut juga sebagai *interaction* diagram karena menggambarkan interaksi objek.
4. *Activity* Diagramyaitu suatu alur kerja yang menggambarkan akifitas apa saja yang ada dalam sistem. Diagram ini juga sama seperti *use case* diagram dalam hal sifat yaitu sebagai *behavior* diagram.

### 2.1.3 *Tools* Pengembangan

Pada *tools* pengembangan yang akan digunakan di dalam penelitian ini yaitu php *framework* laravel dan postgresql.

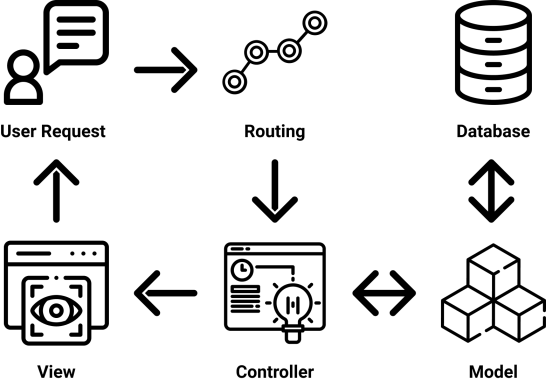
#### 2.1.3.1 PHP Framework Laravel

PHP atau *Hypertext Preprocessor* merupakan bahasa pemrograman untuk pengembangan situs web dan dapat digabungkan dengan *script* HTML atau *Hypertext Markup Language*, serta berfungsi dalam mengolah data dan mengirimkannya kembali ke situs web browser (Firman dkk., 2016). Sedangkan, *framework* merupakan kumpulan intruksi yang disatukan ke dalam suatu *class* dan *function* supaya memudahkan pengembang situs web dalam memanggil intruksi tersebut, tanpa harus menuliskan kembali *syntax program* yang sama berulang kali sehingga dapat menghemat waktu (Destiningrum & Adrian, 2017). Dengan kata lain, PHP *framework* merupakan kumpulan intruksi dalam pengembangan situs web dengan menggunakan bahasa pemrograman PHP agar sebuah data dikelola disisi server.

Adapun PHP *framework* laravel atau yang biasa disebut laravel saja menurut (Handika & Purbasari, 2019) merupakan kerangka kerja dalam pengembangan situs web yang memiliki arsitektur dengan menggabungkan banyak fitur terbaik dalam membangun situs web sehingga akan meningkatkan kecepatan situs web tersebut. Laravel selain dibuat dari gabungan fitur terbaik dalam pengembangan situs web, akan tetapi memiliki kelebihan tersendiri yaitu kinerja yang lebih cepat, *reload data* lebih stabil, memiliki keamanan data yang *secure*, menggunakan fitur khusus yang sering disebut blade, terdapat *library* yang mendukung pengembangan situs web, dan fitur pengelolaan *database.*

Kelebihan-kelebihan di atas tidak luput dari adanya arsitektur pada laravel yang terdiri dari *routes, controller, model, view,* dan *migrations.* Masing-masing arsitektur tersebut memiliki fungsi tersendiri berikut penjelasannya :

1. *Routes* berfungsi untuk memberikan akses ke setiap *request* yang telah dideklarasikan dan memiliki 4 *function* utama yaitu :
   * *Get*, berfungsi memanggil *request*.
   * *Put*, berfungsi mengambil data dari *request*.
   * *Post*, berfungsi menambahkan data dari *request*.
   * *Delete*, berfungsi menghapus data dari *request*.
2. *Controller* berfungsi sebagai penghubung antara *model* dan *view* dan mempunyai *function* dalam memproses data yang akan ditampilkan dari *model* ke *view* ataupun dari *view* ke *model*,adapun *function* tersebut yaitu :
   * *Index*, berfungsi menampilkan keseluruhan data.
   * *Create*, berfungsi memanggil *form* untuk proses data baru.
   * *Store*, berfungsi menyimpan data ke dalam *database* berdasarkan *request*.
   * *Show*, berfungsi menampilkan data berdasarkan *key*.
   * *Edit*, berfungsi mengambil data berdasarkan *key* dan mengambil *form* untuk proses memperbaharui data.
   * *Update*, berfungsi mengubah data pada database berdasarkan *request*.
   * *Delete*, berfungsi menghapus data berdasarkan *key*.
3. *Model* berfungsi melakukan pengelolaan pada tabel di sebuah *database* yang telah dideklarsikan dan dalam mengelola tabel dapat menggunakan banyak *function,* akan tetapi hanya beberapa *function* saja yang sering digunakan yaitu :
   * *Table*, berfungsi untuk mendeklarasikan suatu tabel dari *database* yang akan digunakan pada suatu *model.*
   * *Fillable*, berfungsi untuk mendeklarasikan kolom mana saja dari tabel yang telah ditentukan agar dapat digunakan pada suatu *model*.
4. *View* berfungsi menampilkan suatu data ke dalam situs web dan pada laravel menggunakan format khusus yang dinamakan *blade* dengan kode yang berisikan HTML (*HyperText Markup Language*) dan PHP.
5. *Migrations* berfungsi sebagai cetak biru dari pembuatan suatu tabel yang kemudian dimasukkan ke dalam *database.*

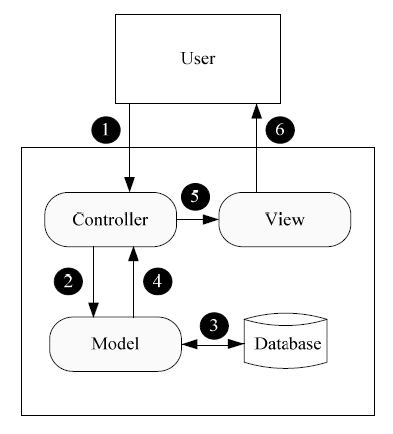


Gambar 5 : Arsitektur Laravel

Terlihat arsitektur laravel pada gambar 5 di atas dengan kelebihan-kelebihannya yang sebelumnya telah dijelaskan. Selain itu laravel juga menggunakan alur kerja MVC atau *Model View Controller* merupakan konsep dalam pengembangan web dengan menjadi tiga proses utama yaitu:

1. *Model* yaitu proses untuk mengaitkan dengan operasi basis data.
2. *View* yaitu proses sebagai penghubung dengan antarmuka web.
3. *Controller* yaitu proses untuk menghubungkan dengan logika web dan mengatur alur data antara *view* dan *model*.

Ketiga proses di atas adalah bentuk pengembangan dari konsep pemrograman berorientasi objek atau OOP (Suendri, 2018) dan berikut merupakan konsep dari proses MVC itu sendiri yang dapat dilihat pada gambar 6.

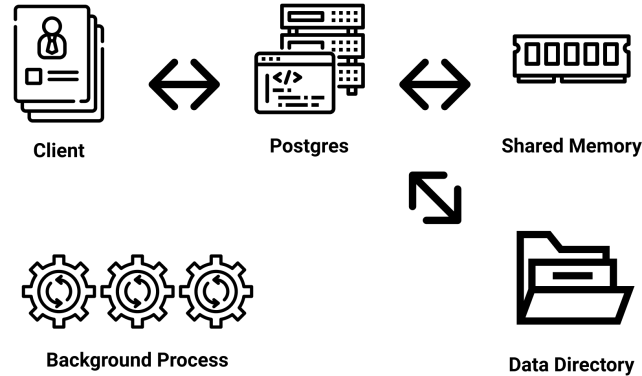


Gambar 6 : Konsep MVC

Berdasarkan penjelasan tersebut dapat peneliti tentukan PHP *framework* laravel sebagai pengembangan web untuk aplikasi Link-Match STT-NF modul dosen. Hal tersebut dikarenakan PHP framework laravel memliki arsitektur yang optimal, kinerja yang cepat, menerapkan konsep MVC, dan kelebihan lainnya yang telah dijelaskan sebelumnya.

#### 2.1.3.2 PostgreSQL

*Tools* pengembangan selanjutnya dalam penelitian ini yaitu PostgreSQL yang merupakan aplikasi dalam mengelola kumpulan basis data atau disebut *database.* PostgreSQL ini termasuk ke dalam *Object-Relational Database Management System* (ORDBMS) yang merupakan perbaikan dari *Database Management System* (DBMS) tradisional dalam sisi fungsional dan bersifat *open source*. Adapun bentuk arsitektur dari PostgreSQL dapat dilihat pada gambar 7 berikut.



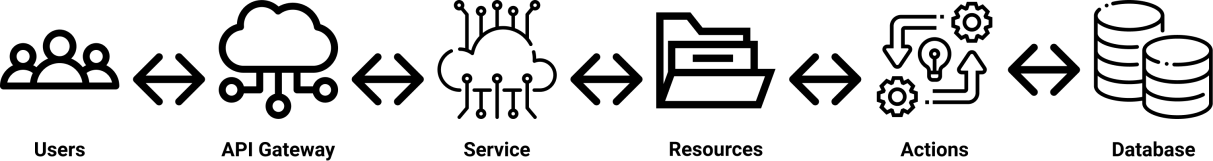
Gambar 7 : Arsitektur PostgreSQL

Dalam menggunakan PostgreSQL dapat dibantu dengan adanya interface yang telah disediakan salah satunya yaitu PgAdmin, berfungsi untuk mendesain dan manajemen basis data secara komprehensif. Selain PgAdmin terdapat banyak *plugin* dalam menggunakan PostgreSQL diantaranya PostGIS yang berfungsi sebagai *backend* *spatial database* dan PgRouting yang berfungsi sebagai penyedia fungsionalitas routing pada *database* PostgreSQL serta dapat menyelesaikan masalah pencarian *Shortest Path* (Aminullah dkk., 2018). Hal tersebut yang menjadikan PostgreSQL ini dipilih oleh peneliti karena sifatnya *open source* namun tetap *powerfull* mulai dari fungsionalnya hingga kinerjanya.

### 2.1.4 *Web Service*

Aplikasi Link-Match STT-NF berbasis web tidak terlepas dari *web service* supaya terintegrasi dengan aplikasi Link-Match STT-NF yang berbasis android. Menurut (Sibagariang, 2016), *web service* merupakan *software* yang bertujuan untuk mendukung interaksi antar platform melalui suatu jaringan dengan format tertentu agar dapat dibaca oleh suatu platform menggunakan protokol standar internet. Terdapat dua jenis *web service* yaitu:

1. *Simple Object Access Protokol* atau SOAP merupakan protokol dalam saling bertukar pesan dengan format *Extensible Markup Language* (XML) antara komputer disuatu jaringan yang menggunakan *Hypertext Transfer Protocol* (HTTP) atau *Hypertext Transfer Protocol Secure* (HTTPS).
2. *Representational State Transfer* atau disebut REST merupakan konsep perpindahan antar *state* ketika adanya *request* suatu halaman web kemudian server web tersebut akan mengirimkan kembali *state* yang diberikan, melalui *link* HTTP untuk melakukan aktivitas tertentu sehingga terjadi perpindahan *state* antara satu sama lain. Adapun fungsi HTTP yang dapat digunakan yaitu GET, POST, PUT atau DELETE. *State* yang dikirimkan dalam bentuk XML tanpa adanya *protocol data package* sehingga informasi yang diterima lebih mudah dibaca dan diparsing pada *client side*. Implementasi REST sering digunakan di *web service* yang berorientasi pada *resource*, sehingga menyediakan *resource as* *service* akan tetapi bukanlah kumpulan dari suatu aktifitas yang mengolah *resource* itu.



Gambar 8 : Arsitektur *Web Service*

Gambar di atas merupakan bentuk arsitektur dari *web service* dan akan digunakan dalam penelitian ini dengan jenis REST akan tetapi dengan suatu *micro framework* yaitu Lumen. Lumen merupakan *framework* yang dikhususkan untuk pembuatan *Application Programming Interface* atau API dan memiliki keunggulan dalam performa yang lebih baik dalam menangani *request* serta dapat menangani 1900 *request* perdetik dibandingkan API lainnya (Surahman dkk., 2018).

### 2.1.5 Pengujian Sistem

Aplikasi Link-Match STT-NF yang dikembangkan melalui proses pengujian sistem diantaranya *Black Box Testing*, *User Acceptance Testing*, Postman, Kuesioner, dan Skala Likert.

#### 2.1.5.1 Black Box Testing

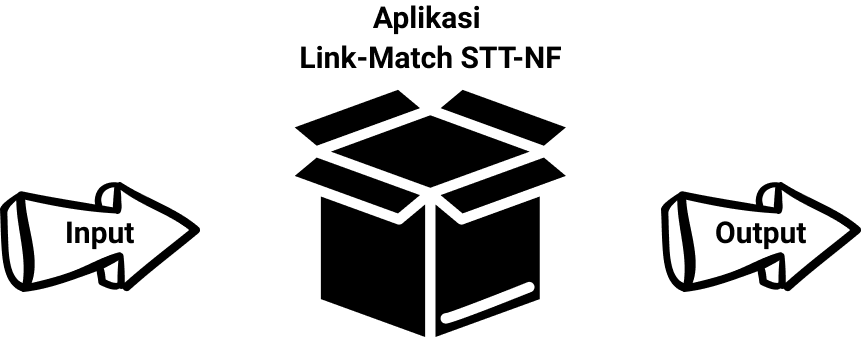
Menurut (Mustaqbal dkk., 2015), *Black Box Testing* merupakan pengujian terhadap spesifikasi fungsional dari suatu sistem dan sebagai pelengkap dalam menguji beberapa hal diantaranya:

1. *Interface errors*
2. *Performance errors*
3. *Incorrect or missing function*
4. *Initialization and termination errors*
5. *Errors in data structures and database access*

Selain pelengkap di atas dalam melaksanakan *black box testing* terdapat teknik yang dapat dilakukan diantaranya:

1. *Sample Testing*
2. *Behavior Testing*
3. *Endurance Testing*
4. *Robustness Testing*
5. *Comparison Testing*
6. *Requirement Testing*
7. *Performance Testing*
8. *Equivalence Partitioning*
9. *Boundary Value Analysis / Limit Testing*
10. *Cause-Effect Relationship Testing*

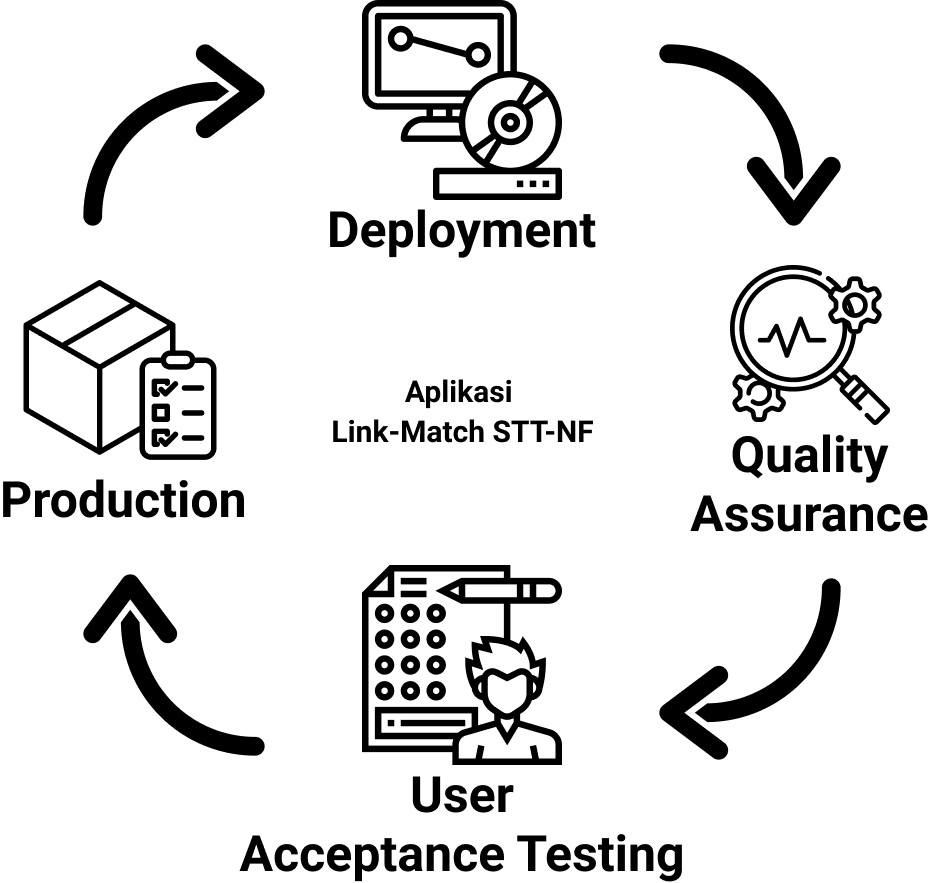
Berikut adalah ilustrasi bagaimana cara kerja dari *black box testing* dalam aplikasi Link-Match STT-NF:



Gambar 9 : Alur Kerja *Black Box Testing* Link-Match STT-NF

#### 2.1.5.2 User Acceptance Testing

Metode pengujian selanjutnya yaitu *User Acceptance Testing* atau UAT merupakan pengujian yang dikhususkan dengan melibatkan pengguna atau *user* namun tidak secara keseluruhan sistem akan tetapi hanya sistem yang berkaitan dengan pengguna saja. UAT bertujuan dalam mengetahui seberapa layak suatu sistem yang akan dikeluarkan dan mengetahui manfaat apa saja yang dapat diperoleh dari sudut pandang *user* sehingga suatu sistem dapat disusun berdasarkan penilaian *user* (Utomo dkk., 2018). Seperti yang diilustrasikan pada gambar berikut:



Gambar 10 : Alur Kerja *User Acceptance Testing* Link-Match STT-NF

**2.1.5.3 Postman**

Pengujian sistem selanjutnya yaitu Postman yang merupakan aplikasi dalam menguji REST *web service* sehingga dapat memudahkan dalam mengembangkan API dengan cepat dalam bentuk *request* HTTP (Choiri dkk., 2017). Selain pengujian sistem menggunakan *black box testing,* *user acceptance testing,* dan Postman, ditambahkan suatu metode pengumpulan data berupa kuesioner.

#### 2.1.5.4 Kuesioner

Menurut (Purnomo & Palupi, 2016), kuesioner merupakan metode pengumpulan data dengan cara memberikan pertanyaan kepada responden untuk dimintai tanggapan sesuai pertanyaan tersebut. Kuesioner pada penelitian ini ditujukan agar mengetahui kebutuhan aplikasi Link-Match STT-NF dari *user* tersebut yaitu mahasiswa dan dosen, adapun jenis kuesioner yang digunakan ada dua macam yaitu:

1. Wawancara, sebagai analisis data kualitatif yang dilakukan kepada mahasiswa maupun dosen agar mengetahui kebutuhan apa saja pada aplikasi Link-Match STT-NF serta tanggapan terkait aplikasi tersebut.
2. Angket, sebagai analisis data kuantitatif hanya kepada mahasiswa supaya dapat mengukur data yang dibutuhkan dalam pengembangan aplikasi Link-Match STT-NF dari segi fungsionalitas. Metode yang digunakan dalam menganalisis data kuantitatif ini menggunakan skala likert. Skala likert merupakan skala pengukuran kumpulan data yang memiliki empat atau lebih pilihan yang disesuaikan sehingga terbentuk sebuah skor yang merepresentasikan data dari suatu pertanyaan (Maryuliana dkk., 2016). Adapun pilihan dengan skornya yang sering digunakan seperti berikut:

|  |  |  |
| --- | --- | --- |
| **No** | **Pilihan** | **Skor** |
| 1 | Sangat Setuju (SS) | 4 |
| 2 | Setuju (S) | 3 |
| 3 | Tidak Setuju (TS) | 2 |
| 4 | Sangat Tidak Setuju (STS) | 1 |

Tabel 1 : Skala Likert

## 2.2 Penelitian Terkait

Penelitian ini tidaklah secara keseluruhan hal yang baru, melainkan sudah ada penelitian sebelumnya terkait dengan penelitian ini, berikut diantaranya:

### 2.2.1 Tabel Penelitian Terkait

|  |  |  |
| --- | --- | --- |
| **Judul Penelitian** | **Peneliti** | **Kesimpulan** |
| Pembangunan Aplikasi Monitoring Budget *Event Organizer* pada PT Indi Notokreasi Berbasis Web Menggunakan PHP Framework Laravel | (Tarmizi, 2018)  (Sekolah Tinggi Teknologi Terpadu Nurul Fikri) | Aplikasi monitoring budget event organizer berbasis web menggunakan php *framework* laravel dapat berfungsi dengan baik. Hal tersebut dibuktikan dari hasil pengujian fungsional *black box testing* yang telah berjalan 100%, hasil pengujian UAT dinyatakan 83% fitur aplikasi berjalan, dan 17% fitur diterima dengan catatan. Serta memberi kemudahan untuk memantau keuangan di PT INDI Notokreasi, dibuktikan dengan akses informasi menjadi lebih cepat dari sebelumnya. |
| Rancang Bangun Aplikasi Web untuk Manajemen Proyek Berbasis Scrum | (Alamsyah & Yuliansyah, 2016)  (Universitas Ahmad Dahlan) | Aplikasi web manajemen proyek berbasis scrum yang memiliki fitur *burndown chart* sebagai alat bantu untuk merefleksikan progress dari proyek perangkat lunak dan fitur untuk membuat prioritas dan penjadwalan. Berdasarkan hasil pengujian pada aplikasi tersebut dapat diketahui bahwa 100% *testcase* yang diuji pada aplikasi, telah sesuai dengan *Exfected Result*. Dapat disimpulkan bahwa aplikasi tersebut dapat berjalan dengan baik dan berfungsi dalam mengelola manajemen proyek berbasis scrum. |
| Aplikasi Manajemen Multi Proyek Menggunakan Metode Scrum | (Kusuma dkk., 2019)  (Universitas Brawijaya) | Aplikasi manajemen multi proyek menggunakan metode scrum ini berbasis web dengan php framework laravel. Berawal dari proses analisis kebutuhan menggunakan skenario *use case* dilanjutkan dengan perancangan aplikasi berupa pemodelan diagram hingga implementasi. Proses terakhir yaitu pengujian aplikasi menggunakan *white box testing* dan *black box testing*. Hasil yang didapat dapat dikatakan baik setelah dioperasikan pada versi 9 dan 11 browser. |

Tabel 2 : Penelitian Terkait

### 2.2.2 Posisi Penelitian

|  |  |  |  |
| --- | --- | --- | --- |
| **PHP Framework Laravel** | **Manajemen Proyek** | **Scrum** | **PostgreSQL** |
| Muhamad Tarmizi, 2018 (Sekolah Tinggi Teknologi Terpadu Nurul Fikri) Pembangunan Aplikasi Monitoring Budget Event Organizer pada PT Indi Notokreasi Berbasis Web Menggunakan PHP Framework Laravel |  |  |  |
|  | Adi Alamsyah & Herman Yuliansyah, 2016 (Universitas Ahmad Dahlan) Rancang Bangun Aplikasi Web untuk Manajemen Proyek Berbasis Scrum | |  |
| Andri Wijaya Kusuma, Nurudin Santoso, & Arief Andy Soebroto, 2019 (Universitas Brawijaya) Aplikasi Manajemen Multi Proyek menggunakan Metode Scrum | | |  |
| Muhammad Azhar Rasyad, 2020 (Sekolah Tinggi Teknologi Terpadu Nurul Fikri) Rancang Bangun Aplikasi Link-Match STT-NF Modul Dosen Berbasis Web Menggunakan Framework Laravel | | | |

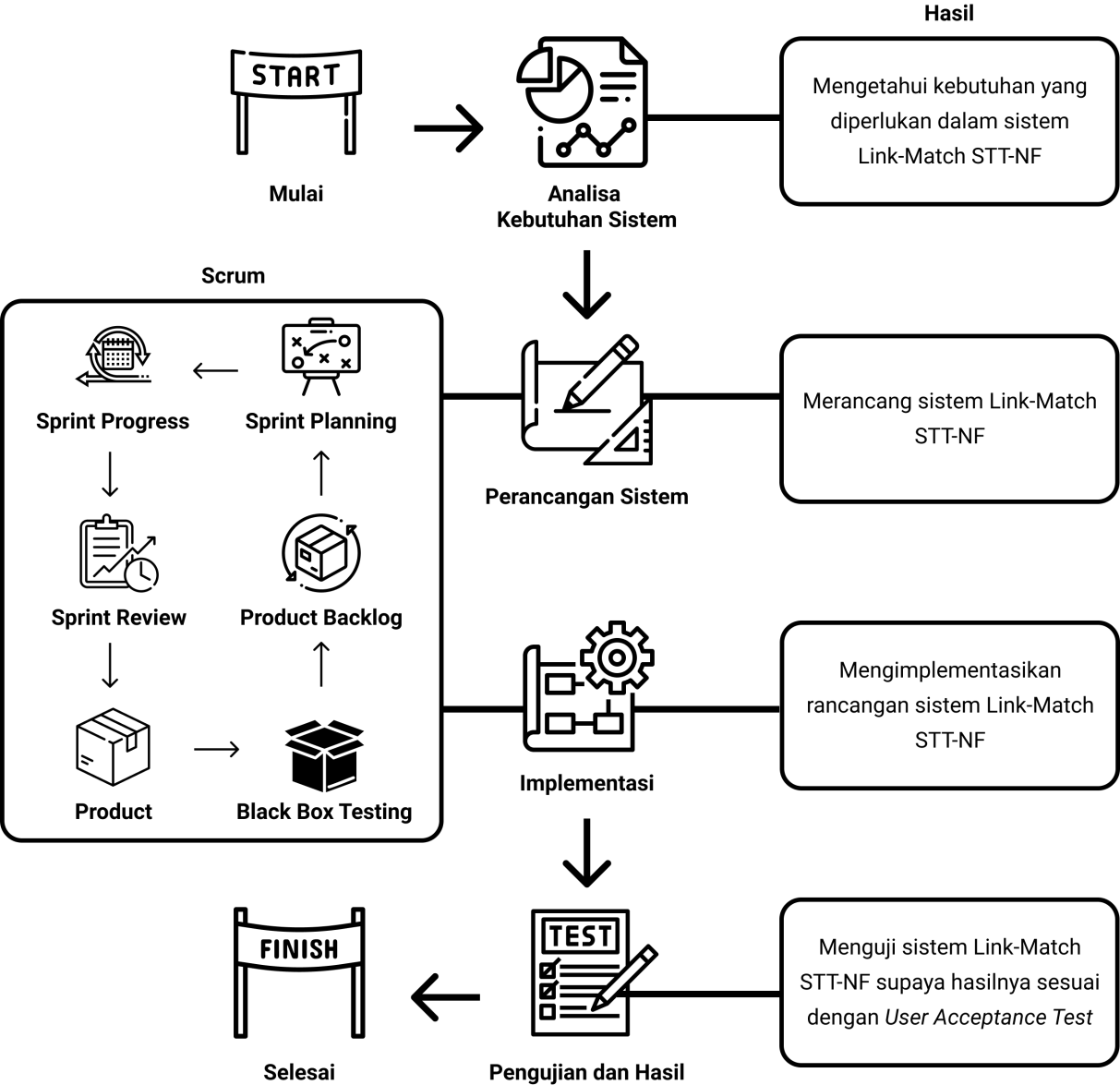
Tabel 3 : Posisi Penelitian

# BAB III

# METODE PENELITIAN

## 3.1 Tahapan Penelitian

Penelitian ini memiliki tahapan yang secara sistematis harus dilakukan selangkah demi selangkah diantaranya yaitu analisa kebutuhan sistem, perancangan sistem, implementasi, serta pengujian dan hasil.



Gambar 11 : Tahapan Penelitian

### 3.1.1 Analisa Kebutuhan Sistem

Tahap awal dari penelitian ini yaitu menganalisa kebutuhan apa saja yang dibutuhkan dalam sistem Link-Match STT-NF. Analisa yang akan digunakan dengan mengumpulkan data terlebih dahulu melalui pengguna sistem tersebut yaitu mahasiswa dan dosen dengan menggunakan kuesioner yang berupa angket dan wawancara. Setelah didapatkan data yang dibutuhkan dari kuesioner tersebut kemudian data tersebut diolah supaya dapat dianalisis kebutuhan apa saja yang dibutuhkan dalam sistem Link-Match STT-NF.

### 3.1.2 Perancangan Sistem

Tahapan setelah mengetahui kebutuhan apa saja dalam sistem tersebut yaitu merancang sistem itu dengan menggunakan analisa yang telah didapat. Perancangan sistem diawali dengan membuat UML diagram Link-Match STT-NF yang dapat berupa *use case diagram, class diagram, sequence diagram,* dan *activity diagram*. Setelah dirancang UML diagram maka dilanjutkan dengan merancang prototipe sehingga sistem tersebut dapat dengan mudah diimplementasikan.

### 3.1.3 Implementasi

Tahapan berikutnya yaitu implementasi, setelah sistem dirancang sedemikian rupa melalui prototipe maka dapat dilanjutkan dengan mengimplementasikannya ke dalam kode program hingga sistem Link-Match STT-NF bekerja dengan baik. Adapun jika terjadi perubahan pada rancangan sistem sebelumnya dapat dengan mudah ditangani karena metode pengembangan yang digunakan yaitu Scrum, sehingga perancangan sistem dan implementasinya bersifat fleksibel.

### 3.1.4 Pengujian dan Hasil

Tahap akhir yaitu pengujian sistem Link-Match STT-NF dengan menggunakan *black box testing* dan *user acceptance testing*. Pengujian dilakukan supaya ketika terdapat kesalahan kode program atau kesalahan fungsi dapat diketahui, mengingat yang akan menggunakan sistem ini yaitu mahasiswa dan dosen sehingga perlu adanya penyesuaian dari pengembang ke pengguna melalui tahapan pengujian.

## 3.2 Rancangan Penelitian

Sebelum penelitian ini dilakukan maka ada rancangan yang telah dipersiapkan sebelumnya diantaranya yaitu jenis penelitian, lingkungan pengembangan, serta bahan dan alat.

### 3.2.1 Jenis Penelitian

Pada penelitian ini termasuk ke dalam jenis penelitian kualitatif dan kuantitatif. Termasuk penelitian kualitatif karena penelitian ini bertujuan menggambarkan dan menjelaskan sistem Link-Match STT-NF, sedangkan termasuk penelitian kuantitatif juga karena dalam menjelaskan sistem tersebut membutuhkan data yang sebelumnya telah dikumpulkan. Sehingga metode yang digunakan pada penelitian ini melalui wawancara untuk digunakan sebagai analisis data kualitatif dan angket untuk digunakan sebagai analisis data kuantitatif.

### 3.2.2 Lingkungan Pengembangan

Pengembangan penelitian ini dilakukan dalam lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri pada dua tempat yaitu:

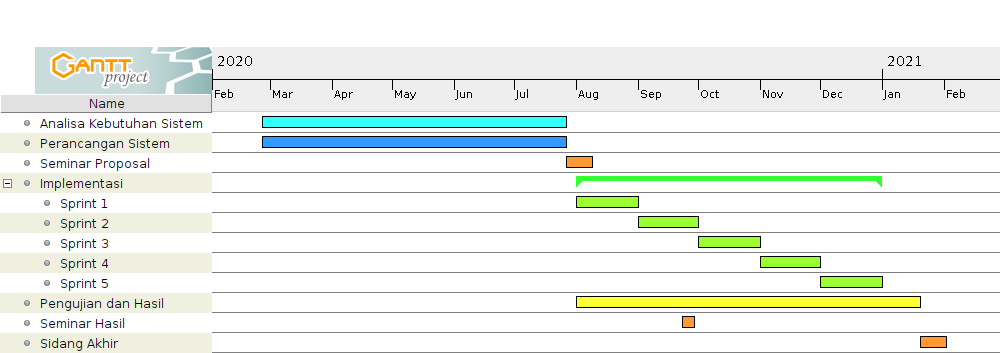
1. Kampus A STT Terpadu Nurul Fikri di alamat Jl. Setu Indah No.116, Tugu, Kec. Cimanggis, Kota Depok, Jawa Barat 16451
2. Kampus B STT Terpadu Nurul Fikri di alamat Jalan Lenteng Agung Raya No.20 RT.5/RW.1 Lenteng Agung, Kelurahan, RT.4/RW.1, Srengseng Sawah, Kec. Jagakarsa, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta 12640

### 3.2.3 Bahan dan Alat

Dalam mendukung penelitian ini diperlukan bahan serta alat diantaranya sebagai berikut:

1. Laptop Acer Aspire V5 dengan spesifikasi:
   * *Memory* : 4,0 GB
   * *Processor* : AMD® A6-1450 apu with radeon(tm) hd graphics × 4
   * Graphics : AMD® Kabini
   * *Disk Capacity* : 120,0 GB
2. Sistem Operasi Ubuntu 20.04 LTS 64 bit
3. StarUML
4. diagrams.net
5. Whimsical
6. Figma
7. Visual Studio Code
8. Mozilla Firefox
9. PHP *Framework* Laravel
10. Lumen
11. PostgreSQL

## 3.3 Jadwal Penelitian

Gambar 12 : Jadwal Penelitian

# BAB IV

# ANALISIS DAN PERANCANGAN

## 4.1 Analisis Sistem

Pada tahapan analisis sistem akan dilakukan beberapa analisa yaitu *end user* dan *user requirement*, hal tersebut dilakukan untuk menentukan siapa pengguna dan apa saja yang dibutuhkan dalam aplikasi Link-Match STT-NF modul dosen berbasis web.

### 4.1.1 *End User*

Setelah dilakukan analisis terhadap aplikasi Link-Match STT-NF modul dosen berbasis web dari proses wawancara, maka didapatkan *end user* diantaranya:

1. Dosen berperan dalam mengawasi proyek yang dikerjakan mahasiswa.
2. *Scrum master* berperan sebagai pembimbing bagi para mahasiswa dalam mengerjakan proyek yang diberikan dosen.
3. *Product owner* berperan dalam memberikan proyek yang harus dikerjakan oleh mahasiswa.
4. *Administrator* berperan sebagai pengelola administrasi pada aplikasi Link-Match STT-NF.
5. Mahasiswa berperan sebagai penerima informasi yang diberikan.

### 4.1.2 Kebutuhan Sistem

Pada aplikasi Link-Match STT-NF modul dosen berbasis web didapatkan kebutuhan sistem yang dibutuhkan setelah dilakukan proses analisis yaitu:

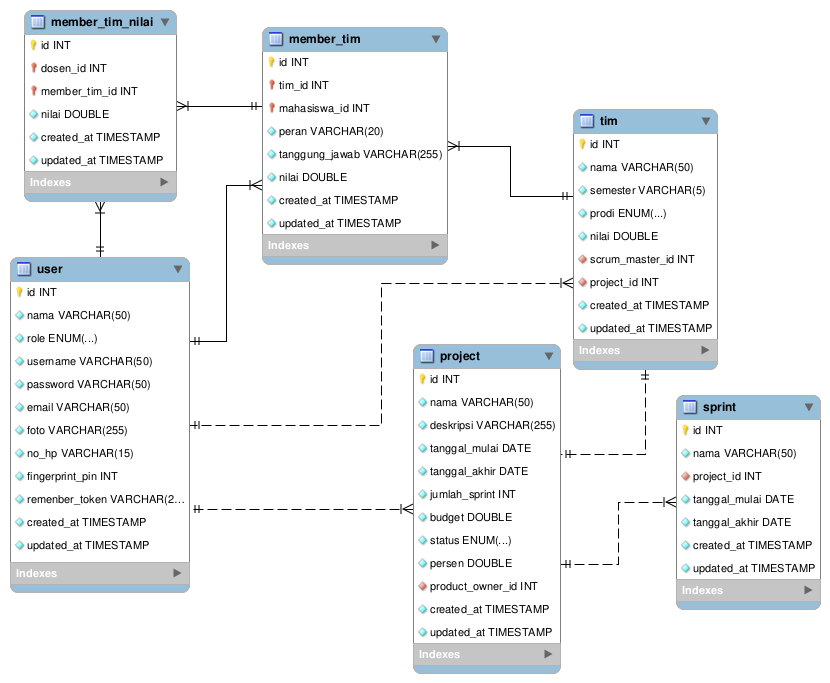
1. *User management* merupakan kebutuhan dalam memanajemen *user* mulai dari pembuatan *user*, memilih *user* ke dalam suatu tim, memantau progress *user* dalam mengerjakan proyek, hingga memberikan penilaian terhadap *user*.
2. *Team management* merupakan kebutuhan untuk memanajemen tim dalam menentukan *user* mana saja yang harus berada dalam tim tertentu, sehingga dapat optimal dalam mengerjakan proyek dan juga dilakukan penilaian terhadap tim tersebut.
3. *Project management* merupakan kebutuhan memanajemen proyek yang telah diberikan oleh *product owner* kepada setiap tim, mulai dari memberikan spesifikasi apa yang harus dikerjakan, memantau progress yang dikerjakan, hingga mendapatkan hasil dari proyek tersebut.

## 4.2 Perancangan Sistem

Tahapan setelah analisis yaitu merancang sistem pada aplikasi Link-Match STT-NF modul dosen berbasis web dalam bagian *entity relationship diagram, use case diagram, activity diagram,* dan *user interface*.

### 4.2.1 *Entity Relationship Diagram*

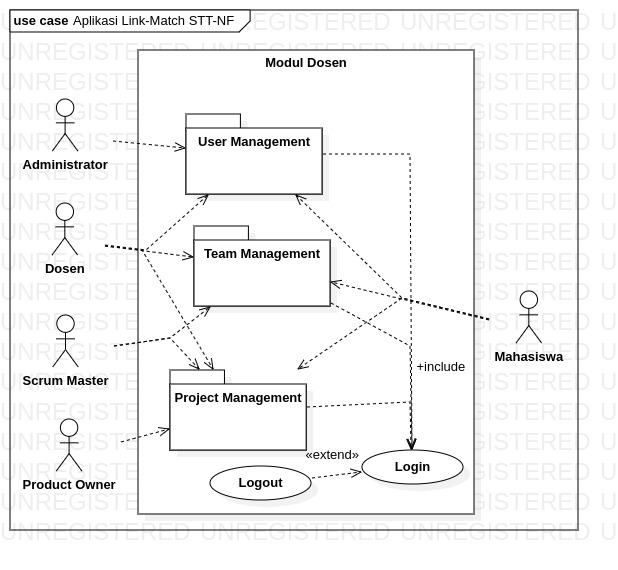
Gambar berikut adalah *entity relationship diagram* yang bertujuan menggambarkan relasi antara setiap entitas:



Gambar 13 : *Entity Relationship Diagram*

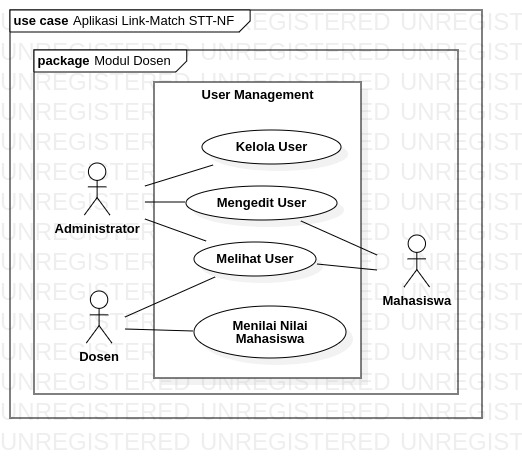
### 4.2.2 *Use Case Diagram*

Perancangan sistem setelah *entity relationship diagram* yaitu *use case diagram* yang bertujuan menggambarkan *end user* dapat melakukan apa saja pada aplikasi Link-Match STT-NF modul dosen, pada gambar berikut:



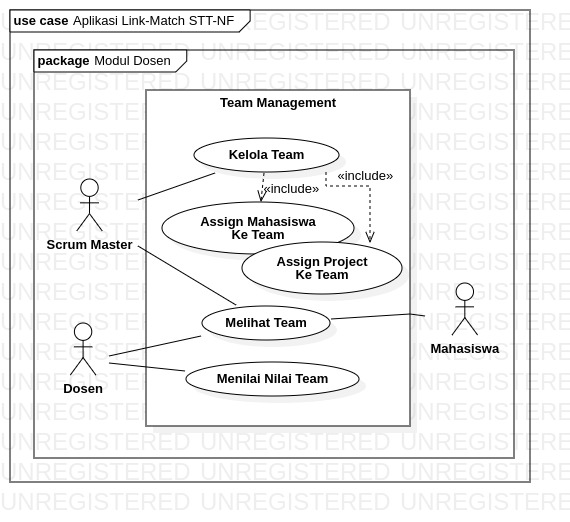
Gambar 14 : *Use Case Diagram*

Gambar berikut merupakan *use case* dari fitur *user management* yang melibatkan *end user* administrator, dosen, dan mahasiswa:



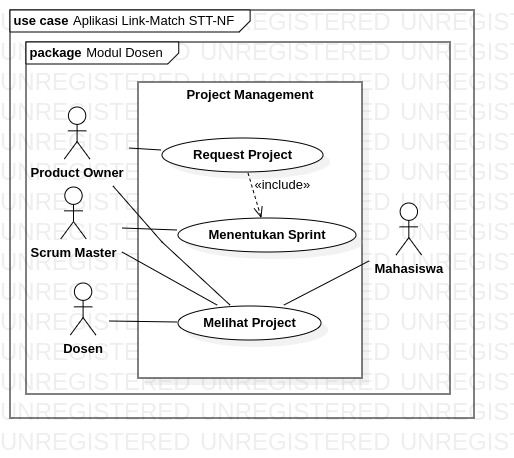
Gambar 15 : *Use Case Diagram User Management*

Gambar berikut merupakan *use case* dari fitur *team management* yang melibatkan *end user scrum master*, dosen, dan mahasiswa:



Gambar 16 : *Use Case Diagram Team Management*

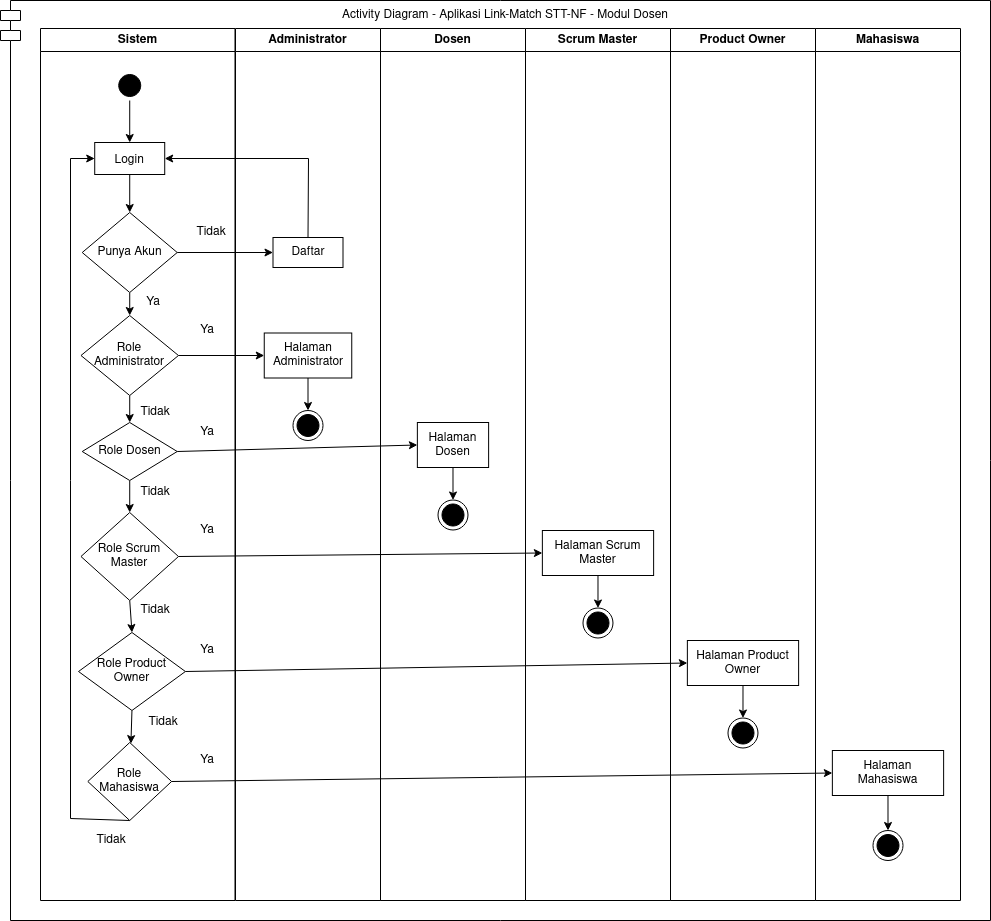
Gambar berikut merupakan *use case* dari fitur *project management* yang melibatkan *end user product owner, scrum master*, dosen, dan mahasiswa:



Gambar 17 : *Use Case Diagram Project Management*

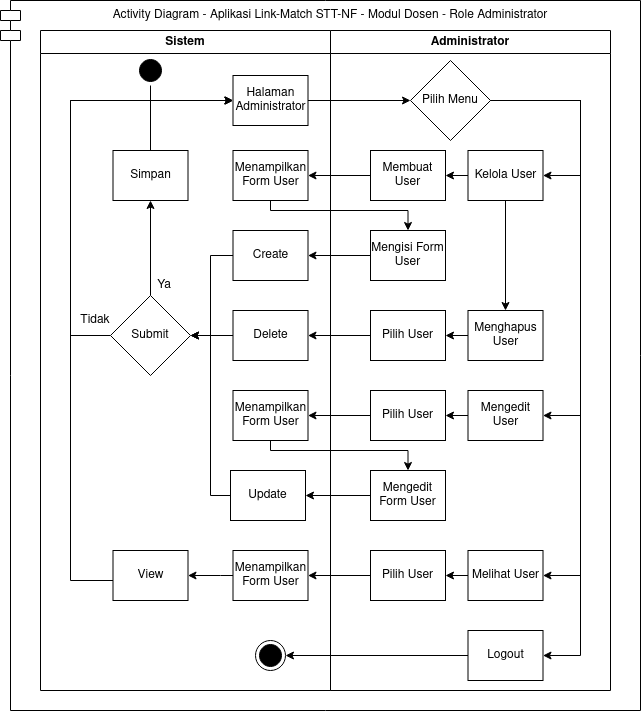
### 4.2.3 *Activity Diagram*

Penggambaran sistem dari *use case diagram* sebelumnya dilanjutkan dengan *activity diagram* yangbertujuan mengetahui aktifitas apa saja yang terjadi pada *use case* di atas, seperti yang terlihat pada gambar berikut:



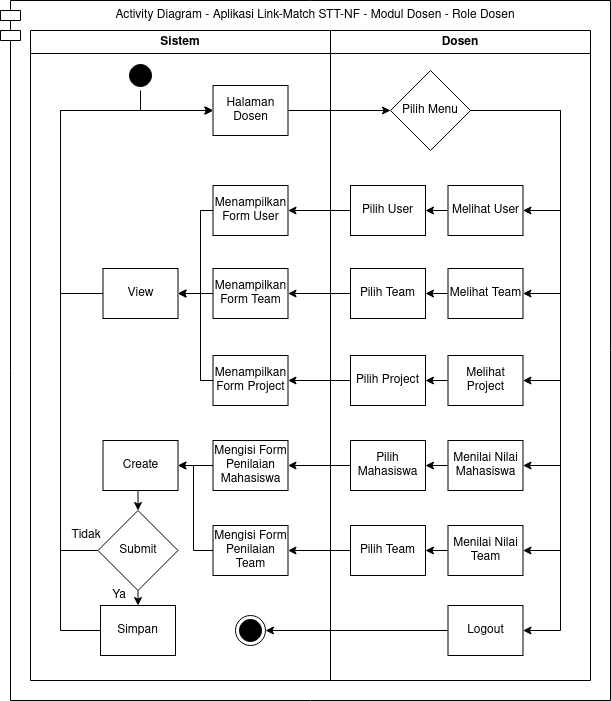
Gambar 18 : *Activity Diagram*

Gambar berikut merupakan *activity diagram* dari *role* administrator:



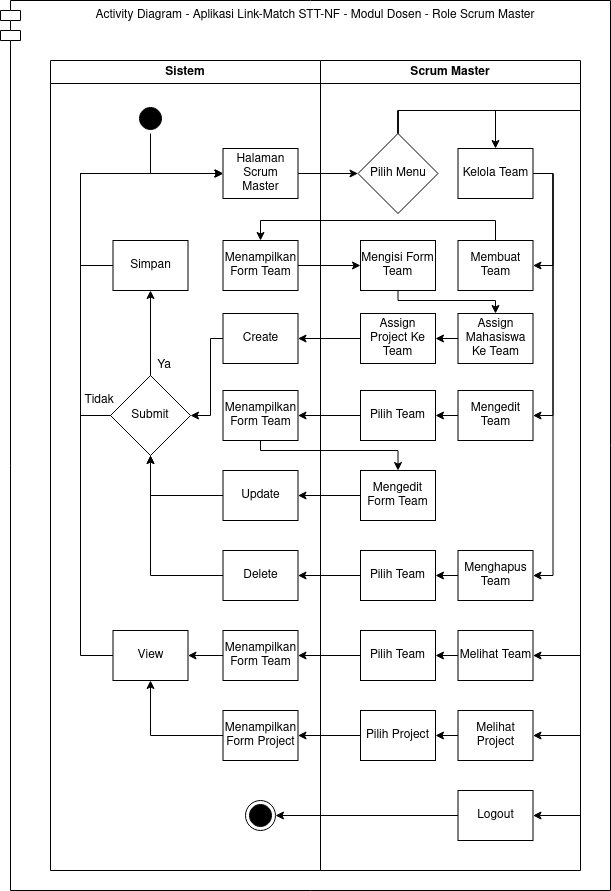
Gambar 19 : *Activity Diagram Role* Administrator

Gambar berikut merupakan *activity diagram* dari *role* dosen:



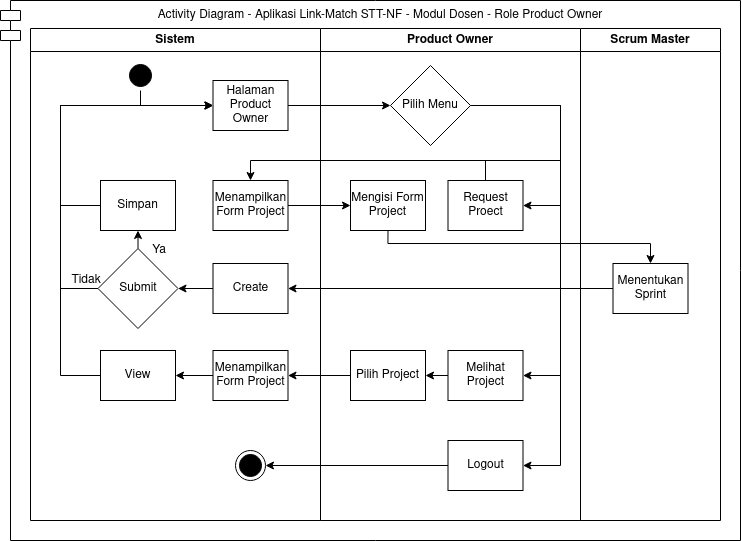
Gambar 20 : *Activity Diagram Role* Dosen

Gambar berikut merupakan *activity diagram* dari *role* *scrum master*:



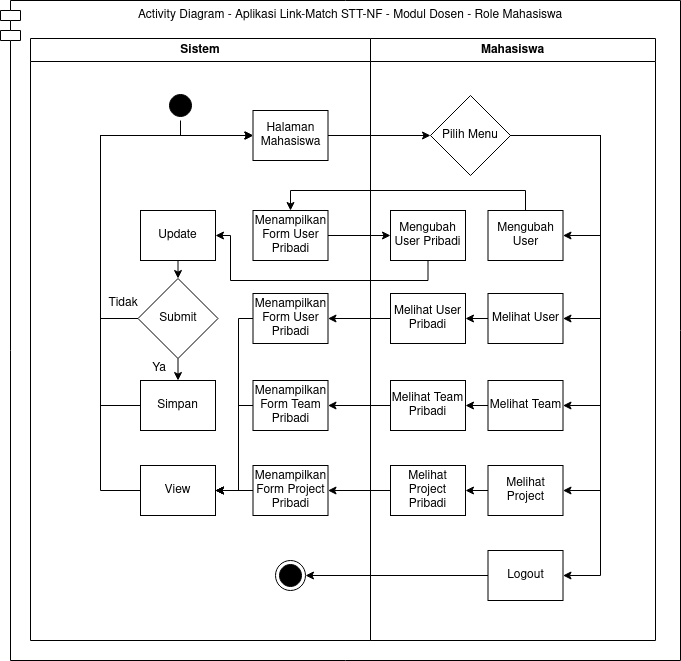
Gambar 21 : *Activity Diagram Role Scrum Master*

Gambar berikut merupakan *activity diagram* dari *role* *product owner*:



Gambar 22 : *Activity Diagram Role Product Owner*

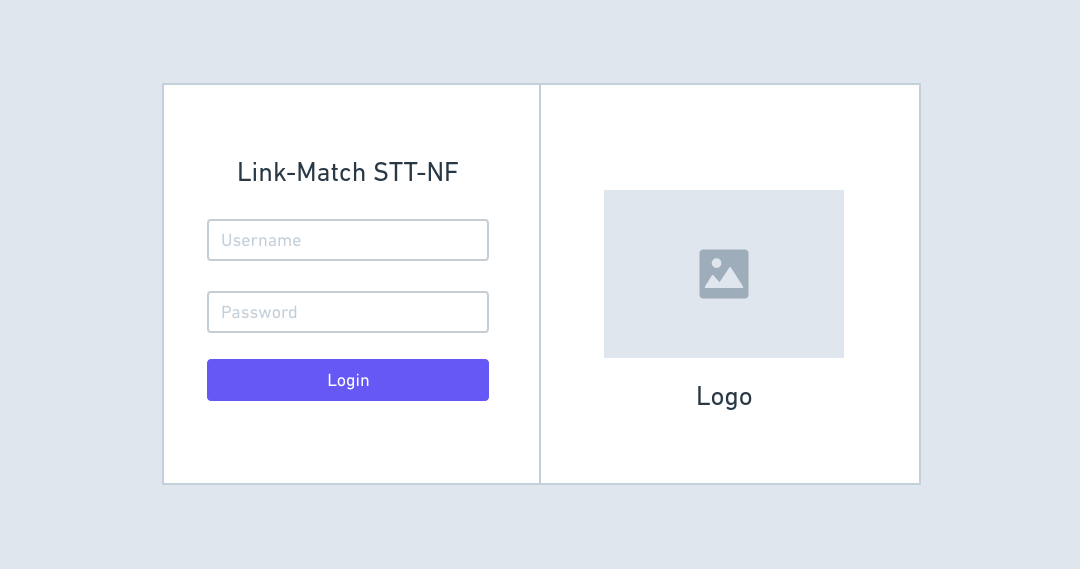
Gambar berikut merupakan *activity diagram* dari *role* mahasiswa:



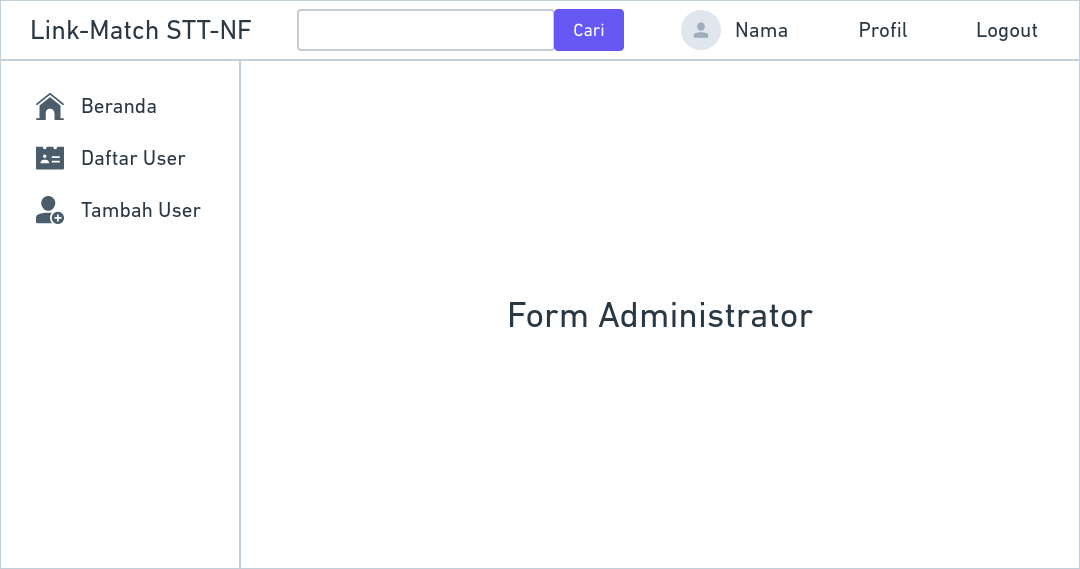
Gambar 23 : *Activity Diagram Role* Mahasiswa

### 4.2.4 *User Interface*

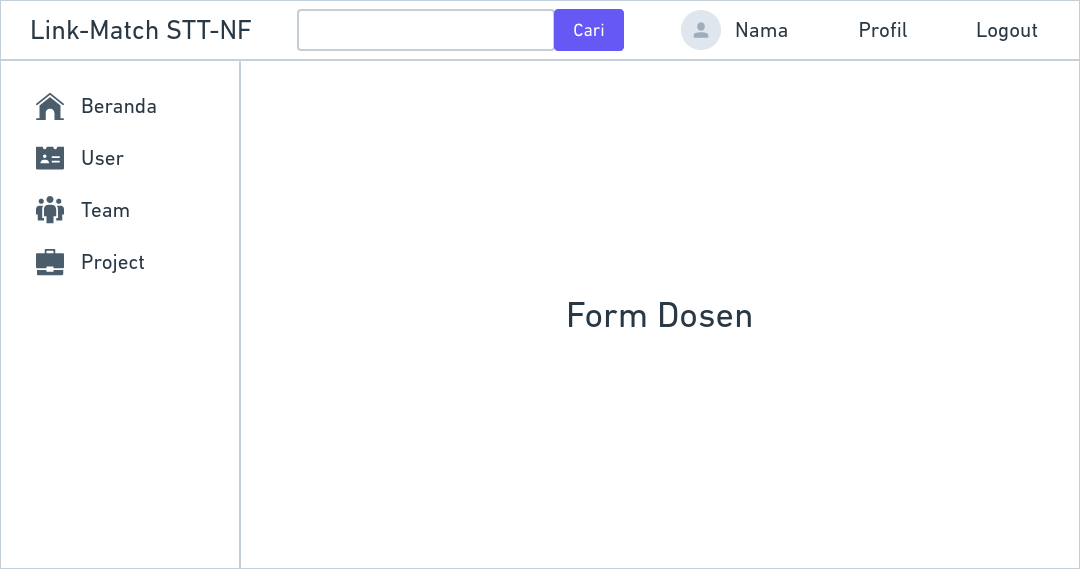
Perancangan sistem yang terakhir yaitu menyusun *user interface* dengan tujuan mempermudah konsep dari aplikasi Link-Match STT-NF modul dosen berbasis web dengan mudah. Berikut gambar-gambar yang merupakan *user interface* dari penelitian ini:



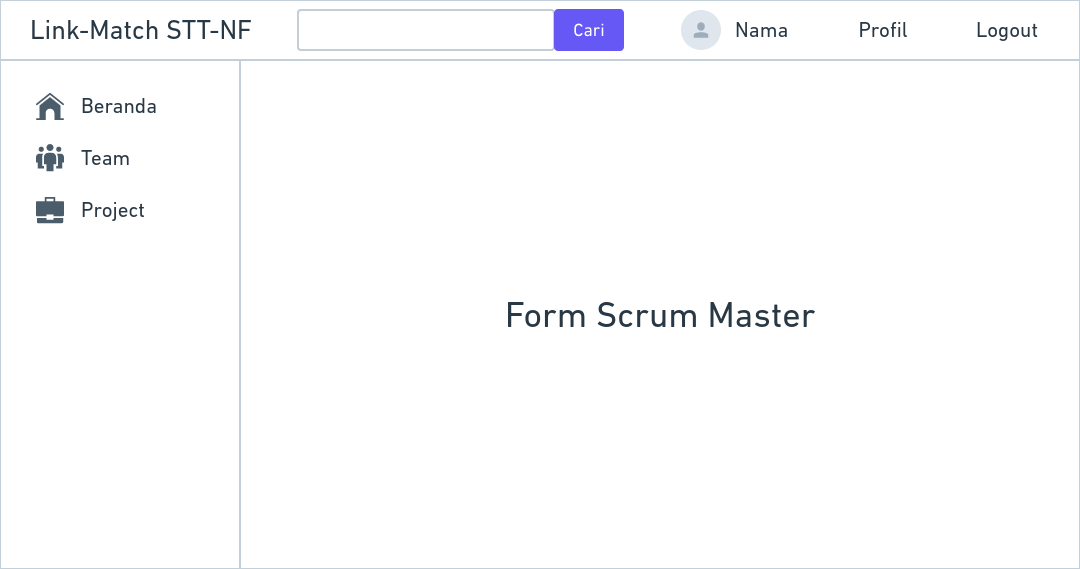
Gambar 24 : *User Interface Login*



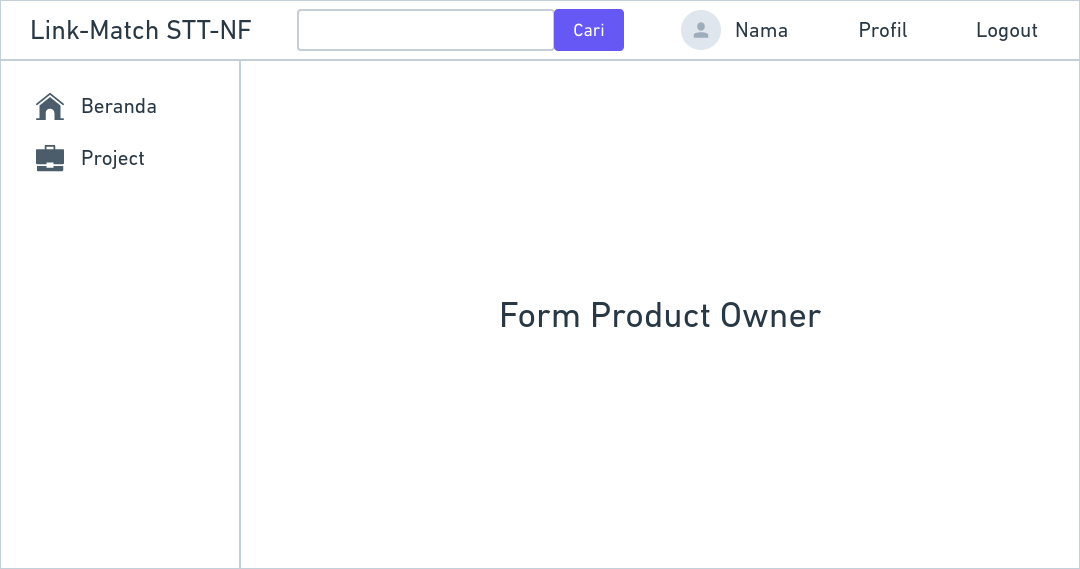
Gambar 25 : *User Interface Role* Administrator



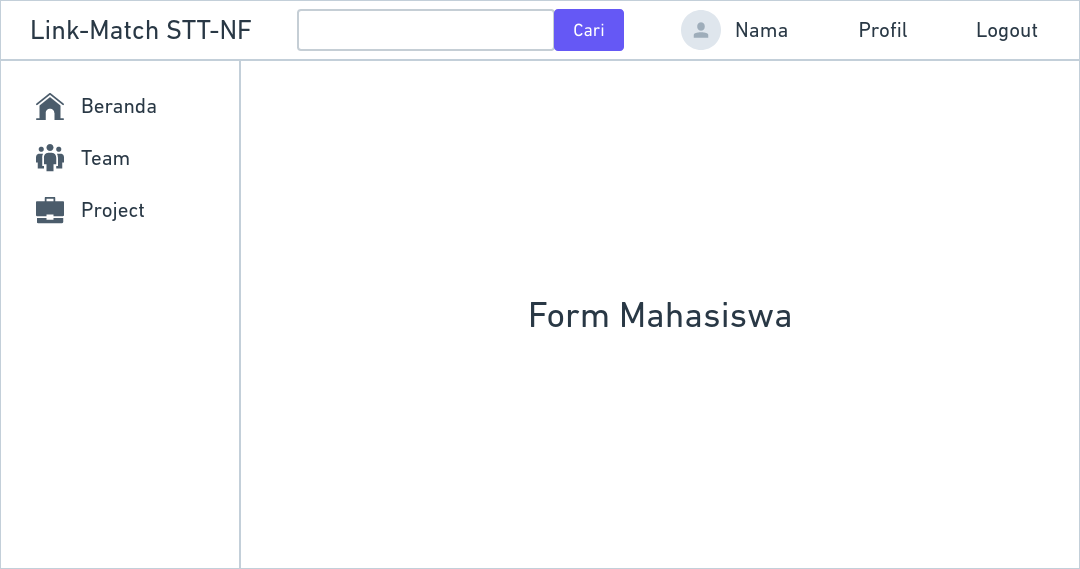
Gambar 26 : *User Interface Role* Dosen



Gambar 27 : *User Interface Role Scrum Master*



Gambar 28 : *User Interface Role Product Owner*



Gambar 29 : *User Interface Role* Mahasiswa

## 4.3 Rancangan Pengujian

Bagian ini disusun untuk menyusun rancangan dari pengujian yang akan dilakukan diantaranya menggunakan *black box testing, user acceptance testing,* postman, dan kuesioner.

### 4.3.1 *Black Box Testing*

Rancangan pengujian pertama yaitu *black box testing* yang dilakukan dengan cara membuat suatu kasus agar mengetahui *end user* apakah hasil yang didapatkan sesuai dengan yang diharapkan atau tidak, berikut kasus yang diberikan dalam bentuk tabel:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Pengujian** | **Keterangan** | **Hasil** | **Status** |
| 1 | *User* melakukan *login* | Mengisi *field* username dan password sesuai dengan akun yang telah didaftarkan | Berhasil *login* dan masuk ke halaman beranda sesuai *role* | Berhasil atau Gagal |
| Tidak mengisi *field* username dan password sesuai akun yang telah didaftarkan | Gagal *login* dan mendapatkan pesan *error* |
| 2 | *User* dengan *role* administrator menambah *user* | Mengisi *form* *user* sesuai ketentuan | Berhasil menambah *user* ke *database* |
| Tidak mengisi *form* *user* sesuai ketentuan | Gagal menambah *user* dan mendapatkan pesan *error* |
| 3 | *User* dengan *role* administrator menghapus *user* | Memilih *user* dan konfirmasi menekan tombol hapus | Berhasil menghapus *user* dari *database* |
| Memilih *user* dan tidak konfirmasi menekan tombol hapus | Gagal menghapus *user* dari *database* |
| 4 | *User* dengan *role* administrator dan mahasiswa mengedit *user* | Administrator memilih *user* dan mengedit *form* *user* sesuai ketentuan | Berhasil mengedit *user* ke *database* |
| Mahasiswa mengedit data pribadi | Berhasil mengedit data pribadi ke *database* |
| 5 | *User* dengan *role* administrator, dosen dan mahasiswa melihat *user* | Administrator dan dosen melihat semua data *user* | Berhasil melihat semua data *user* sesuai *database* |
| Mahasiswa melihat data pribadi | Berhasil melihat data pribadi sesuai *database* |
| 6 | *User* dengan *role* dosen menilai nilai mahasiswa | Memilih mahasiswa dan menilai nilai mahasiswa sesuai ketentuan | Berhasil menilai nilai mahasiswa ke *database* |
| Memilih mahasiswa dan tidak menilai nilai mahasiswa sesuai ketentuan | Gagal menilai nilai mahasiswa dan mendapatkan pesan *error* |
| 7 | *User* dengan *role* *scrum master* melakukan menambah *team* | Mengisi *form* *team* sesuai ketentuan | Berhasil menambah *team* ke dalam *database* |
| Tidak mengisi *form* *team* sesuai ketentuan | Gagal menambah *team* dan mendapatkan pesan *error* |
| 8 | *User* dengan *role* *scrum master* melakukan *assign* mahasiswa ke *team* | *Assign* mahasiswa ke dalam *team* yang tersedia sesuai ketentuan | Berhasil *assign* mahasiswa ke dalam *team* |
| *Assign* mahasiswa ke dalam *team* yang tersedia tidak sesuai ketentuan | Gagal *assign* mahasiswa ke dalam *team* dan mendapatkan pesan *error* |
| 9 | *User* dengan *role* *scrum master* melakukan *assign project* ke *team* | *Assign project* ke dalam *team* yang tersedia sesuai ketentuan | Berhasil *assign* *project* ke dalam *team* |
| *Assign project* ke dalam *team* yang tersedia tidak sesuai ketentuan | Gagal *assign* *project* ke dalam *team* dan mendapatkan pesan *error* |
| 10 | *User* dengan *role* *scrum master* menghapus *team* | Memilih *team* dan konfirmasi menekan tombol hapus | Berhasil menghapus *team* dari *database* |
| Memilih *team* dan tidak konfirmasi menekan tombol hapus | Gagal menghapus *team* dari *database* |
| 11 | *User* dengan *role* *scrum master* mengedit *team* | Memilih *team* dan mengedit *form* *team* sesuai ketentuan | Berhasil mengedit *team* ke *database* |
| Memilih *team* dan tidak mengedit *form* *team* sesuai ketentuan | Gagal mengedit *team* dan mendapatkan pesan *error* |
| 12 | *User* dengan *role* *scrum master*, dosen dan mahasiswa melihat *team* | Melihat data *team* | Berhasil melihat data *team* sesuai *database* |
| 13 | *User* dengan *role* dosen menilai nilai team | Memilih *team* dan menilai nilai *team* sesuai ketentuan | Berhasil menilai nilai *team* ke *database* |
| Memilih *team* dan tidak menilai nilai *team* sesuai ketentuan | Gagal menilai nilai *team* dan mendapatkan pesan *error* |
| 14 | *User* dengan *role* *product owner* melakukan *request project* | Mengisi *form* *project* sesuai ketentuan | Berhasil menambah *project* ke dalam *database* |
| Tidak mengisi *form* *project* sesuai ketentuan | Gagal menambah *project* dan mendapatkan pesan *error* |
| 15 | *User* dengan *role*  *scrum master* menentukan sprint dari *project* yang di*request* | Mengisi *form* *sprint* ke *project* sesuai ketentuan | Berhasil menambah *sprint* ke *project* ke dalam *database* |
| Tidak mengisi *form sprint* ke *project* sesuai ketentuan | Gagal menambah *sprint* ke *project* dan mendapatkan pesan *error* |
| 16 | *User* dengan *role* *product owner, scrum master*, dosen dan mahasiswa melihat *project* | Melihat data *project* | Berhasil melihat data *project* sesuai *database* |
| 17 | *User* melakukan *logout* | *User* menekan tombol *logout* | *User* berhasil *logout* |

Tabel 4 : Rancangan Pengujian *Black Box Testing*

### 4.3.2 *User Acceptance Testing*

Rancangan pengujian kedua yaitu *user acceptance testing* yang dilakukan untuk mengetahui apakah *end user* dapat melaksanakan tugasnya sesuai *user requirement* atau tidak. Dalam pembahasan sebelumnya terdapat 5 *end user* yaitu administrator, dosen, *scrum master, product owner,* dan mahasiswa, berikut pengujian yang akan diberikan melalui bentuk tabel:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Administrator** | | | | |
| **No** | **Pengujian** | **Deskripsi** | **Hasil** | **Komentar** |
| 1 | *Login* | Melakukan *login* dengan akun yang telah didaftarkan dan masuk ke halaman administrator | Berhasil atau Gagal | ... |
| 2 | Kelola *User* | Administrator dapat menambah dan menghapus *user* |
| 3 | Mengedit *User* | Administrator dapat mengedit data *user* |
| 4 | Melihat *User* | Administrator dapat melihat data *user* |
| 5 | *Logout* | Melakukan *logout* dan masuk ke halaman *login* |
| **Dosen** | | | | |
| **No** | **Pengujian** | **Deskripsi** | **Hasil** | **Komentar** |
| 1 | *Login* | Melakukan *login* dengan akun yang telah didaftarkan dan masuk ke halaman dosen | Berhasil atau Gagal | ... |
| 2 | Melihat *User* | Dosen dapat melihat data *user* |
| 3 | Menilai Nilai *User* | Dosen dapat menilai nilai *user* khusus mahasiswa |
| 4 | Melihat *Team* | Dosen dapat melihat data *team* |
| 5 | Menilai Nilai *Team* | Dosen dapat menilai nilai team |
| 6 | Melihat *Project* | Dosen dapat melihat data *project* |
| 7 | *Logout* | Melakukan *logout* dan masuk ke halaman *login* |
| ***Scrum Master*** | | | | |
| **No** | **Pengujian** | **Deskripsi** | **Hasil** | **Komentar** |
| 1 | *Login* | Melakukan *login* dengan akun yang telah didaftarkan dan masuk ke halaman s*crum master* | Berhasil atau Gagal | ... |
| 2 | Kelola *Team* | *Scrum master* dapat menambah, mengedit, dan menghapus *team* miliknya |
| 3 | *Assign* Mahasiswa Ke *Team* | *Scrum master* dapat *assign* mahasiswa ke *team* miliknya |
| 4 | *Assign* *Project* Ke *Team* | *Scrum master* dapat *assign project* ke *team* miliknya |
| 5 | Melihat *Team* | *Scrum master* dapat melihat data *team* miliknya |
| 6 | Menentukan *Sprint* | *Scrum master* dapat menentukan *sprint* dari *project* yang di*request* |
| 7 | Melihat *Project* | *Scrum master* dapat melihat data *project* miliknya |
| 8 | *Logout* | Melakukan *logout* dan masuk ke halaman *login* |
| ***Product Owner*** | | | | |
| **No** | **Pengujian** | **Deskripsi** | **Hasil** | **Komentar** |
| 1 | *Login* | Melakukan *login* dengan akun yang telah didaftarkan dan masuk ke halaman *product owner* | Berhasil atau Gagal | ... |
| 2 | *Request* *Project* | *Product owner* dapat melakukan *request project* |
| 3 | *Logout* | Melakukan *logout* dan masuk ke halaman *login* |
| **Mahasiswa** | | | | |
| **No** | **Pengujian** | **Deskripsi** | **Hasil** | **Komentar** |
| 1 | *Login* | Melakukan *login* dengan akun yang telah didaftarkan dan masuk ke halaman mahasiswa | Berhasil atau Gagal | ... |
| 2 | Mengedit *User* | Mahasiswa dapat mengedit *user* miliknya |
| 3 | Melihat *User* | Mahasiswa dapat melihat *user* miliknya |
| 4 | Melihat *Team* | Mahasiswa dapat melihat *team* miliknya |
| 5 | Melihat *Project* | Mahasiswa dapat melihat *project* miliknya |
| 6 | *Logout* | Melakukan *logout* dan masuk ke halaman *login* |

Tabel 5 : Rancangan Pengujian *User Acceptance Testing*

### 4.3.3 Postman

Rancangan pengujian ketiga yaitu postmanuntuk menguji *Uniform Resource Locator* atauURL dari setiap fitur berfungsi atau tidak, berikut pengujian pada postman yang akan diberikan melalui bentuk tabel:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Pengujian** | **URL** | **Metode** | **Hasil** |
| 1 | *Login* | ... | ... | Berhasil atau Gagal |
| 2 | Kelola *User* |
| 3 | Mengedit *User* |
| 4 | Melihat *User* |
| 5 | Menilai Nilai *User* |
| 6 | *Kelola Team* |
| 7 | *Assign* Mahasiswa Ke *Team* |
| 8 | *Assign* *Project* Ke *Team* |
| 9 | *Melihat Team* |
| 10 | *Menilai Nilai Team* |
| 11 | *Request Project* |
| 12 | *Menentukan Sprint* |
| 13 | *Melihat Project* |
| 14 | *Logout* |

Tabel 6 : Rancangan Pengujian Postman

### 4.3.4 Kuesioner

Rancangan pengujian terakhir yaitu kuesioneruntuk menilai aplikasi Link-Match STT-NF modul dosen berbasis web yang telah dibuat. Kuesioner ini menggunakan skala likert dengan 4 pilihan yaitu Sangat Tidak Setuju (STS) dengan skor 1, Tidak Setuju (TS) dengan skor 2, Setuju (S) dengan skor 3, dan Sangat Setuju (SS) dengan skor 4, berikut kuesioner yang akan diberikan melalui bentuk tabel:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **STS** | **TS** | **S** | **SS** |
| 1 | Aplikasi dapat diakses dengan lancar |  |  |  |  |
| 2 | Tampilan pada aplikasi mudah dipahami |  |  |  |  |
| 3 | Fitur pada aplikasi dapat digunakan |  |  |  |  |
| 4 | Fitur pada aplikasi sesuai kebutuhan *user* |  |  |  |  |
| 5 | Fitur pada aplikasi mudah dipahami |  |  |  |  |

Tabel 7 : Rancangan Pengujian Kuesioner

# BAB V

# IMPLEMENTASI

## 5.1 Implementasi Sistem

Deskripsi...

### 5.1.1 Persiapan

Deskripsi

### 5.1.2 Pengembangan

Deskripsi

## 5.2 Implementasi Scrum

Deskripsi...

### 5.2.1 *Product Backlog*

Deskripsi

### 5.2.2 *Sprint Planning*

Deskripsi

### 5.2.3 *Sprint Progress*

Deskripsi

### 5.2.4 *Sprint Review*

Deskripsi

## 5.3 Implementasi Pengujian

Deskripsi...

### 5.3.1 *Black Box Testing*

Deskripsi

### 5.3.2 *User Acceptance Testing*

Deskripsi

### 5.3.3 Postman

Deskripsi

### 5.3.4 Kuesioner

Deskripsi

# BAB VI

# KESIMPULAN DAN SARAN

## 6.1 Kesimpulan

Aplikasi Link-Match STT-NF modul dosen berbasis web dikembangkan dengan tujuan melakukan *user management, team management,* dan *project management*, dalam mendukung program *Link and Match* di STT-NF. Dalam pengembangan aplikasi ini dilalui oleh tahapan analisis, perancangan, rancangan pengujian, dan implementasi. Tahap analisis dilakukan untuk mengetahui siapa penggunanya dan apa saja kebutuhannya, kemudian pada tahapan perancangan dibuat *entity relationship diagram, use case diagram, activity diagram*, dan *user interface* sebagai bahan pendukung dalam pengembangan aplikasi. Tahapan selanjutnya yaitu merancang pengujian dengan menggunakan *black box testing, user acceptance testing,* postman, dan kuesioner, serta tahapan terakhir yaitu melakukan implementasi mulai dari implementasi sistem, scrum, dan pengujian. Hasil yang didapatkan dari aplikasi ini ...

## 6.2 Saran

Pada pengembangan aplikasi Link-Match STT-NF modul dosen membutuhkan integrasi sistem dengan modul lain sehingga diharuskan sinkron antara modul satu dengan modul lainnya. Integrasi sistem ini dilakukan supaya aplikasi Link-Match STT-NF dapat berfungsi dengan baik untuk program *Link and Match* di STT-NF. Oleh karena itu dalam pengembangan aplikasi ini diperlukan kerja sama yang baik antara pengembang satu dengan yang lainnya. Selain integrasi sistem, aplikasi Link-Match STT-NF modul dosen perlu dilakukan *deploy* ke hosting supaya digunakan secara *online* sehingga *user* dapat mengakses aplikasi tersebut.

# DAFTAR PUSTAKA

Alamsyah, A., & Yuliansyah, H. (2016). RANCANG BANGUN APLIKASI WEB UNTUK MANAJEMEN PROYEK BERBASIS SCRUM. *Jurnal Sarjana Teknik Informatika*, *4*(1), 1–12.

Aminullah, R., Suprayogi, A., & Sukmono, A. (2018). APLIKASI PGROUTING UNTUK PENENTUAN RUTE ALTERNATIF MENUJU WISATA BATIK DI KOTA PEKALONGAN BERBASIS WEBGIS. *Jurnal Geodesi Undip*, *7*(1), 109–119.

Azdy, R. A., & Azhari. (2012). IMPLEMENTASI SCRUM PADA PENGEMBANGAN SOFTWARE TERDISTRIBUSI. *UPN ”Veteran” Yogyakarta*, 32–37.

Badan Pusat Statistik. (2019). *Berita Resmi Statistik*. https://www.bps.go.id/

Choiri, T., Dengen, N., & Islamiyah. (2017). WEB SERVICE UNTUK WEB PROFIL SMP NEGERI 2 RANTEPAO TORAJA UTARA. *Prosiding Seminar Ilmu Komputer dan Teknologi Informasi*, *2*(1), 91–98.

Destiningrum, M., & Adrian, Q. J. (2017). SISTEM INFORMASI PENJADWALAN DOKTER BERBASSIS WEB DENGAN MENGGUNAKAN FRAMEWORK CODEIGNITER (STUDI KASUS: RUMAH SAKIT YUKUM MEDICAL CENTRE). *Jurnal TEKNOINFO*, *11*(2), 30–37.

Disas, E. P. (2018). Link and Match sebagai Kebijakan Pendidikan Kejuruan. *Jurnal Penelitian Pendidikan*, 231–242.

Firman, A., Wowor, H. F., & Najoan, X. (2016). Sistem Informasi Perpustakaan Online Berbasis Web. *Jurusan Teknik Elektro Fakultas Teknik UNSRAT*, *5*(2), 29–36.

Handika, I. G., & Purbasari, A. (2019). Pemanfaatan Framework Laravel Dalam Pembangunan Aplikasi E-Travel Berbasis Website. *STMIK Atma Luhur Pangkalpinang*, 1329–1334.

Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Jurnal Informatika Mulawarman*, *6*(1), 1–15.

I Gede Handika, & Ayi Purbasari. (2018). Pemanfaatan Framework Laravel Dalam Pembangunan Aplikasi E-Travel Berbasis Website. *STMIK Atma Luhur Pangkalpinang*, 1329–1334.

Kusuma, A. W., Santoso, N., & Soebroto, A. A. (2019). Aplikasi Manajemen Multi Proyek menggunakan Metode Scrum. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, *3*(9), 8905–8913.

Maryuliana, Subroto, I. M. I., & Haviana, S. F. C. (2016). Sistem Informasi Angket Pengukuran Skala Kebutuhan Materi Pembelajaran Tambahan Sebagai Pendukung Pengambilan Keputusan Di Sekolah Menengah Atas Menggunakan Skala Likert. *Jurnal Transistor Elektro dan Informatika*, *1*(2), 1–12.

Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus: Aplikasi Prediksi Kelulusan SNMPTN). *Jurnal Ilmiah Teknologi Informasi Terapan*, *1*(3), 31–36.

Purnomo, P., & Palupi, M. S. (2016). PENGEMBANGAN TES HASIL BELAJAR MATEMATIKA MATERI MENYELESAIKAN MASALAH YANG BERKAITAN DENGAN WAKTU, JARAK DAN KECEPATAN UNTUK SISWA KELAS V. *Jurnal Penelitian (Edisi Khusus PGSD)*, *20*(2), 151–157.

Sekolah Tinggi Teknologi Terpadu Nurul Fikri. (2019, Desember 18). *Program Class Link & Match di STT Terpadu Nurul Fikri* [News]. https://nurulfikri.ac.id. https://nurulfikri.ac.id/program-class-link-match-di-stt-terpadu-nurul-fikri/

Sholeh, M. (2007). PERMINTAAN DAN PENAWARAN TENAGA KERJA SERTA UPAH : TEORI SERTA BEBERAPA POTRETNYA DI INDONESIA. *Jurnal Ekonomi & Pendidikan Universitas Negeri Yogyakarta*, *4*(1), 62–75.

Sibagariang, S. (2016). PENERAPAN WEB SERVICE PADA PERPUSTAKAAN BERBASIS ANDROID. *Jurnal Mahajana Inforamasi*, *1*(1), 28–32.

Suendri. (2018). PENERAPAN KONSEP MODEL VIEW CONTROLLER PADA PERANCANGAN SISTEM MANAJEMEN SOFTWARE BERBASIS WEB. *JISTech*, *3*(2), 36–45.

Surahman, F., Ikhsan, S. H. A., & Kusumah, F. S. F. (2018). RANCANG BANGUN WEB SERVICE UNTUK TRANSAKSI DATA PADA APLIKASI SAHABAT JASA DENGAN METODE REST. *Seminar Nasional Teknologi Informasi Universitas Ibn Khaldun Bogor*, 256–264.

Tarmizi, M. (2018). *PEMBANGUNAN APLIKASI MONITORING BUDGET EVENT ORGANIZER PADA PT INDI NOTOKREASI BERBASIS WEB MENGGUNAKAN PHP FRAMEWORK LARAVEL*. SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI.

Utomo, D. W., Kurniawan, D., & Astuti, Y. P. (2018). TEKNIK PENGUJIAN PERANGKAT LUNAK DALAM EVALUASI SISTEM LAYANAN MANDIRI PEMANTAUAN HAJI PADA KEMENTERIAN AGAMA PROVINSI JAWA TENGAH. *Jurnal SIMETRIS*, *9*(2), 731–746.

# LAMPIRAN

* Gambar 1 : Icon made by Nhor Phai, xnimrodx, Freepik, Pause08, Payungkead, itim2101, and Iconixar from [www.flaticon.com](http://www.flaticon.com/)
* Gambar 5 : Icon made by Iconixar, Freepik, Good Ware, Kiranshastry, and xnimrodx from [www.flaticon.com](http://www.flaticon.com/)
* Gambar 7 : Icon made by Freepik, srip, and Good Ware from [www.flaticon.com](http://www.flaticon.com/)
* Gambar 8 : Icon made by Freepik, Vitaly Gorbachev, and Eucalyp from [www.flaticon.com](http://www.flaticon.com/)
* Gambar 9 : Icon made by Freepik and srip from [www.flaticon.com](http://www.flaticon.com/)
* Gambar 10 : Icon made by surang, Freepik, Eucalyp, and icongeek26 from [www.flaticon.com](http://www.flaticon.com/)
* Gambar 11 : Icon made by Freepik, srip, Darlus Dan, monkik, surang, and Eucalyp from [www.flaticon.com](http://www.flaticon.com/)