



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**RANCANG BANGUN APLIKASI *LINK-MATCH KAMPUS*
STT-NF MODUL DOSEN BERBASIS *MOBILE*
MENGGUNAKAN *FLUTTER***

TUGAS AKHIR

**ARDITH LUTFIAWAN
0110217068**

**PROGRAM STUDI TEKNIK INFORMATIKA
DEPOK
FEBRUARI 2021**



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

RANCANG BANGUN APLIKASI *LINK-MATCH KAMPUS* STT-NF MODUL DOSEN BERBASIS *MOBILE* MENGGUNAKAN *FLUTTER*

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana Komputer (S.Kom.)**

**Ardith Lutfiawan
0110217068**

**PROGRAM STUDI TEKNIK INFORMATIKA
DEPOK
FEBRUARI 2021**

HALAMAN PERNYATAAN ORISINALITAS

**Tugas Akhir ini adalah hasil karya penulis,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Ardith Lutfiawan

NIM : 0110217068

Depok, 11 Februari 2021

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Ardith Lutfiawan
NIM : 0110217068
Program Studi : Teknik Informatika
Judul : Rancang Bangun Aplikasi *Link-Match* Kampus STT-NF Modul Dosen Berbasis *Mobile* Menggunakan *Flutter*

Telah berhasil dipertahankan di hadapan Dewan Pengaji 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

(Sirojul Munir, S.Si., M.Kom.)

Pengaji I

Pengaji II

(Hilmy Abidzar Tawakal, ST., M.Kom.) (Tifani Nabarian, S.Kom., M.T.I)

Ditetapkan di : Depok
Tanggal : 11 Februari 2021

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah *subhanahu wa ta'ala*, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/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 skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. 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 Dr. Lukman Rosyidi, M.T., M.M., selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Ibu Tifani Nabarian, S.Kom., M.T.I., selaku ketua program studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Bapak Sirojul Munir, S.Si., M.Kom., 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.

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 membalaq segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 11 Februari 2021

Penulis

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

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

Nama : Ardith Lutfiawan

NIM : 0110217068

Program Studi : Teknik Informatika

Jenis karya : Tugas Akhir

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

RANCANG BANGUN APLIKASI *LINK-MATCH* KAMPUS STT-NF MODUL DOSEN BERBASIS MOBILE MENGGUNAKAN *FLUTTER*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif 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 : 11 Februari 2021

Yang menyatakan

(Ardith Lutfiawan)

ABSTRAK

Nama : Ardith Lutfiawan
NIM : 0110217068
Program Studi : Teknik Informatika
Judul : Rancang Bangun Aplikasi *Link-Match* Kampus STT-NF Modul Dosen Berbasis *Mobile* Menggunakan *Flutter*

Link and match dalam dunia pendidikan adalah suatu bentuk penyelenggaraan pendidikan dalam mempersiapkan peserta didik untuk dapat memenuhi kebutuhan kerja. Sebagai kampus yang memperhatikan kebutuhan dunia kerja yang semakin meningkat, STT Terpadu Nurul Fikri membuka kelas *link-match* sebagai pengembangan mata kuliah, program ini bertujuan untuk mempersiapkan mahasiswa agar terbiasa dengan lingkungan kerja. Saat ini program *link-match* masih menggunakan aplikasi pihak ketiga berbasis *website* dan cukup banyak aplikasi lainnya yang digunakan. Hal ini menjadi tantangan untuk membuat aplikasi yang dapat diakses dengan mudah yaitu dengan menggunakan *smartphone* dan berfokus pada pengelolaan *project*. Tujuan penelitian ini yaitu membantu pelaksanaan program *link-match* yang berjalan di STT Terpadu Nurul Fikri. Dilakukan dengan menerapkan metode *scrum* sesuai dengan prinsip-prinsip yang terdapat pada metode pengembangan perangkat secara cepat. Perancangan sistem pada penelitian ini menggunakan UML *diagram* sebagai representasi dari sistem yang akan dibangun. Aplikasi ini dibangun menggunakan Flutter Framework dan RESTful API untuk bertukar data yang terintegrasi secara *real-time* dengan *website link-match*. Hasil penelitian ini adalah aplikasi berbasis Android dengan beberapa fitur untuk melakukan manajemen sprint, lihat informasi *project*, tim dan mahasiswa. Berdasarkan pengujian *User Acceptance Test* dan kuesioner, aplikasi ini memiliki tingkat keberhasilan maupun kepuasan mencapai lebih dari 75% dan dapat diambil kesimpulan bahwa aplikasi *link-match* modul dosen ini sudah cukup baik serta dapat membantu pelaksanaan program *link-match* di STT Terpadu Nurul Fikri.

Kata kunci :

Link and match, Scrum, Flutter Framework, RESTful API, STT Terpadu Nurul Fikri

ABSTRACT

Name : Ardith Lutfiawan
NIM : 0110217068
Study Program : Information Technology
Title : Design and Build Link-Match Mobile Application of Lecturer's Module in STT-NF

Link and match in the world of education is a form of organizing education in preparing students to be able to meet work needs. As a campus that pays attention to the growing needs of the world of work, STT Terpadu Nurul Fikri opens a link-match class as a course development, this program aims to prepare students to get used to the work environment. Currently, link-match programs still use website-based third-party applications and quite a lot of other applications are used. It is a challenge to create an application that can be accessed easily by using a smartphone and focusing on project management. The purpose of this research is to help the implementation of link-match program that runs in STT Terpadu Nurul Fikri. It is carried out by applying the scrum method in accordance with the principles contained in the method of rapid development of the device. The system design in this study used UML diagrams as a representation of the system. The application is built using the Flutter Framework and RESTful API to exchange real-time integrated data with link-match's websites. The result of this research is an Android-based application with several features to perform sprint management, see projects, teams and student information. Based on User Acceptance Test and questionnaire, this application has a success rate and satisfaction of more than 75% and can be concluded that the application of link-match lecturer module is good enough and can help the process of link-match program in STT Terpadu Nurul Fikri.

Keywords :

Link and match, Scrum, Flutter Framework, RESTful API, STT Terpadu Nurul Fikri

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR	v
ABSTRAK	2
ABSTRACT	3
DAFTAR ISI	4
DAFTAR GAMBAR	7
DAFTAR TABEL	9
BAB I PENDAHULUAN	10
1.1 Latar belakang	10
1.2 Identifikasi Masalah.....	12
1.3 Tujuan dan Manfaat Penelitian.....	12
1.4 Batasan Masalah	12
1.5 Sistematika Penulisan	12
BAB II LANDASAN TEORI	14
2.1 Tinjauan Pustaka.....	14
2.1.1 <i>Link-Match</i>	14
2.1.2 Model Pengembangan.....	15
2.1.3 <i>Tools</i> Pengembangan	19
2.1.4 Pengujian Sistem	24
2.2 Penelitian Terkait.....	26
2.2.1 Tabel Penelitian.....	26
2.2.2 Posisi Penelitian.....	27
BAB III METODE PENELITIAN	29
3.1 Tahapan Penelitian.....	29
3.1.1 Perumusan Masalah	30
3.1.2 Studi Literatur.....	30
3.1.3 Pengumpulan Data dan Informasi	30
3.1.4 Analisis Kebutuhan Sistem	30

3.1.5	Perancangan Sistem	31
3.1.6	Implementasi & <i>Testing</i>	31
3.1.7	Evaluasi	31
3.2	Rancangan Penelitian	31
3.2.1	Jenis Penelitian	31
3.2.2	Lingkungan Pengembangan	32
3.2.3	Alat dan Bahan	32
	BAB IV PERANCANGAN SISTEM	33
4.1	Analisis Sistem	33
4.1.1	<i>End-user</i>	33
4.1.2	Kebutuhan Sistem	33
4.1.3	<i>Product Backlog</i>	34
4.2	Perancangan Sistem	36
4.2.1	<i>Deployment Diagram</i>	36
4.2.2	<i>Entity Relationship Diagram</i>	37
4.2.3	<i>State Management</i>	39
4.2.4	<i>Use Case Diagram</i>	39
4.2.5	<i>Activity Diagram</i>	43
4.2.6	<i>User Interface</i>	47
4.3	Rencana Pengujian	56
4.3.1	<i>Black-box Testing</i>	56
4.3.2	<i>User Acceptance Testing</i>	59
4.3.3	Kuesioner	65
	BAB V IMPLEMENTASI DAN EVALUASI	67
5.1	Implementasi Scrum	67
5.1.1	<i>Scrum Team</i>	67
5.1.2	<i>Sprint Planning</i>	67
5.1.3	<i>Sprint Review</i>	87
5.1.4	Evaluasi <i>Scrum</i>	88
5.2	Pengujian Sistem	90
5.2.1	Hasil <i>Black-box Testing</i>	90

5.2.2	Hasil <i>User Acceptance Test</i>	93
5.2.3	Hasil Kuesioner	94
BAB VI KESIMPULAN DAN SARAN		96
6.1	Kesimpulan.....	96
6.2	Saran	96
DAFTAR PUSTAKA		98
LAMPIRAN		100

DAFTAR GAMBAR

Gambar 1.1 Penggunaan perangkat digital	10
Gambar 2.1 Tahapan <i>scrum</i>	17
Gambar 2.2 <i>Native architecture</i>	20
Gambar 2.3 <i>Web-view architecture</i>	20
Gambar 2.4 <i>Cross-platform architecture</i>	21
Gambar 2.5 <i>Flutter architecture</i>	22
Gambar 3.1 Tahapan penelitian.....	29
Gambar 4.1 <i>Deployment diagram</i>	37
Gambar 4.2 <i>Entity relationship diagram</i>	38
Gambar 4.3 Rancangan arsitektur BLoC	39
Gambar 4.4 <i>Use case diagram</i>	40
Gambar 4.5 <i>Use case diagram user management</i>	41
Gambar 4.6 <i>Use case diagram team management</i>	42
Gambar 4.7 <i>Use case project management</i>	43
Gambar 4.8 <i>Activity diagram</i>	44
Gambar 4.9 <i>Activity diagram role product owner</i>	45
Gambar 4.10 <i>Activity diagram role scrum master</i>	46
Gambar 4.11 <i>Activity diagram role dosen</i>	47
Gambar 4.12 <i>User interface login</i>	48
Gambar 4.13 <i>User interface view profile dan logout</i>	49
Gambar 4.14 <i>User interface dashboard</i>	49
Gambar 4.15 <i>User interface list project</i>	50
Gambar 4.16 <i>User interface detail project</i>	50
Gambar 4.17 <i>User interface list sprint</i>	51
Gambar 4.18 <i>User interface detail sprint</i>	51
Gambar 4.19 <i>User interface add sprint</i>	52
Gambar 4.20 <i>User interface memulai sprint</i>	52
Gambar 4.21 <i>User interface mengakhiri sprint</i>	53
Gambar 4.22 <i>User interface edit sprint</i>	53

Gambar 4.23 <i>User interface delete sprint</i>	54
Gambar 4.24 <i>User interface view tim</i>	54
Gambar 4.25 <i>User interface view mahasiswa</i>	55
Gambar 4.26 <i>User interface view nilai tim</i>	55
Gambar 4.27 <i>User interface view nilai mahasiswa</i>	56
Gambar 5.1 <i>Login user</i>	73
Gambar 5.2 Validasi <i>login user</i>	73
Gambar 5.3 <i>Splash screen</i>	74
Gambar 5.4 <i>View profile</i>	74
Gambar 5.5 <i>Logout user</i>	75
Gambar 5.6 <i>Dashboard scrum master</i> dan dosen.....	75
Gambar 5.7 <i>Dashboard product owner</i>	76
Gambar 5.8 <i>List project</i>	77
Gambar 5.9 <i>Detail project</i>	77
Gambar 5.10 <i>List sprint</i>	78
Gambar 5.11 <i>Detail sprint</i>	78
Gambar 5.12 <i>Add Sprint</i>	82
Gambar 5.13 <i>Edit sprint</i>	82
Gambar 5.14 <i>Delete sprint</i>	83
Gambar 5.15 <i>List tim</i>	83
Gambar 5.16 <i>Detail tim</i>	84
Gambar 5.17 <i>List mahasiswa</i>	84

DAFTAR TABEL

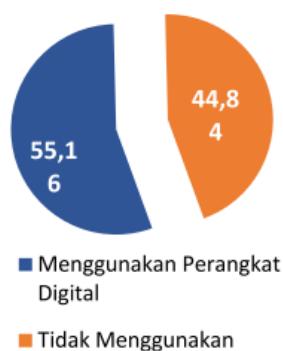
Tabel 1.1 Tabel penelitian terkait	26
Tabel 1.2 Posisi penelitian	27
Tabel 4.1 <i>End-user</i>	33
Tabel 4.2 <i>Product backlog</i>	34
Tabel 4.3 <i>Black-box testing</i>	56
Tabel 4.4 <i>UAT user product owner</i>	59
Tabel 4.5 <i>UAT user dosen</i>	61
Tabel 4.6 <i>UAT user scrum master</i>	63
Tabel 4.7 Bobot nilai jawaban	65
Tabel 4.8 Pertanyaan kuesioner.....	65
Tabel 5.1 <i>Scrum team</i>	67
Tabel 5.2 <i>Sprint 1</i>	68
Tabel 5.3 <i>Sprint 2</i>	68
Tabel 5.4 <i>Sprint 3</i>	69
Tabel 5.5 <i>BLoC sprint 3</i>	71
Tabel 5.6 <i>Sprint 4</i>	79
Tabel 5.7 <i>BLoC sprint 4</i>	80
Tabel 5.8 <i>Sprint 5</i>	85
Tabel 5.9 <i>BLoC sprint 5</i>	85
Tabel 5.10 <i>Sprint review</i>	87
Tabel 5.11 Hasil <i>black-box testing</i>	90
Tabel 5.12 Hasil pengujian <i>UAT user product owner</i>	93
Tabel 5.13 Hasil pengujian <i>UAT user dosen</i>	94
Tabel 5.14 Hasil pengujian <i>UAT user scrum master</i>	94
Tabel 5.15 Indeks persentase.....	94
Tabel 5.16 Hasil skala likert.....	95

BAB I

PENDAHULUAN

1.1 Latar belakang

Sumber daya manusia yang berkualitas merupakan suatu keharusan bagi sebuah bangsa di era globalisasi. Salah satu wahana untuk menciptakan sumber daya manusia yang berkualitas adalah bidang pendidikan. Pendidikan menjadi pondasi utama dalam membangun bangsa Indonesia menjadi negara maju dan berperadaban. Apabila pendidikan dibangun dengan baik dan berkelanjutan, niscaya terciptalah masyarakat yang sejahtera. Salah satu sektor penting yang secara langsung memberikan kontribusi terbesar dalam mengembangkan kualitas Sumber Daya Manusia (SDM) adalah sektor pendidikan (Ni Ayu Krisna Dewi et al., 2014).



Gambar 1.1 Penggunaan perangkat digital

Menurut (Badan Pusat Statistik, 2019), revolusi industri mengalami puncaknya saat ini dengan lahirnya teknologi digital yang masif dan berdampak terhadap hidup manusia di seluruh dunia. Revolusi industri terkini atau generasi keempat mendorong sistem otomatisasi di dalam semua proses aktivitas bahkan lapangan usaha sekalipun. Lebih dari setengah usaha di Indonesia telah menggunakan perangkat digital, baik *personal computer* (PC), *laptop*, *tablet*, ataupun *smartphone*. Lapangan usaha dengan penggunaan perangkat digital tertinggi adalah aktivitas professional, ilmiah, dan teknis sebesar 97,53% dari seluruh usaha di Indonesia. Berdasarkan jenis perangkat digital yang digunakan, sebesar 40,71% usaha menggunakan *smartphone* dari seluruh usaha

yang menggunakan perangkat digital. Perangkat yang paling banyak digunakan adalah *personal computer* (26,17%) , diikuti oleh *notebook* (18,47%), serta *tablet* (3,18%).

Teknologi internet yang semakin masif ini tidak hanya menghubungkan jutaan manusia di seluruh dunia tetapi juga telah menjadi basis bagi sistem pembelajaran yang terintegrasi secara *online*. Munculnya kursus-kursus yang disediakan secara *online*, portal diskusi *online* berbasis *website* dan Android, dan sistem informasi akademik menunjukkan integrasi aktivitas manusia dengan teknologi informasi dan pendidikan menjadi semakin meningkat.

Membangun sumber daya manusia yang berkualitas dan menciptakan pembelajaran yang terintegrasi secara *online* demi meningkatkan kualitas sumber daya manusia adalah sebuah tantangan dalam menerapkan *link-match* antara dunia kerja dan pendidikan formal.

Sebagai kampus yang memperhatikan kebutuhan industri yang semakin meningkat, STT Terpadu Nurul Fikri membuka kelas untuk *link-match* sebagai pengembangan mata kuliah, program *link-match* bertujuan untuk mempersiapkan mahasiswa agar terbiasa dengan lingkungan kerja yang akan mereka hadapi suatu saat kelak dan yang paling penting adalah mempersiapkan mahasiswa agar dapat memenuhi kebutuhan industri (*Program Class Link & Match Di STT Terpadu Nurul Fikri*, 2019).

Sebagai salah satu *pilot project*, program ini masih menggunakan aplikasi pihak ketiga berbasis *website* dan cukup banyak aplikasi yang digunakan. Hal ini menjadi tantangan untuk mengembangkan aplikasi yang dapat diakses dengan mudah yaitu dengan menggunakan *smartphone* dan berfokus pada pengelolaan *project*.

Melihat permasalahan tersebut maka dilakukan penelitian mengenai perancangan aplikasi Android untuk pembelajaran *link-match* menggunakan metode *scrum* yang diharapkan dapat meningkatkan kemampuan belajar mahasiswa agar lebih siap untuk terjun ke dunia kerja juga membantu dalam pelaksanaan program *link-match* di STT Terpadu Nurul Fikri.

1.2 Identifikasi Masalah

Berdasarkan latar belakang tersebut, masalah yang akan diangkat dalam tugas akhir ini yaitu “**Bagaimana membuat aplikasi *link-match* berbasis Android untuk mendukung proses pembelajaran di STT Terpadu Nurul Fikri?**” dan dapat dirumuskan beberapa permasalahan yaitu:

1. Bagaimana proses pengembangan aplikasi *link-match* berbasis Android di lingkungan kampus STT Terpadu Nurul Fikri?
2. Apakah aplikasi yang dirancang dapat membantu pelaksanaan pembelajaran *link-match* di STT Terpadu Nurul Fikri?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini diantaranya:

1. Merancang dan membangun aplikasi *link-match* modul dosen berbasis Android menggunakan *Flutter Framework*.
2. Memastikan fitur-fitur yang dibangun dapat berfungsi dengan baik untuk membantu pelaksanaan program *link-match* yang berjalan di STT Terpadu Nurul Fikri.

1.4 Batasan Masalah

1. Pembuatan aplikasi *link-match* pada modul dosen ini meliputi *user management, team management, project management*.
2. Pembuatan aplikasi *link-match* modul dosen berbasis Android ini mengambil obyek penelitian di STT NF.
3. Penelitian ini tidak melakukan pengujian keamanan aplikasi.

1.5 Sistematika Penulisan

Untuk memudahkan dalam memahami tugas akhir ini secara keseluruhan, maka sistematika penulisan tugas akhir yang digunakan pada penelitian ini yaitu:

1. BAB I PENDAHULUAN, bab ini berisi latar belakang, perumusan masalah, tujuan dan manfaat penelitian, batasan masalah, dan sistematika penulisan.

2. BAB II LANDASAN TEORI, menjelaskan berbagai teori pendukung yang digunakan untuk membuat aplikasi *link-match* berbasis Android menggunakan *Flutter Framework*.
3. BAB III METODOLOGI PENELITIAN, bab ini berisi tahapan yang dilakukan dalam pengembangan sistem aplikasi berbasis Android menggunakan *Flutter Framework*. Rancangan penelitian yang menjelaskan mengenai cara menyelesaikan permasalahan, jenis penelitian yang dilakukan, rancangan analisis dan pengumpulan data, tempat lokasi serta waktu dilaksanakannya penelitian.
4. BAB IV PERANCANGAN SISTEM, berisi analisis sistem, perancangan sistem, rencana pengujian aplikasi sebelum proses penulisan kode program.
5. BAB V IMPLEMENTASI DAN EVALUASI, berisi implementasi metodologi *scrum* yaitu pengembangan produk aplikasi dengan membuatnya menjadi bagian-bagian yang harus diselesaikan. Di tahap inilah penulisan kode program dan pengujian aplikasi juga dilaksanakan.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

2.1.1 *Link-Match*

A. Pengertian *Link-Match*

Istilah *link and match* bukanlah suatu hal yang baru muncul belakangan ini melainkan istilah ini telah dikenalkan oleh Menteri Pendidikan dan Kebudayaan (1993-1998), Wardiaman Djojonegoro. Menurut (Herman, 2016), *link and match* dalam dunia pendidikan adalah suatu bentuk penyelenggaraan pendidikan dalam mempersiapkan peserta didik untuk dapat memenuhi kebutuhan kerja. Lebih jauh lagi Wardiaman mengungkapkan bahwa Pendidikan harus mulai berubah dari *supply minded* (orientasi jumlah) menjadi *demand minded* (kebutuhan), sangat relevan apabila kompetensi yang dihasilkan dari proses belajar bila dikaitkan dengan kesiapan kerja di masa mendatang yang akan banyak rintangannya seiring kemajuan teknologi.

Menurut (Disas, 2018), *link and match* adalah konsep yang menjadi landasan agar keterkaitan antara lembaga pendidikan dan dunia kerja dapat dilaksanakan. Menurutnya, konsep ini ideal dalam menciptakan pemasok tenaga kerja dengan industri sebagai penggunanya sehingga selain terhubung juga dapat berjalan secara serasi. Dengan adanya hubungan timbal balik ini diharapkan lembaga pendidikan mulai mengacu pada konsep *link and match* sebagai kurikulum dalam membantu memenuhi kebutuhan kerja.

B. *Link-Match* di STT-NF

Menurut (*Program Class Link & Match Di STT Terpadu Nurul Fikri*, 2019), Program *link-match* resmi diterapkan di STT Terpadu Nurul Fikri pada bulan September tahun 2019. Program ini difasilitasi oleh *Student Learning and Incubator Business Center (SLIB-C)* STT NF yang diketuai oleh Yekti Wirani, S.T., M.TI sekaligus menjadi penanggung jawab *link-match*. Program ini diperuntukan bagi

mahasiswa semester 5 yang menggabungkan 3 mata kuliah sekaligus, yaitu Manajemen Proyek, Pemrograman *Mobile*, dan *Object-oriented Analysis and Design* (OOAD).

2.1.2 Model Pengembangan

A. Scrum

Scrum adalah metode untuk mengembangkan dan mengelola perangkat lunak (produk) yang kompleks secara cepat. Prinsip *scrum* sesuai dengan prinsip-prinsip yang terdapat pada metode pengembangan perangkat secara cepat yang digunakan untuk menuntun kegiatan pengembangan perangkat lunak, seperti: pemenuhan kebutuhan, analisa, desain, dan penyampaian (*delivery*). Kerangka kerja ini memungkinkan para pengembang aplikasi secara bersamaan menciptakan produk bernilai tinggi dengan tetap memperhatikan aspek produktivitas dan kreativitas (Saepurrahman et al., 2019).

Scrum dikembangkan untuk mengelola dan mengembangkan produk. *Scrum* telah digunakan secara meluas di seluruh dunia untuk (Schwaber & Sutherland, 2017):

- Meneliti dan menggali potensi pasar, teknologi, dan kemampuan produk;
- Mengembangkan produk dan peningkatan-peningkatannya;
- Merilis produk dan peningkatan-peningkatannya, sesering mungkin;
- Mengembangkan dan memelihara operasional sistem komputasi awan (daring, keamanan, sesuai permintaan) dan lingkungan operasional lain untuk penggunaan produk; dan,
- Mengelola dan memperbarui sebuah produk.

❖ Scrum Team

Menurut (Schwaber & Sutherland, 2017), dalam proses implementasi pada suatu proyek menggunakan metode *scrum*, ada tahapan-tahapan pengembangan yang harus dilakukan oleh *Scrum Team*. *Scrum Team* terdiri dari *Product Owner*, *Development Team* dan *Scrum Master*.

- ***Product Owner***

Product Owner adalah orang yang bertanggung jawab untuk memaksimalkan nilai bisnis dari produk yang dihasilkan oleh *Development Team*. Cara melakukannya sangat bervariasi antar organisasi, *Scrum Team* dan individu. *product owner* adalah satu-satunya orang yang bertanggung jawab dalam pengelolaan *product backlog*.

- ***Development Team***

Development Team terdiri dari para ahli profesi yang bekerja untuk menghasilkan sebuah produk. Bagian ini memiliki peran penting dalam melaporkan hasil kerja dan diberikan wewenang untuk menyusun dan mengelola pekerjaan mereka sendiri.

- ***Scrum Master***

Scrum Master bertanggung jawab untuk mengenalkan dan menyokong penggunaan *scrum* dalam pengembangan proyek. *Scrum master* biasanya membantu dengan memberikan edukasi agar sebuah tim dapat memahami teori, praktik, dan aturan main dalam *scrum*.

❖ **Artefak Scrum**

Seiring dengan berjalannya proses pengembangan *scrum*, ada bagian-bagian penting yang perlu diperhatikan dalam pelaksanaan metode *scrum* ini agar terciptanya transparansi dan dapat menyesuaikan dengan tujuan kerja.

- ***Product Backlog***

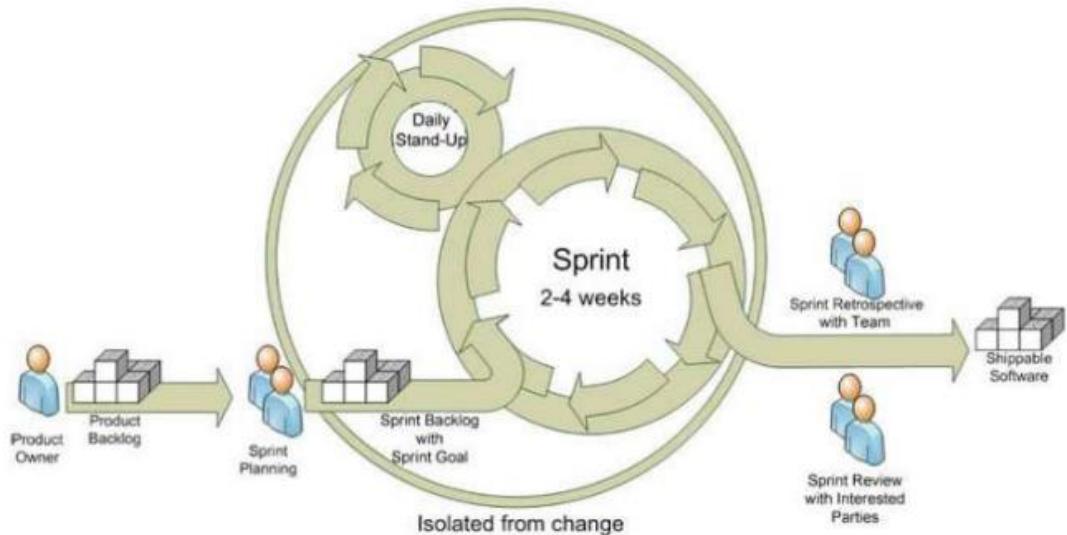
Product Backlog memiliki daftar terurut yang dibutuhkan di dalam produk beserta dengan rincian prioritasnya yang disebut *user story*, urutan teratas dalam *product backlog* item biasanya lebih jelas dan lebih detail dibandingkan dengan item di urutan lebih bawah (Ardiansyah, 2017).

- ***Sprint Backlog***

Sprint Backlog adalah daftar kumpulan dari *product backlog* yang terpilih untuk diselesaikan pada *sprint* yang berlangsung. *Sprint Backlog* menampilkan pekerjaan-pekerjaan apa saja yang akan diselesaikan oleh *Development Team*

untuk memastikan proses pengembangan berjalan dengan semestinya dan mencapai target (Schwaber & Sutherland, 2017).

❖ Tahapan Scrum



Gambar 2.1 Tahapan scrum

- **Sprint**

Inti dari *Scrum* terletak pada *sprint*, yaitu sebuah batasan waktu dengan durasi satu bulan atau kurang, dimana dilakukan secara iteratif dalam proses pengembangan produk dan bersifat konsisten sepanjang *sprint* berlangsung. *Sprint* merupakan pembungkus untuk semua tahapan formal (*event*) dalam *scrum* (Christmantara, 2013).

Scrum adalah kerangka kerja dengan menggunakan pendekatan iteratif, menurut (Christmantara, 2013), oleh karenanya ada 4 (empat) tahapan formal (*event*) untuk dilakukan, beberapa *event* tersebut, antara lain:

- **Sprint Planning**

Pekerjaan yang akan dikerjakan di *sprint* direncanakan pada saat *sprint planning*. Perencanaan ini adalah tahap awal setelah dilakukan secara kolaboratif oleh seluruh anggota *scrum team*. *Sprint planning* memiliki batasan waktu maksimal delapan jam untuk *sprint* yang berdurasi satu bulan. Untuk *sprint* yang lebih singkat, tahapan ini biasanya lebih singkat.

- **Daily Scrum Meeting**

Daily scrum adalah tahapan untuk *development team* yang memiliki batasan waktu 15 menit. Tahapan ini dilakukan setiap hari selama *sprint* berlangsung. Di tahapan ini, *development team* membuat rencana kerja untuk 24 jam ke depan. Tahapan ini mengoptimalkan kolaborasi dan performa dari tim dengan melakukan inspeksi pada pekerjaan yang dilakukan semenjak *daily scrum* sebelumnya dan melakukan prakiraan terhadap pekerjaan selanjutnya di dalam *sprint*. *Daily scrum* dilakukan di waktu dan tempat yang sama setiap harinya.

- ***Sprint Review***

Sprint review dilakukan di setiap akhir *sprint*. Pada saat *sprint review*, *scrum team* dan *stakeholder* meninjau kembali apa yang sudah diselesaikan selama *sprint* berlangsung. Setelah melakukan peninjauan maka hasilnya dapat menentukan pekerjaan yang belum terdaftar di *product backlog* dan jika sudah ada tetapi belum sesuai maka *development team* dapat membuat revisi dari hasil tersebut.

B. *Unified Modelling Language (UML)*

Unified Modelling Language (UML) merupakan alat perancangan sistem yang menggunakan bentuk khusus untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak (Dharwiyanti & Wahono, 2003). Dalam sebuah perancangan sistem menggunakan UML dapat disajikan dalam bentuk model yang terdiri dari kumpulan notasi untuk menggambarkan diagram. UML menerapkan konsep *Object-Oriented Analysis Design (OOAD)* dalam pemodelan informasi pada proses pembuatan perangkat lunak, seperti pemodelan bisnis dan sistem non perangkat lunak lainnya (Riansyah, 2015).

Penelitian ini menggunakan diagram UML dalam pembuatan aplikasi. Dengan menggunakan diagram UML, pengembang aplikasi dapat melakukan pemrograman kode yang biasa dikenal dengan sebutan *forward engineering*, yaitu metode membuat aplikasi dengan bantuan persyaratan yang diberikan dari model-model UML. Adapun diagram-diagram yang digunakan dalam UML antara lain (Dharwiyanti & Wahono, 2003):

- *Use Case Diagram*
- *Class Diagram*
- *Statechart Diagram*
- *Activity Diagram*
- *Sequence Diagram*
- *Collaboration Diagram*
- *Component Diagram*
- *Deployment Diagram*

2.1.3 Tools Pengembangan

A. Android

Menurut (J. F. DiMarzio, 2016), Android adalah sistem operasi yang berjalan pada perangkat *mobile*, menggunakan basis kernel linux yang dimodifikasi. Sistem Operasi ini dikembangkan oleh Google sejak tahun 2005 setelah membelinya dari Android, Inc. Google ingin OS Android bisa digunakan oleh semua, sehingga sebagian besar kode Android dirilis di bawah Lisensi Apache sumber terbuka yang artinya semua orang dapat dengan mudah menggunakan secara gratis. Dalam pengembangan aplikasi Android, pengembang hanya perlu mengembangkannya secara umum, dan harus dapat berjalan pada banyak perangkat yang berbeda, asalkan perangkat tersebut didukung menggunakan Android.

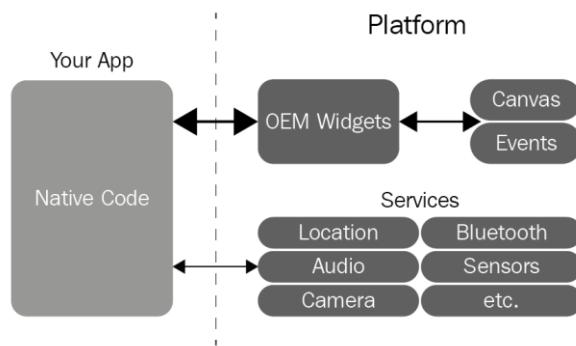
B. Framework

Menurut (Riansyah, 2015), *framework* adalah kerangka kerja. *Framework* menyediakan struktur dasar yang digunakan oleh *developer* untuk memudahkan pembangunan aplikasi berbasis web maupun *mobile*. *Framework* pada dasarnya dapat digunakan kembali sebagai bagian dari platform perangkat lunak yang ingin dibuat. Kerangka kerja yang terdapat pada sebuah *framework* dapat berupa program pendukung seperti fungsi, *class*, *tools*, antarmuka pemrograman aplikasi (API), dan

compiler. Umumnya di dalam *framework* juga telah tersedia fungsi untuk akses basis data, *authentication*, *templating*, *controls*, dan fungsi-fungsi lainnya.

Menurut (Prajyot Mainkar & Giordano, 2019), ada beberapa pendekatan pada pengembangan aplikasi berbasis *mobile*, namun pada akhir pendekatan-pendekatan tersebut adalah turunan dari pendekatan *native* ataupun *cross-platform*. Beberapa pendekatan tersebut antara lain:

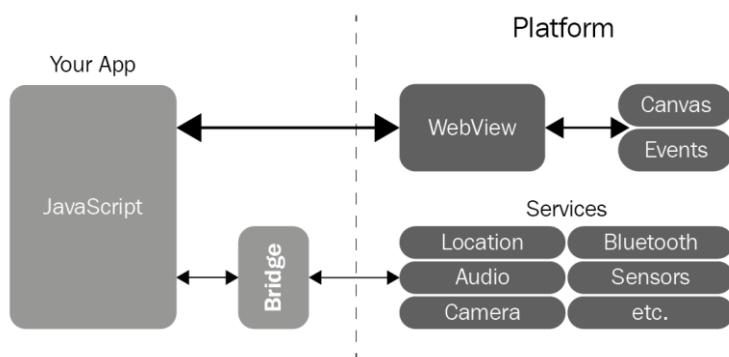
1. Native Platforms



Gambar 2.2 Native architecture

Pada diagram diatas, aplikasi dalam *native framework* berkomunikasi langsung dengan sistem. Hal ini membuat *native framework* dapat menjadi pilihan yang sangat baik dalam hal kegunaan. Pendekatan ini menggunakan bahasa yang berbeda seperti bahasa Kotlin atau Java untuk Android, Objective-C atau Swift untuk iOS dan SDK.

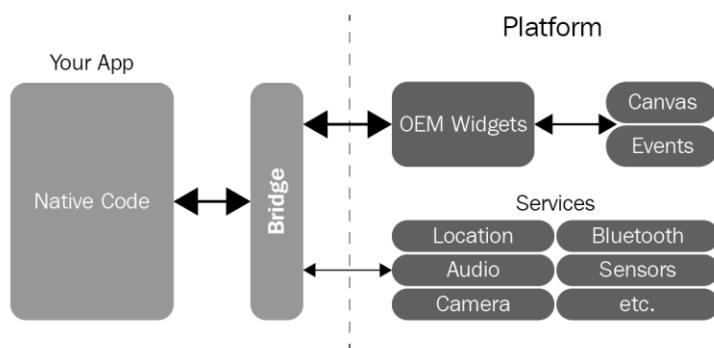
2. Web-View Systems



Gambar 2.3 Web-view architecture

Pada pendekatan ini, kita bisa mengakses aplikasi untuk Android dan iOS dari basis kode tunggal. Tampilan aplikasi dalam pendekatan ini disusun oleh *HTML rendering WebView* yang berarti aplikasi pada dasarnya adalah sebuah situs web. Sistem menggunakan jembatan untuk beralih antara JavaScript ke sistem aslinya. Contoh *framework* yang menerapkan pendekatan ini yaitu, Cordova-, Ionic-, PhoneGap-, dan *WebView-based frameworks*.

3. *Cross-platform Framework (React Native)*

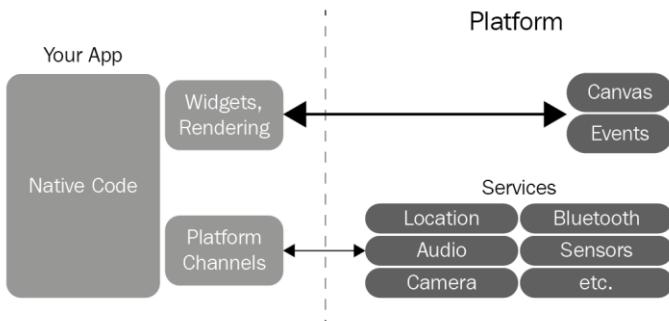


Gambar 2.4 *Cross-platform architecture*

React Native memperluas konsep jembatan (*bridge*) di sistem *WebView*, dan menggunakan tidak hanya untuk layanan, tetapi juga untuk membangun *widget*.

C. *Flutter Framework*

Flutter adalah *Software Development Kit (SDK)* buatan Google yang berfungsi untuk membuat aplikasi *mobile* menggunakan bahasa pemrograman Dart, baik untuk Android maupun iOS (Raharjo, 2019). Dengan *Flutter* aplikasi Android dan iOS dapat dibuat menggunakan basis kode dan bahasa pemrograman yang sama, yaitu Dart. Sebelumnya, aplikasi murni (*native*) untuk Android perlu dibuat dengan bahasa pemrograman Java atau Kotlin, sedangkan aplikasi iOS perlu dibuat menggunakan bahasa pemrograman Objective-C atau Swift. *Flutter* digunakan untuk mempermudah dan mempercepat proses pengembangan aplikasi *mobile* yang dapat berjalan di atas Android dan iOS, tanpa harus mempelajari dua bahasa pemrograman secara terpisah.



Gambar 2.5 Flutter architecture

Flutter menghilangkan jembatan (*bridge*) dan platform *OEM Widgets* (atau *DOM WebViews*) dan menggunakan *Widget Rendering* sebagai gantinya, *Flutter* menyediakan *widget* bawaan dan tidak menggunakan dari *native* juga bukan *DOM WebViews*.

Flutter bisa menjadi pilihan yang baik untuk pengembangan lintas-platform karena banyak fitur dan beberapa hal berbeda dari *framework* lain yang sejenis, seperti React Native, dan Ionic. *Flutter* juga dapat memberikan kelebihan tidak hanya bagi pengembang, tetapi juga bagi penggunanya (Prajyot Mainkar & Giordano, 2019).

- Untuk pengguna, *Flutter* memiliki antarmuka pengguna yang menarik untuk aplikasi, dan hal ini dapat meningkatkan pengalaman penggunaan.
- Untuk pengembang, *Flutter* memudahkan para pengembang untuk memasuki dunia membangun aplikasi *mobile*, karena sangat mudah untuk membangun aplikasi dengan *Flutter*. Keuntungannya tidak hanya mengurangi waktu untuk pengembangan aplikasi, tetapi juga mengurangi biaya dan kompleksitas pembuatan aplikasi.

Flutter dapat dikatakan sebagai produk Google yang masih relatif baru. Rilis perdana *Flutter*, versi *Alpha* (v.0.0.6), dipublikasikan pada bulan Mei 2017. Saat penelitian ini ditulis, versi *Flutter* terbaru adalah v.1.12.13+hotfix.9. Versi ini merupakan versi stabil terbaru yang dirilis oleh Google pada tanggal 4 April 2020.

D. Flutter State Management

Aplikasi yang dirancang dengan cara yang biasa akan kesulitan menangani aktivitas pengguna (Joyo et al., 2020). Dengan menerapkan paradigma pemrograman akan memungkinkan pengguna berinteraksi dengan aplikasi menjadi lebih reaktif (*asynchronous*). Salah satu paradigma yang sering digunakan adalah *reactive programming*, yaitu suatu paradigma yang menangani segala perubahan dalam aplikasi dilakukan dengan cara mensinkronisasi perubahan yang terjadi pada data secara global.

Flutter dikembangkan dengan konsep *reactive programming*, hal ini dikarenakan pada sebuah aplikasi *Flutter* terdapat *state management* yang mengatur tampilan antarmuka untuk mencerminkan kondisi/status aplikasi saat ini. Ada beberapa macam *state management* dalam pengembangan *reactive app*, berikut adalah beberapa diantaranya:

1. BLoC
2. Redux
3. Provider

E. BLoC

BLoC merupakan *state management* sekaligus *design pattern* yang memisahkan *event* dengan proses bisnis aplikasi (*Bloc, a Predictable State Management Library for Dart*, n.d.). BLoC dibuat untuk memungkinkan menggunakan kembali kode yang sama secara independen tanpa merubah baris kode di luar bisnis proses. Umumnya dalam implementasi BLoC terdapat 2 istilah yaitu *sinks* dan *streams*. *Widget* mengirim *events* ke BLoC melalui *sinks*, kemudian *widget* diberitahukan oleh BLoC melalui *streams*.

F. RESTful API

Menurut (Putra et al., 2019), API (*Application Programming Interface*) terdiri dari dokumentasi yang terdiri dari antarmuka, fungsi, kelas, dan sebagainya dalam membangun perangkat lunak. Sedangkan REST (*Representational State Transfer*)

merupakan struktur metode komunikasi yang menggunakan protokol HTTP untuk pertukaran data. RESTful API bertujuan untuk menjadikan sistem yang memiliki performa yang baik, cepat dan mudah untuk dikembangkan terutama dalam pertukaran dan komunikasi data pada aplikasi.

REST memanfaatkan empat (4) metode di HTTP, yaitu:

1. GET

Metode GET akan meminta representasi data dari sumber daya tertentu. Permintaan menggunakan GET hanya akan mengambil datanya saja. HEAD Metode HEAD meminta respon yang identik dengan permintaan GET, tetapi tanpa badan respons seperti *JSON (Javascript Object Notation)*.

2. POST

Metode POST digunakan untuk mengirimkan entitas ke sumber daya tertentu, apabila ada penambahan data dalam server hal ini menggunakan metode POST dalam penambahannya.

3. PUT

Metode PUT mirip seperti POST namun metode ini menggantikan semua representasi dari sumber daya tertentu dan pada umumnya tidak perlu menggunakan *form-data* seperti yang digunakan metode POST, PUT menggunakan *x-www-form-urlencoded* atau *JSON*.

4. DELETE

Metode DELETE menghapus sumber daya yang diinginkan.

2.1.4 Pengujian Sistem

A. *Black-box Testing*

Menurut (Munir et al., 2016), *Black-box testing* yaitu pengujian yang berfokus pada *functional requirement* sebuah sistem perangkat lunak. *Black-box testing* berfokus pada spesifikasi fungsional dari perangkat lunak, karena penguji hanya berkepentingan dengan fungsionalitas dan bukan implementasi perangkat lunak.

Black-box testing memungkinkan untuk menemukan hal-hal berikut (Mustaqbal et al., 2015):

1. Fungsi yang tidak benar atau tidak ada
2. Kesalahan antarmuka (*interface errors*)
3. Kesalahan pada struktur data dan akses basis data
4. Kesalahan performansi (*performance errors*)
5. Kesalahan inisialisasi dan terminasi

B. *User Acceptance Test (UAT)*

Menurut (Wardhono et al., 2015), pengujian *User Acceptance Test (UAT)* diimplementasikan dengan mengikutsertakan pengguna dalam mengoperasikan sistem untuk melihat seberapa besar tingkat kemudahan dan kepuasan pengguna dari aplikasi yang telah dikembangkan. Pengujian *UAT* juga berperan sebagai alat ukur kesuksesan dalam mengembangkan suatu sistem. Pada tahap pengujian *UAT* dilakukan dengan menggunakan *in-depth interview*, yang mana keterangan diperoleh untuk tujuan penelitian dengan cara tanya jawab antara pewawancara dengan responden (Wang et al., 2019).

C. Kuesioner

Menurut (Johan et al., 2019), kuesioner merupakan teknik pengumpulan data primer yang dilakukan dengan cara memberi seperangkat pertanyaan atau pernyataan tertulis kepada responden untuk dijawab. Kuesioner dapat digunakan untuk memperoleh informasi pribadi misalnya sikap, opini, harapan dan keinginan responden. Kuesioner dapat didistribusikan kepada responden dengan cara (Pujiastuti, 2010):

1. Langsung oleh peneliti (mandiri)
2. Dikirim lewat pos (*mailquestionair*)
3. Dikirim lewat komputer misalnya surat elektronik (*e-mail*)

D. Skala Likert

Skala likert adalah suatu skala psikometrik yang umum digunakan dalam kuesioner, dan merupakan kala yang paling banyak digunakan dalam riset berupa survei (Helmi et al., 2017). Dengan skala likert, variabel yang akan diukur dijabarkan

menjadi indikator variabel. Kemudian indikator tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen yang dapat berupa pertanyaan atau pernyataan.

Dalam penelitian ini skala likert mempunyai gradasi dari sangat positif sampai sangat negatif, peneliti menggunakan gradasi lima, yaitu: Sangat Setuju, Setuju, Kurang Setuju, Tidak Setuju Sangat Tidak Setuju, Tidak Setuju Sangat Tidak Setuju. Untuk jawaban tersebut diberi skor sebagai berikut:

- Respon dengan nilai skor 4 berarti Sangat Setuju
- Respon dengan nilai skor 3 berarti Setuju
- Respon dengan nilai skor 2 berarti Tidak Setuju
- Respon dengan nilai skor 1 berarti Sangat Tidak Setuju

2.2 Penelitian Terkait

Pada sub bab kali ini, penulis mencoba memberikan gambaran penelitian yang berkaitan dengan penelitian yang akan penulis kembangkan. Penelitian terkait akan disajikan dalam bentuk tabel penelitian dan posisi penelitian seperti pada tabel di bawah ini:

2.2.1 Tabel Penelitian

Tabel 1.1 Tabel penelitian terkait

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
1	Yosef Koko Kurniawan, Yetli Oslan, Harianto Kristanto, 2013	Implementasi REST - API Untuk Portal Akademik UKDW Berbasis Android	Portal Akademik	Civitas Akademik	Aplikasi Portal UKDW Berbasis Android Yang Berjalan Pada Sistem Operasi Android
2	Muhammad Faisal Fahat, Bayu Priyambadha, Fajar Pradana, 2018	Pengembangan Aplikasi Manajemen Proyek Perangkat Lunak Berbasis <i>Scrum</i> Studi Kasus CV. Nusantara Media Mandiri	Manajemen Proyek	Tim Manajemen Proyek	Aplikasi Manajemen Proyek Berbasis Android

3	M Soleh Hapudin, Jani Sujatmoko, 2013	Perancangan Program Project Management Berbasis Android Pada Sistem PMB	Program Project Management	Civitas Akademik	Program Project Management Berbasis Android
4	Ricky Resky Ananda, 2018	Pengembangan Media Pembelajaran <i>Scrum</i> Interaktif Berbasis Aplikasi Android	Media Pembelajaran <i>Scrum</i>	Mahasiswa	Aplikasi <i>Scrum</i> Learning Berbasis Android
5	Ardith Lutfiawan, 2020	Rancang Bangun Aplikasi <i>Link-Match</i> Kampus STT-Nf Modul Dosen Berbasis <i>Mobile</i> Menggunakan <i>Flutter</i>	Aplikasi <i>Link-</i> <i>Match</i> berbasis Android	Dosen STT Nurul Fikri	

2.2.2 Posisi Penelitian

Tabel 1.2 Posisi penelitian

No.	Pengarang	Berbasis <i>Mobile</i>	Manajemen Proyek	<i>Scrum</i>	<i>Flutter</i> <i>Framework</i>
1	<u>Yosef Koko</u> <u>Kurniawan, Yetli</u> <u>Oslan, Harianto</u> <u>Kristanto (2013)</u>	∨			
2	<u>Muhammad Faisal</u> <u>Fahat, Bayu</u> <u>Priyambadha, Fajar</u> <u>Pradana (2018)</u>		∨	∨	
3	<u>M Soleh Hapudin,</u> <u>Jani Sujatmoko</u> <u>(2013)</u>	∨	∨		
4	<u>Ricky Resky</u> <u>Ananda (2018)</u>	∨		∨	

5	<u>Ardith Lutfiawan (2020)</u>	∨	∨	∨	∨
---	------------------------------------	---	---	---	---

Penelitian-penelitian tersebut memiliki beberapa bahasan serupa dengan yang penulis kembangkan terutama pada penelitian oleh Ricky Resky Ananda mengenai “Pengembangan Media Pembelajaran *Scrum* Interaktif Berbasis Aplikasi Android” dimana penelitian ini dimaksudkan untuk menunjang pembelajaran mengenai *scrum* dengan metode perancangan berbasis *scrum*, sedangkan penelitian yang penulis lakukan dimaksudkan untuk menunjang manajemen proyek dengan metode perancangan berbasis *scrum*.

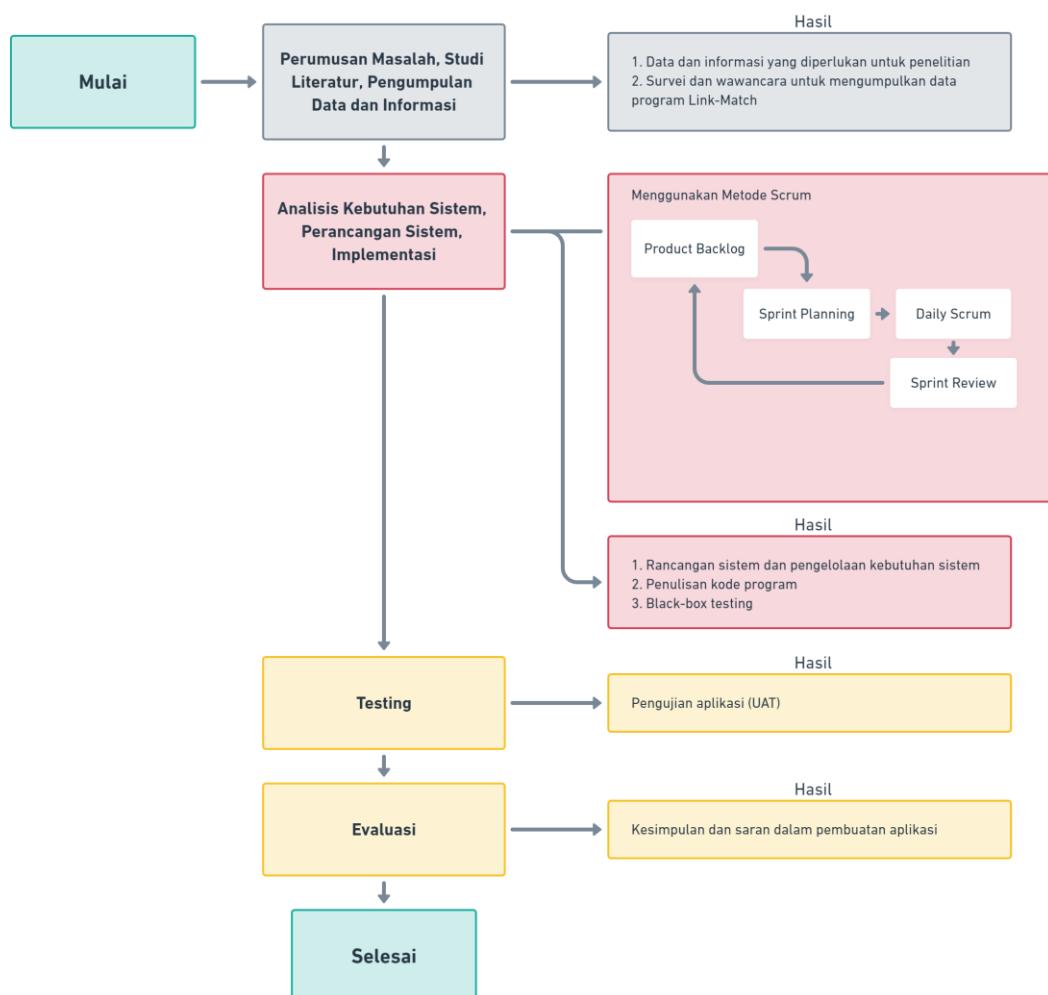
BAB III

METODE PENELITIAN

Pada bab ini berisi tahapan-tahapan yang akan dilakukan mulai dari perumusan masalah sampai dengan kesimpulan. Tahapan-tahapan tersebut akan membentuk alur penelitian yang sistematis, digunakan sebagai pedoman agar hasil yang dicapai tidak menyimpang dari tujuan awal penelitian ini.

3.1 Tahapan Penelitian

Tahapan penelitian mencangkap tahapan– tahapan yang akan dilakukan dalam penelitian ini, adapun tahapannya sebagai berikut:



Gambar 3.1 Tahapan penelitian

3.1.1 Perumusan Masalah

Pada tahap ini dilakukan peninjauan pada program *link-match* yang akan diteliti untuk mengetahui lebih dalam terhadap masalah yang ingin diselesaikan. Tahap ini merupakan tahapan pertama yang diperlukan untuk mendefinisikan sistem yang ingin dicapai.

3.1.2 Studi Literatur

Studi Literatur dilakukan dengan tujuan untuk mengetahui metode apa yang akan digunakan untuk menjawab permasalahan yang akan diteliti, serta untuk memperdalam ilmu terkait dari referensi-referensi yang menjadi dasar dari penelitian ini.

3.1.3 Pengumpulan Data dan Informasi

Tahap selanjutnya yaitu pengumpulan data dan informasi, yang mana data dan informasi yang didapat akan memberikan gambaran dari sistem yang berjalan saat ini sehingga akan menjadi dasar kuat bagi peneliti.

Adapun pengumpulan data dan informasi pada penelitian ini dilakukan dengan 2 metode, yaitu:

a. Wawancara

Wawancara dilakukan dengan cara tanya jawab kepada dosen dan *stakeholder* terkait. Wawancara dapat dilakukan secara langsung maupun tidak langsung dengan tujuan mendapatkan informasi yang dibutuhkan dalam perancangan aplikasi *link-match* di STT-NF berbasis *mobile*.

b. Survei

Survei dilakukan dengan cara memberi pertanyaan atau pernyataan tertulis kepada responden untuk dijawab berkaitan dengan perancangan aplikasi *link-match* di STT-NF berbasis *mobile*.

3.1.4 Analisis Kebutuhan Sistem

Pada tahap ini dilakukan analisa terhadap kebutuhan pada program *link-match* di STT-NF khususnya pada modul dosen dengan tujuan mengetahui permasalahan

yang ada sehingga dapat diberikan solusi dengan menerapkan metode *scrum* guna menjawab permasalahan tersebut.

3.1.5 Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem untuk mempermudah dalam proses mengembangkan perangkat lunak seperti mengelola kebutuhan sistem, analisa, desain, dan implementasi pada kode program yang nantinya digunakan pada perancangan aplikasi *link-match*. Pada tahapan ini pula dilakukan pemodelan dengan menggunakan *Unified Modelling Language (UML)* untuk mendapatkan model aplikasi yang akan dibuat.

3.1.6 Implementasi & Testing

Setelah perancangan sistem dilakukan, tahapan berikutnya yaitu implementasi pada kode program dan selanjutnya dapat digunakan oleh pengguna dengan memperhatikan antarmuka dan fitur yang dibutuhkan dalam program *link-match* di STT-NF agar program mudah dalam digunakan dan sesuai dengan kebutuhan sistem. Pada tahapan ini dilakukan pengujian aplikasi untuk mengetahui fungsionalitas dari aplikasi yang berjalan menggunakan *black-box testing* dan tahapan selanjutnya dilakukan pengujian *User Acceptance Test (UAT)* untuk melihat kepuasan pengguna dari aplikasi *link-match*.

3.1.7 Evaluasi

Dari hasil pengujian dilakukan evaluasi berupa kritik dan saran dalam pengembangan aplikasi *link-match* serta penarikan kesimpulan apakah aplikasi yang dibuat dapat menjawab permasalahan yang terjadi.

3.2 Rancangan Penelitian

3.2.1 Jenis Penelitian

Berdasarkan jenis data dan analisisnya penelitian ini merupakan penelitian gabungan kualitatif dan kuantitatif yang datanya terdiri dari data berupa hasil wawancara dan survei terhadap perancangan aplikasi *link-match* berbasis *mobile* di STT-NF.

3.2.2 Lingkungan Pengembangan

Penelitian ini dilakukan di kampus STT Terpadu Nurul Fikri dimana terdapat 2 lokasi penelitian, 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 Alat dan Bahan

Adapun alat dan bahan yang digunakan dalam penelitian ini antara lain:

1. Laptop Lenovo YOGA Slim 7 14ARE05
 - *Prosesor* : AMD Ryzen 5 4500U CPU @2.38 GHz
 - *RAM* : 16.00 GB
 - *System Type* : 64-bit *Operating System*
2. Windows 10 Home
3. Microsoft Office
4. Figma
5. Android Studio
6. *Flutter Framework*
7. StarUML
8. Postman
9. Google Chrome

BAB IV

PERANCANGAN SISTEM

Pada bab ini akan dijelaskan mengenai analisis dan rancangan dari sistem yang akan dibangun berdasarkan pengamatan. Selanjutnya, tahapan-tahapan yang akan dilakukan antara lain yaitu menganalisa sistem, membuat pemodelan sistem dengan *Unified Modelling Language (UML)*, merancang *user story* yang kemudian akan dikumpulkan menjadi *product backlog*, dan dilanjutkan dengan perancangan antar muka aplikasi.

4.1 Analisis Sistem

Pada tahap analisis sistem ini penulis melakukan analisa terhadap kebutuhan sistem *link-match* khususnya pada modul dosen untuk mengetahui hal-hal apa saja yang dibutuhkan berdasarkan pengamatan yang penulis lakukan.

4.1.1 *End-user*

Berdasarkan hasil wawancara yang tercantum pada lampiran, penulis mendapatkan informasi bahwa terdapat beberapa peranan yang biasa dilakukan pada program *link and match*. Peran-peran tersebut selanjutnya akan dijabarkan pada tabel *end-user* berikut:

Tabel 4.1 *End-user*

Nama	Peran
<i>Product owner</i>	Memberikan <i>project</i> berupa pembuatan sebuah sistem aplikasi
Dosen	Memberikan penilaian kepada mahasiswa dan tim serta mengawasi <i>project</i>
<i>Scrum master</i>	Memberikan arahan <i>scrum</i> kepada mahasiswa dan membuat <i>project</i> dan tim

4.1.2 Kebutuhan Sistem

Setelah mengetahui pengguna aplikasi *link-match* maka selanjutnya penulis dapat membuat kebutuhan sistem berdasarkan hasil pengamatan sebelumnya. Berikut adalah kebutuhan sistem yang diperlukan pada modul dosen berbasis Android:

1. *User management*, memungkinkan *end-user* untuk mengelola *user* seperti melihat *profile* dan melihat *user* mahasiswa.

2. *Project management*, memungkinkan *end-user* untuk melihat informasi *project*, dan melakukan operasi CRUD pada *user story*.
3. *Team management*, memungkinkan *end-user* untuk mengelola tim seperti melihat daftar tim, dan melihat *detail* tim.

4.1.3 *Product Backlog*

Selanjutnya pada penelitian ini akan dirancang *product backlog* dengan tujuan mempermudah suatu pekerjaan perancangan aplikasi karena membagi kebutuhan sistem yang besar menjadi poin-poin penyelesaian yang lebih kecil dan terstruktur. Berikut adalah *product backlog* yang diperlukan pada aplikasi *link-match* modul dosen berbasis Android:

Tabel 4.2 Product backlog

No,	User story	Fitur	Developer
1	Sebagai calon pengguna, saya diharuskan melakukan <i>login</i> dengan mengetikkan <i>username</i> berupa alamat <i>email</i> dan <i>password</i> minimal 8 karakter.	<i>Login user</i>	Ardith Lutfiawan
2	Sebagai pengguna, saya dapat melihat <i>profile</i> saya yang berisi nama lengkap, foto, <i>email</i> , <i>role user</i> , nomor hp dan pilihan <i>logout</i> .	<i>View profile</i>	Ardith Lutfiawan
3	Sebagai pengguna, saya dapat melakukan <i>logout</i> dan dilakukan dengan memilih tombol keluar pada menu <i>profile</i> .	<i>Logout user</i>	Ardith Lutfiawan
4	Sebagai user aplikasi, saya dapat melihat daftar menu yang dapat saya akses sesuai role saya	Halaman <i>dashboard</i>	Ardith Lutfiawan

5	Sebagai pengguna, saya dapat melihat <i>project</i> apa saja yang sedang berlangsung pada menu <i>project</i> dan melihat <i>detail project</i> dari <i>project</i> yang dipilih.	<i>View project</i>	Ardith Lutfiawan
6	Sebagai seorang dosen dan <i>scrum master</i> , saya dapat melihat daftar <i>sprint</i> pada menu <i>sprint</i> dan melihat detail <i>sprint</i> dari <i>sprint</i> yang dipilih.	<i>View sprint</i>	Ardith Lutfiawan
7	Sebagai seorang <i>scrum master</i> , saya dapat membuat <i>sprint</i> baru pada menu <i>sprint</i> .	<i>Add sprint</i>	Ardith Lutfiawan
8	Sebagai seorang <i>scrum master</i> , saya dapat mengupdate status <i>sprint</i> menjadi “berlangsung” pada halaman <i>detail sprint</i> yang dipilih.	Memulai <i>sprint</i>	Ardith Lutfiawan
9	Sebagai seorang <i>scrum master</i> , saya dapat meng-update status <i>sprint</i> menjadi “berakhir” pada halaman <i>detail sprint</i> yang dipilih.	Mengakhiri <i>sprint</i>	Ardith Lutfiawan
10	Sebagai seorang <i>scrum master</i> , saya dapat merubah <i>sprint</i> yang dipilih.	<i>Edit sprint</i>	Ardith Lutfiawan
11	Sebagai seorang <i>scrum master</i> , saya dapat menghapus <i>sprint</i> yang dipilih.	<i>Delete sprint</i>	Ardith Lutfiawan

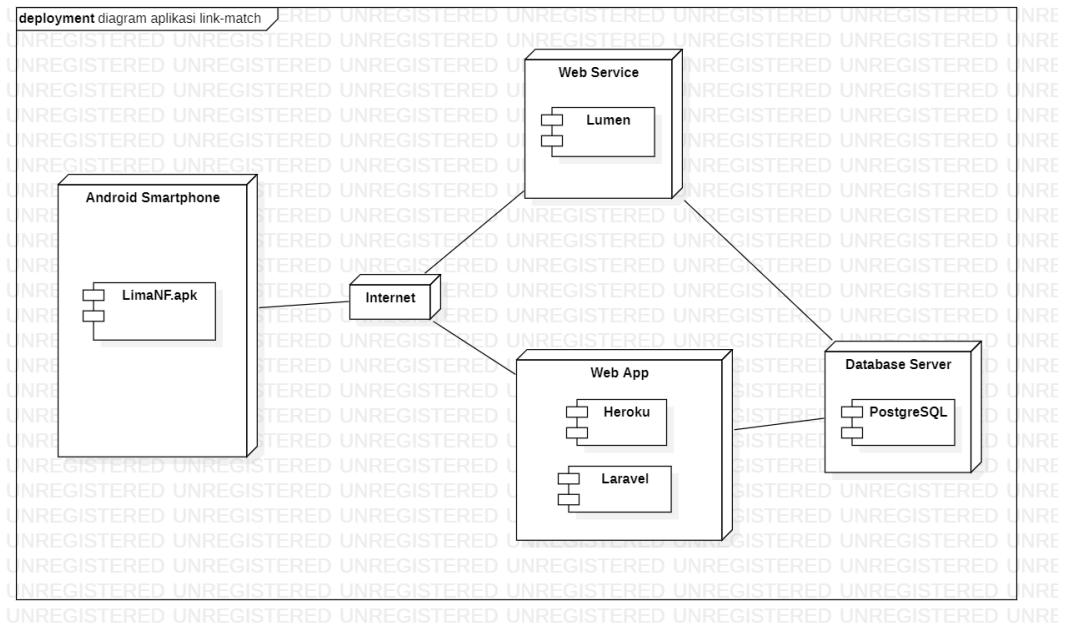
12	Sebagai seorang dosen dan <i>scrum master</i> , saya dapat melihat daftar tim dan <i>detail</i> tim tersebut.	<i>View</i> tim	Ardith Lutfiawan
13	Sebagai seorang dosen dan <i>scrum master</i> , saya dapat melihat daftar mahasiswa.	<i>View</i> mahasiswa	Ardith Lutfiawan
14	Sebagai seorang dosen, saya dapat melihat penilaian tim dan mahasiswa.	<i>View</i> nilai	Ardith Lutfiawan

4.2 Perancangan Sistem

Pada tahap perancangan sistem ini penulis memberikan gambaran sistem aplikasi dan desain antarmuka sebagai tahapan awal sebelum memulai proses penulisan kode program.

4.2.1 Deployment Diagram

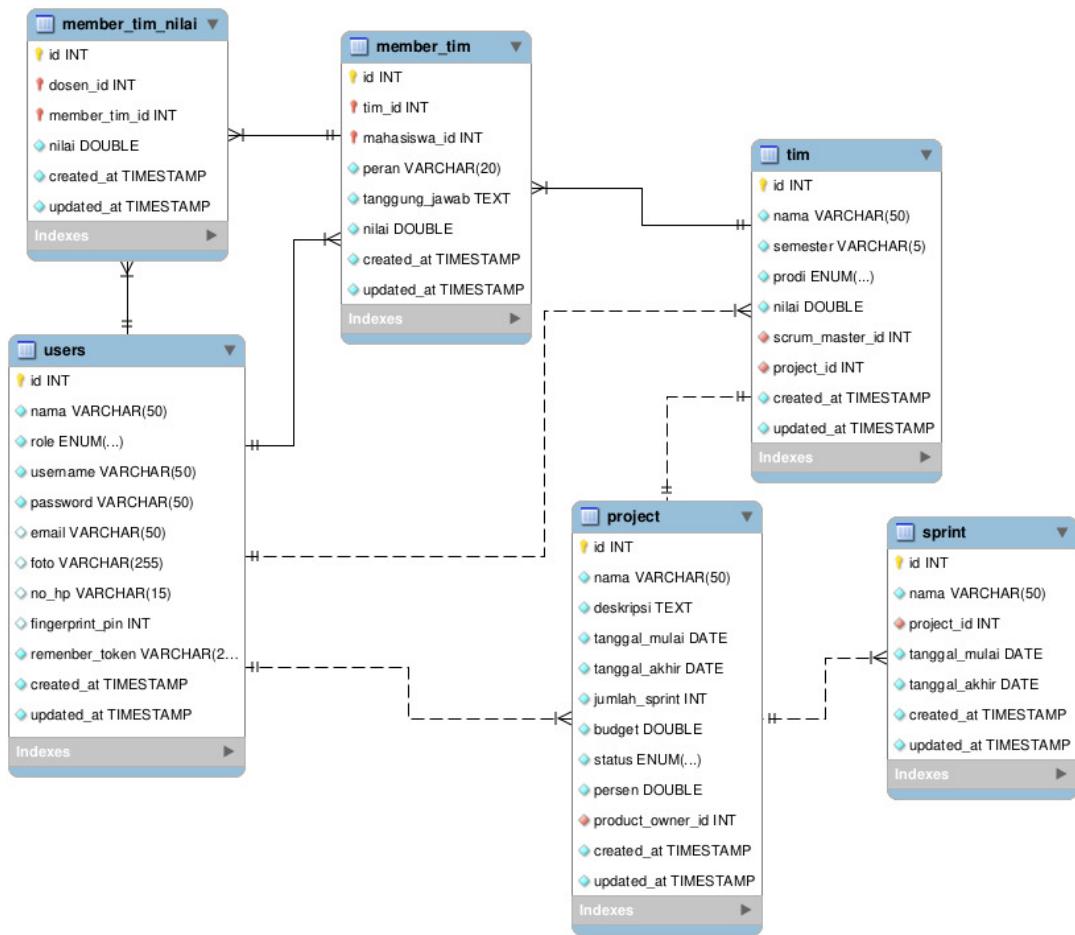
Diagram ini digunakan untuk menggambarkan hubungan antara komponen *hardware*, *software* dan metode yang diterapkan. Diagram ini biasanya terdiri dari beberapa *nodes*, setiap nodes mewakili perangkat keras yang ada. Berikut ini adalah gambar *deployment diagram* aplikasi *mobile link-match* modul dosen:



Gambar 4.1 Deployment diagram

4.2.2 Entity Relationship Diagram

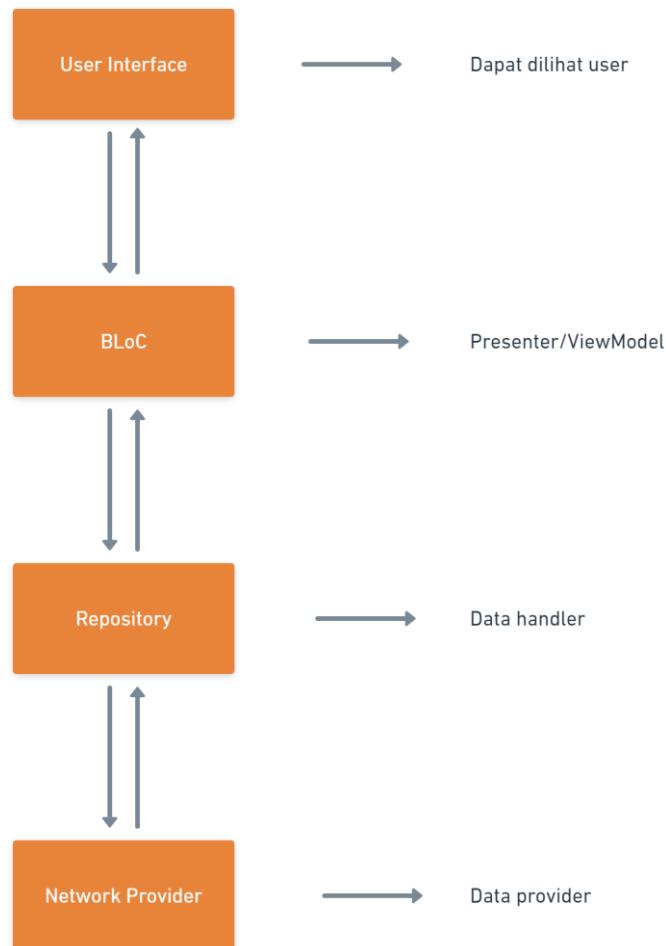
Pada rancangan *Entity Relationship Diagram* ini penulis mengambil gambaran *database* daripada *web service*, dimana pada *database* ini berisi tabel-tabel yang akan penulis gunakan untuk mengkonsumsi API dalam pertukaran data pada aplikasi *mobile*.



Gambar 4.2 *Entity relationship diagram*

4.2.3 State Management

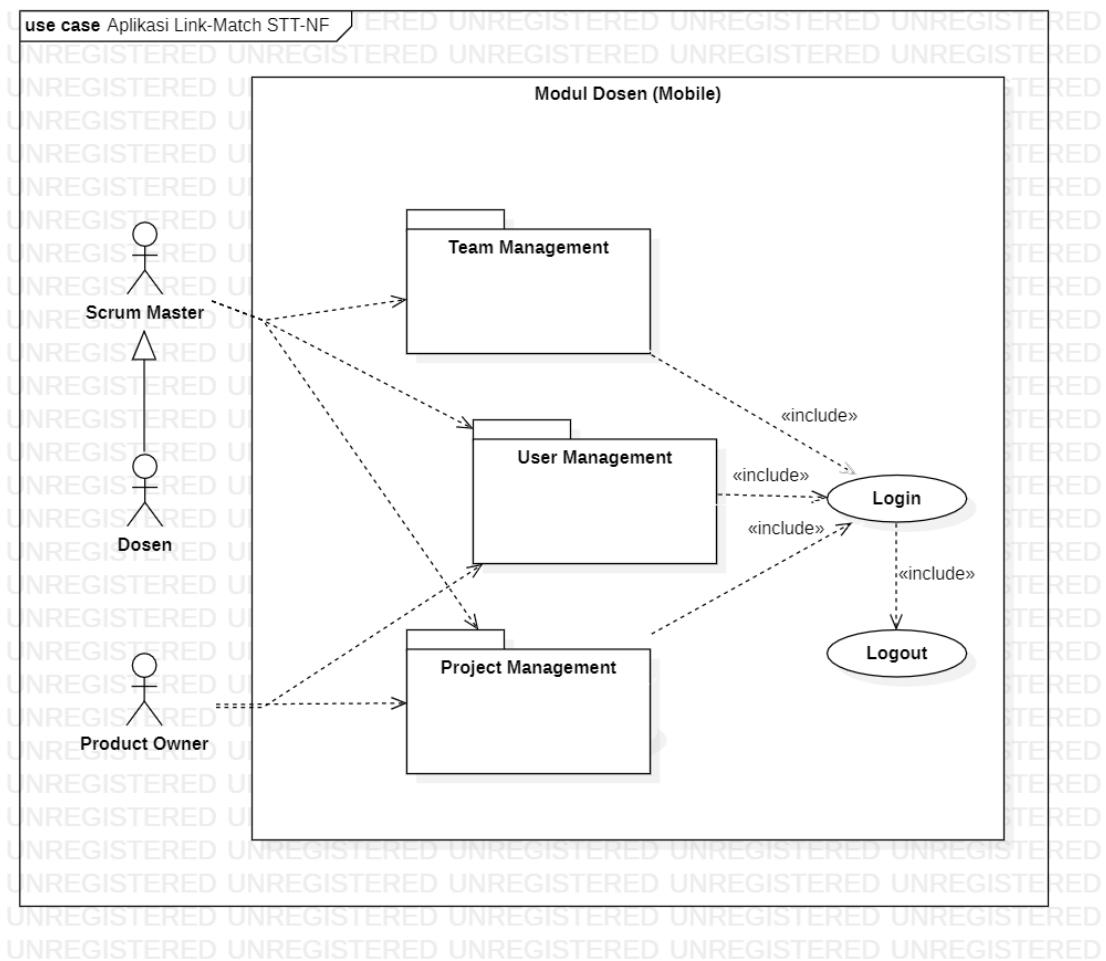
Bagian ini berisi rencana penggunaan *state management* menggunakan *BLoC* (*Business Logic Component*) yang akan dibuat yaitu sebagai berikut:



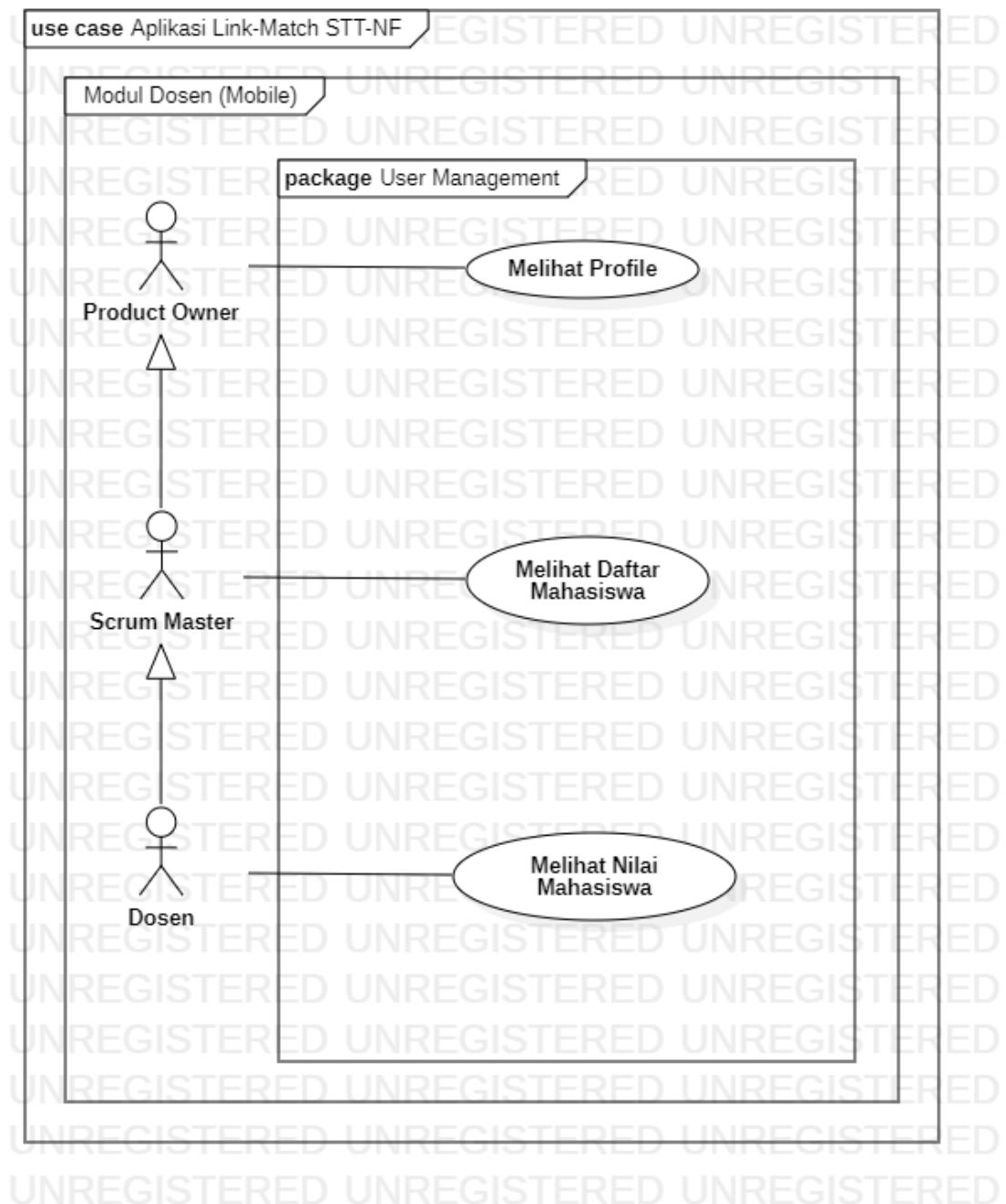
Gambar 4.3 Rancangan arsitektur BLoC

4.2.4 Use Case Diagram

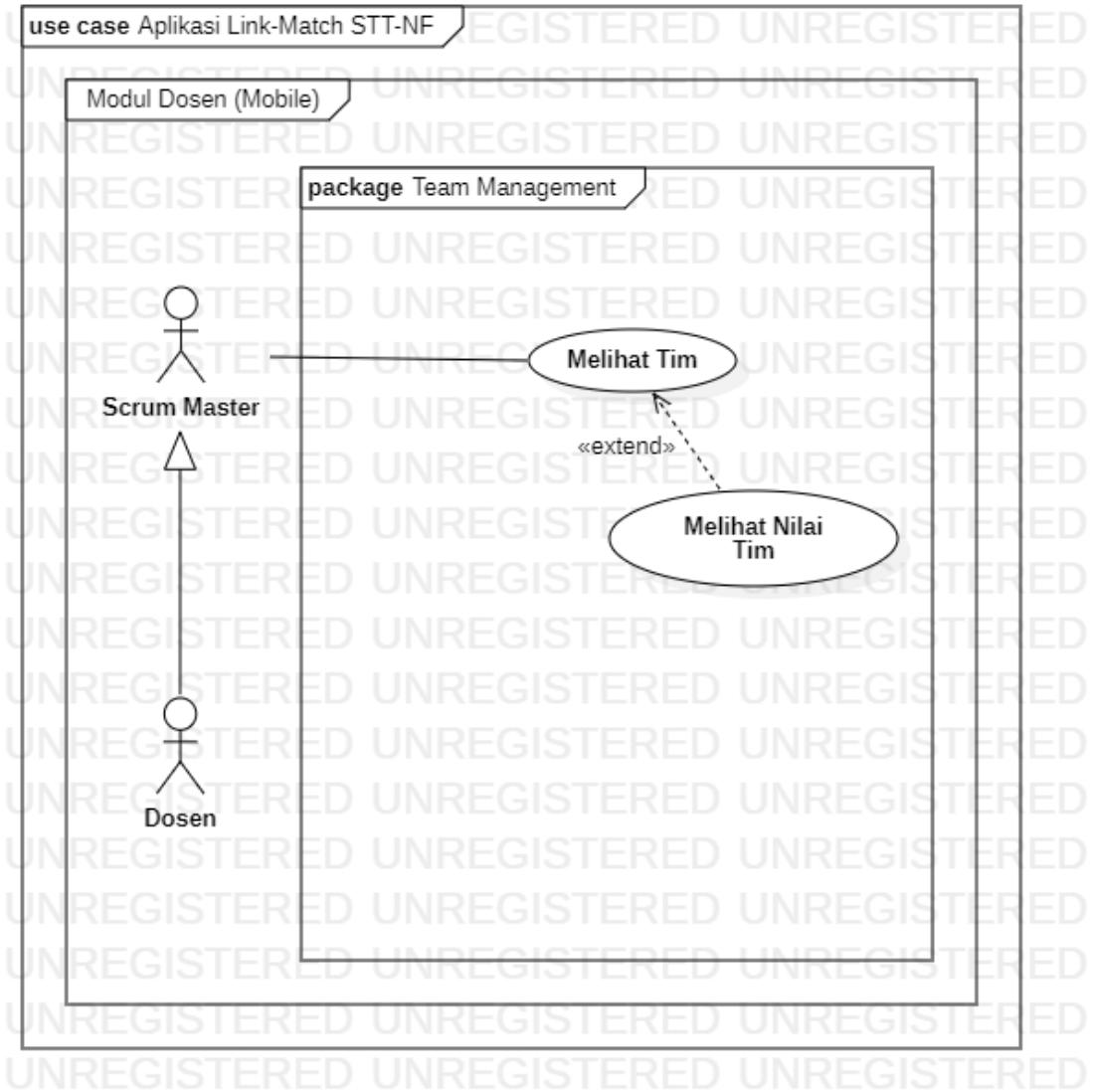
Tahapan dalam perancangan sistem ini menggambarkan hubungan antara *end-user* dengan sistem melalui *use case diagram*, dimana pada diagram tersebut dapat menggambarkan secara umum aplikasi *link-match* berbasis mobile yang akan dibuat. Berikut ini adalah gambar daripada *use case diagram*:



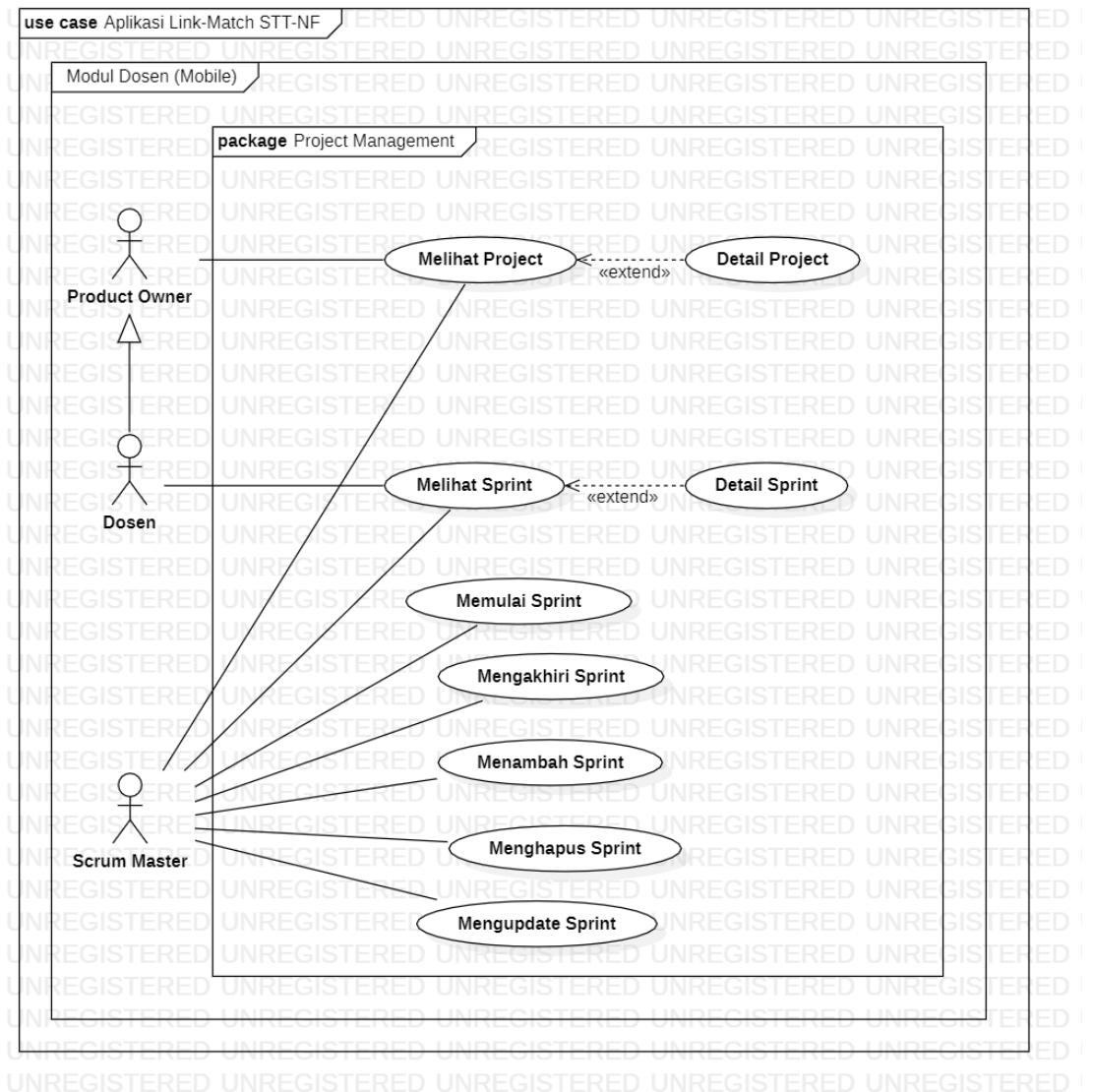
Gambar 4.4 Use case diagram



Gambar 4.5 Use case diagram user management



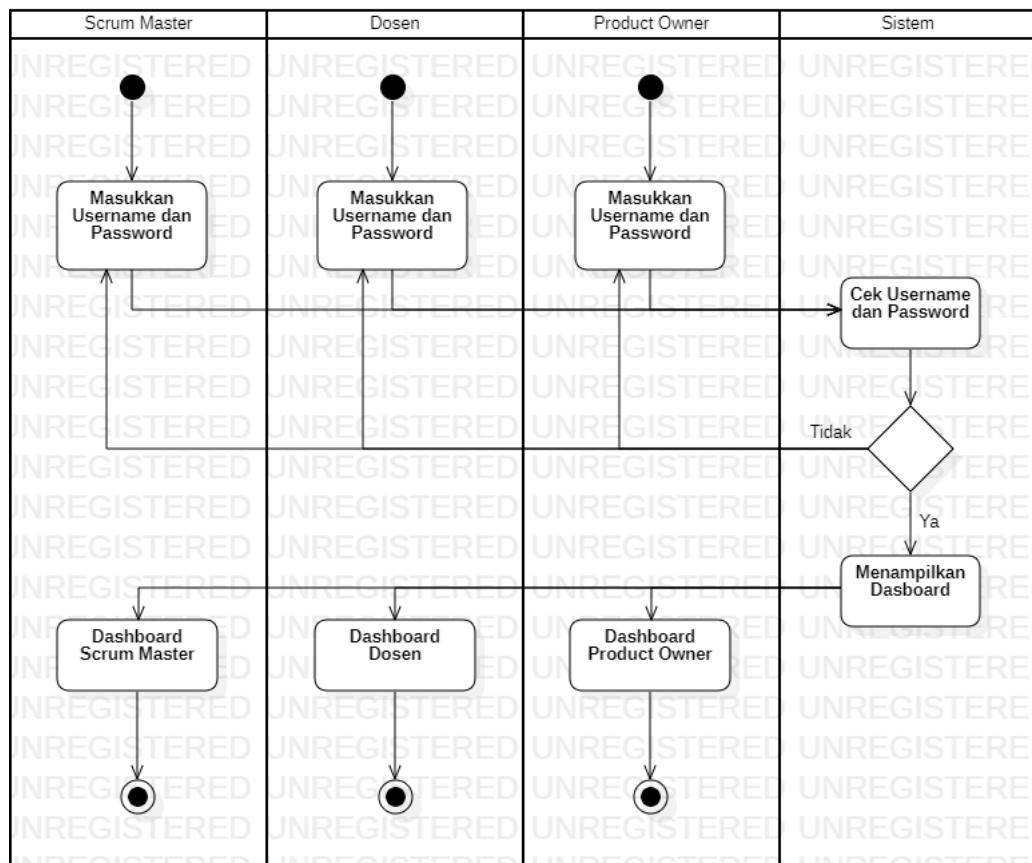
Gambar 4.6 Use case diagram team management



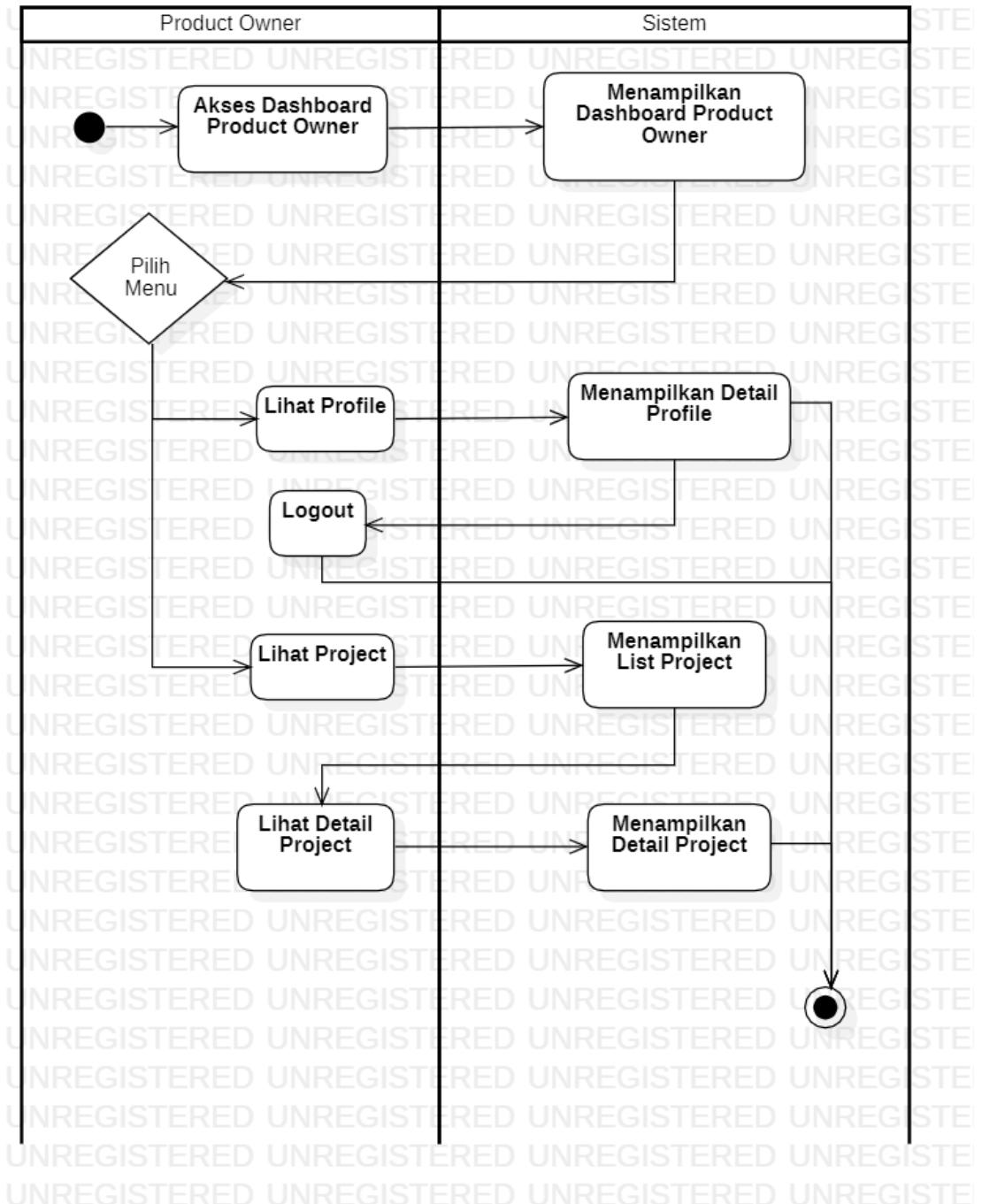
Gambar 4.7 Use case project management

4.2.5 Activity Diagram

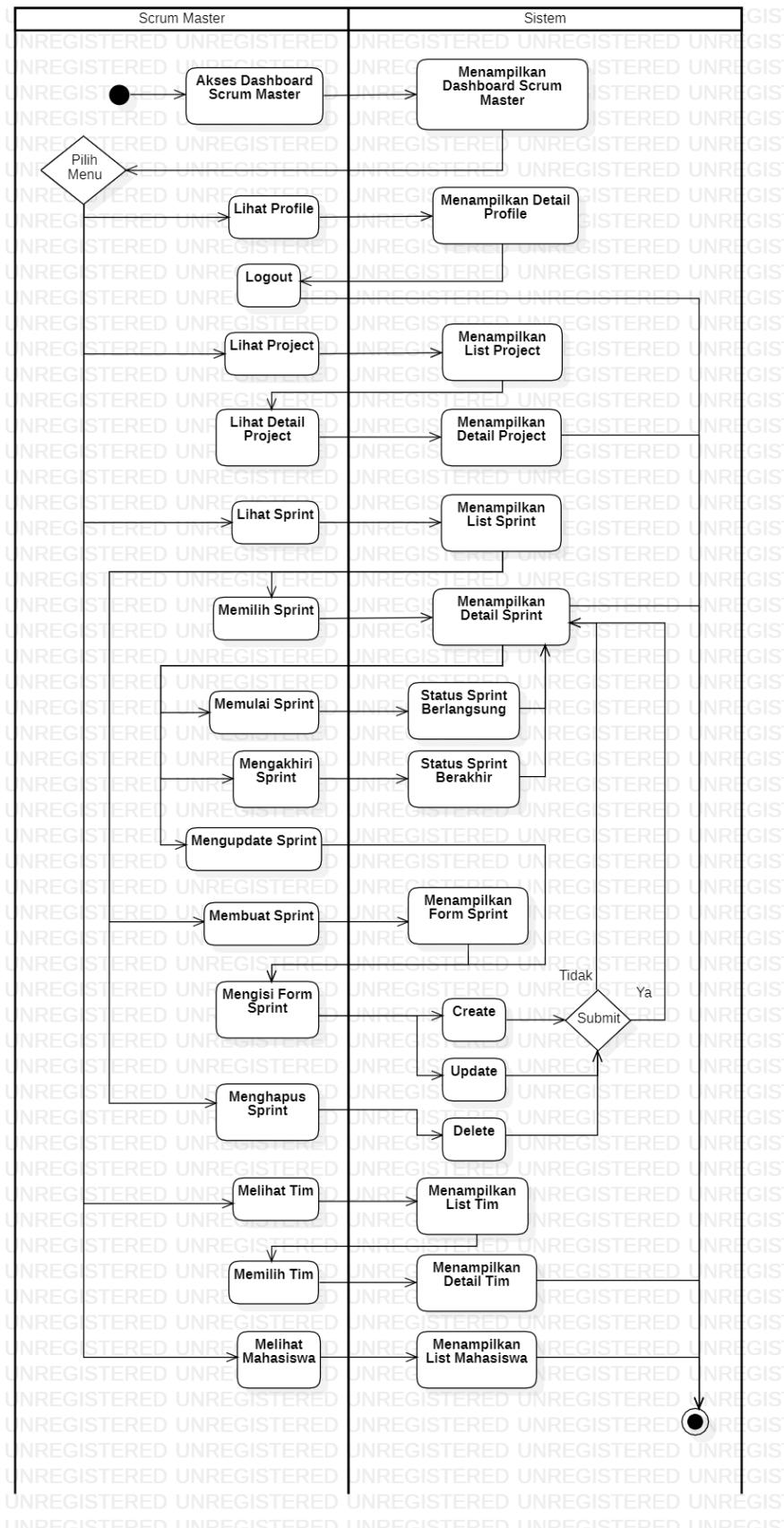
Berikut ini adalah alur sistem pada modul dosen yang dijelaskan dalam bentuk *activity diagram* pada gambar berikut:



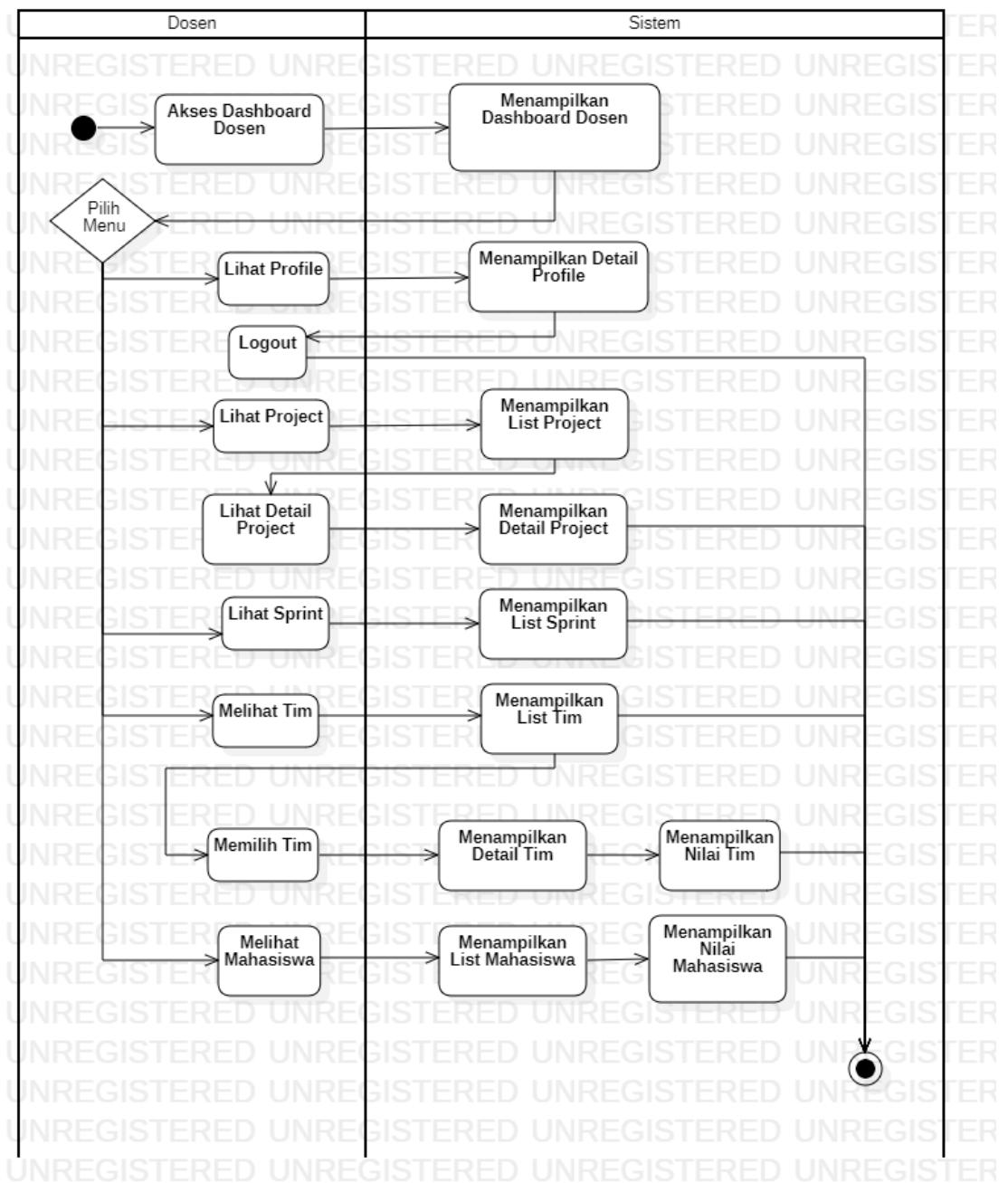
Gambar 4.8 Activity diagram



Gambar 4.9 Activity diagram role product owner



Gambar 4.10 Activity diagram role scrum master



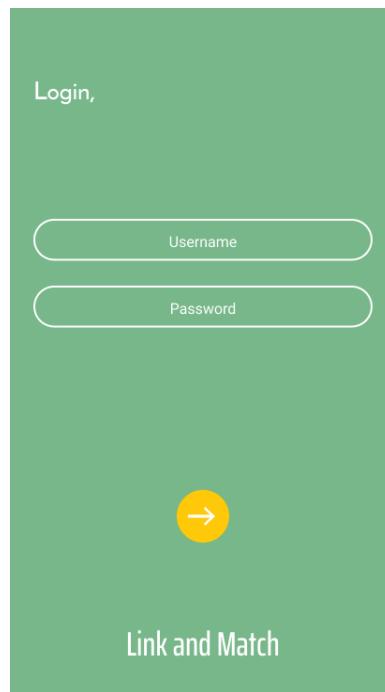
Gambar 4.11 *Activity diagram role dosen*

4.2.6 User Interface

User Interface memberikan gambaran aplikasi yang akan dibangun dan membantu dalam implementasi ke dalam kode program. Gambar-gambar berikut

dibuat berdasarkan fitur apa saja yang terdapat di aplikasi link-match sesuai dengan kebutuhan sistem.

A. Login



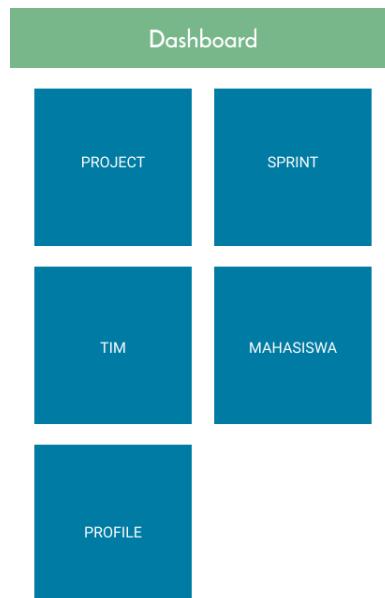
Gambar 4.12 User interface login

B. *View profile dan logout*



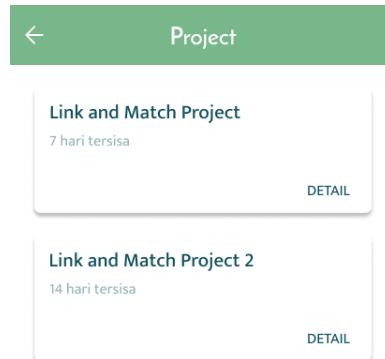
Gambar 4.13 *User interface view profile dan logout*

C. *Dashboard*

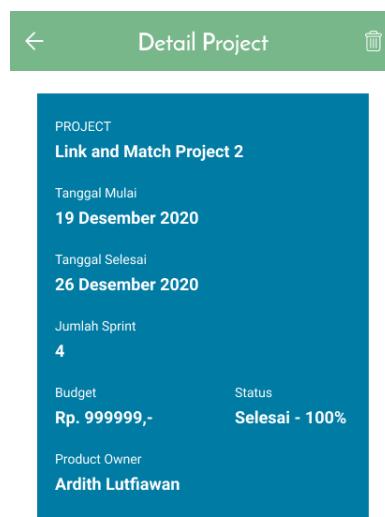


Gambar 4.14 *User interface dashboard*

D. *View project*

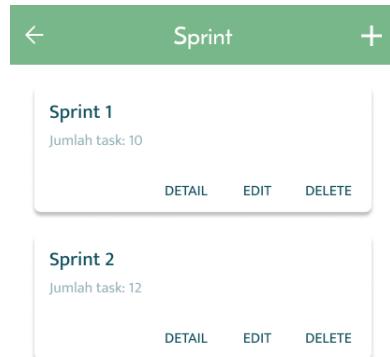


Gambar 4.15 User interface list project

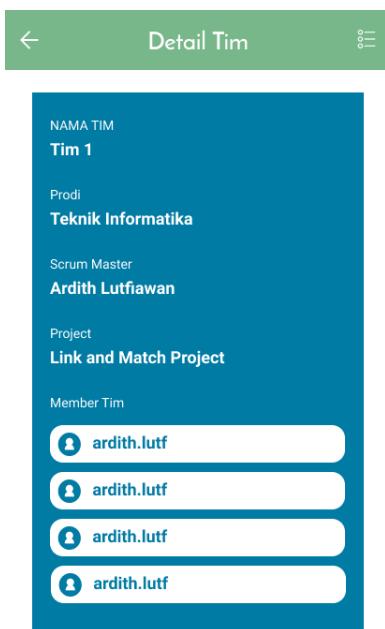


Gambar 4.16 User interface detail project

E. View sprint



Gambar 4.17 User interface list sprint



Gambar 4.18 User interface detail sprint

F. Add sprint

Buat Sprint

Project

Nama Sprint

Tanggal Mulai

Tanggal Selesai

→

Gambar 4.19 User interface add sprint

G. Memulai sprint

Detail Sprint

NAMA SPRINT
Sprint 1

PROJECT
Link and Match Project 2

Tanggal Mulai
19 Desember 2020

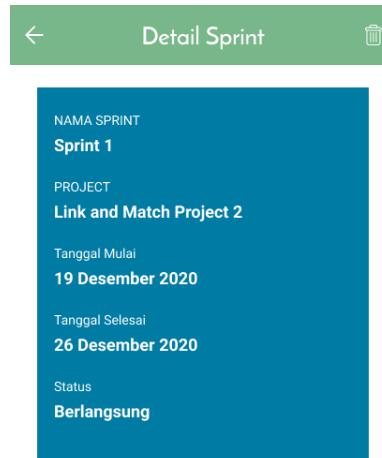
Tanggal Selesai
26 Desember 2020

Status
Belum Berlangsung

Mulai Sprint

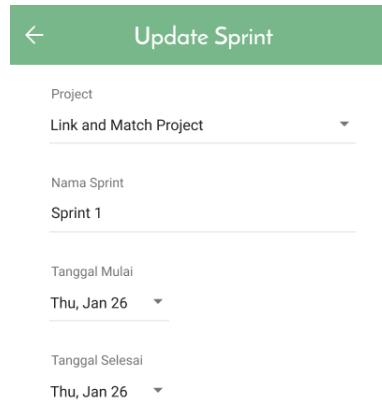
Gambar 4.20 User interface memulai sprint

H. Mengakhiri sprint



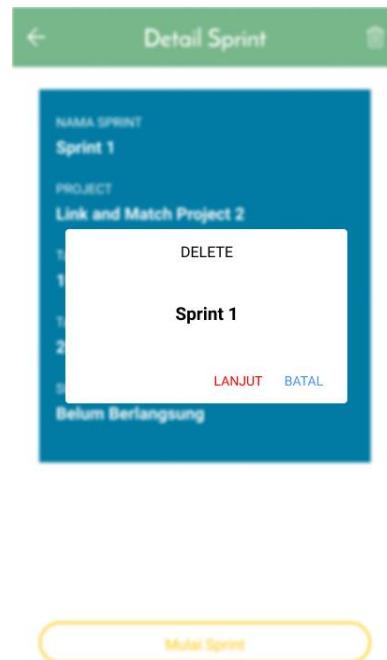
Gambar 4.21 User interface mengakhiri sprint

I. Edit sprint



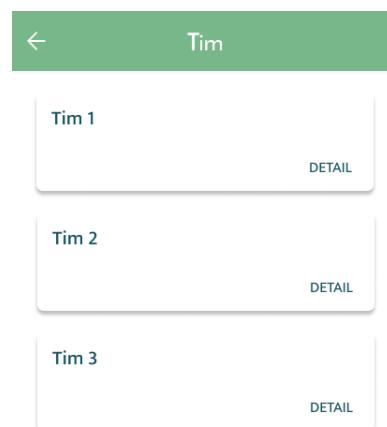
Gambar 4.22 User interface edit sprint

J. Delete sprint



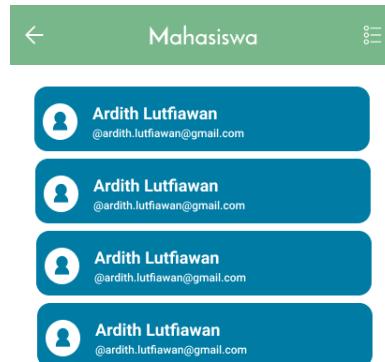
Gambar 4.23 User interface delete sprint

K. View tim



Gambar 4.24 User interface view tim

L. View mahasiswa



Gambar 4.25 User interface view mahasiswa

M. View nilai tim

Nilai Tim	
TI 2019	
Nama Tim	Nilai
Tim Siap	A
Nama Tim	Nilai
Tim Siap	A
Nama Tim	Nilai
Tim Siap	A
SI 2019	
Nama Tim	Nilai
Tim Siap	A
Nama Tim	Nilai
Tim Siap	A
Nama Tim	Nilai
Tim Siap	A

Gambar 4.26 User interface view nilai tim

N. View nilai mahasiswa

Nilai Mahasiswa		
TI 2019		
Nama Lengkap	Role	Nilai
Ardith Lutfiawan	Frontend	A
SI 2019		
Nama Lengkap	Role	Nilai
Ardith Lutfiawan	Frontend	A
SI 2019		
Nama Lengkap	Role	Nilai
Ardith Lutfiawan	Frontend	A
SI 2019		
Nama Lengkap	Role	Nilai
Ardith Lutfiawan	Frontend	A

Gambar 4.27 User interface view nilai mahasiswa

4.3 Rencana Pengujian

Penelitian ini dilengkapi dengan pengujian menggunakan metode pengujian *blackbox-testing* serta *user acceptance testing* atau disingkat *UAT* dengan harapan dapat mengetahui lebih rinci permasalahan yang mungkin dapat terjadi.

4.3.1 Black-box Testing

Pengujian dengan *black-box testing* berfokus pada fungsionalitas, menguji aplikasi apakah dapat berjalan dengan fungsionalitas yang diharapkan dengan melewati proses pengujian pada tiap prosesnya.

Tabel 4.3 Black-box testing

No.	Pengujian	User Story	Hasil yang diharapkan	Hasil
1	<i>Login user</i>	Halaman <i>login</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	Berhasil / Gagal
		<i>Form login</i> berisi <i>username</i> dan <i>password</i>	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	

		<i>Submit button: login</i>	<i>User dapat mengklik submit button untuk proses validasi</i>	
		<i>Splash-screen dan halaman dashboard</i>	<i>User berhasil login, menampilkan splash-screen dan halaman dashboard</i>	
		<i>Pesan validasi error</i>	<i>User dapat melihat pesan error jika form tidak sesuai atau validasi gagal</i>	
2	<i>View profile</i>	<i>Menu profile</i>	<i>User dapat memilih menu profile pada halaman dashboard</i>	Berhasil / Gagal
		<i>Halaman profile</i>	<i>User dapat menampilkan halaman profile</i>	
3	<i>Logout user</i>	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	Berhasil / Gagal
		<i>Halaman login</i>	<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	
4	<i>Halaman dashboard</i>	<i>Halaman dashboard</i>	<i>User dapat menampilkan halaman dashboard sesuai dengan role tertentu</i>	Berhasil / Gagal
5	<i>View project</i>	<i>Menu project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	Berhasil / Gagal
		<i>Halaman project</i>	<i>User dapat menampilkan halaman project</i>	
		<i>Halaman detail project</i>	<i>User dapat menampilkan halaman project yang dipilih</i>	
6	<i>View sprint</i>	<i>Menu sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	Berhasil / Gagal
		<i>Halaman sprint</i>	<i>User dapat menampilkan halaman sprint</i>	
		<i>Halaman detail sprint</i>	<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	

		<i>Add button: tambah sprint</i>	<i>User dapat mengklik add button untuk menampilkan halaman tambah sprint</i>	
		<i>Delete button: hapus sprint</i>	<i>User dapat mengklik delete button untuk menampilkan pop-up hapus sprint</i>	
7	<i>Add sprint</i>	Halaman tambah sprint	<i>User dapat menampilkan form tambah sprint</i>	Berhasil / Gagal
		Pesan form error	<i>User dapat melihat pesan error jika form tidak sesuai</i>	
		<i>Submit button: tambah sprint</i>	<i>User dapat mengklik submit button untuk proses tambah sprint</i>	
8	Memulai sprint	<i>Submit button: mulai sprint</i>	<i>User dapat mengklik submit button untuk proses mulai sprint</i>	Berhasil / Gagal
9	Mengakhiri sprint	<i>Submit button: akhiri sprint</i>	<i>User dapat mengklik submit button untuk proses akhiri sprint</i>	Berhasil / Gagal
10	<i>Edit sprint</i>	Halaman edit sprint	<i>User dapat menampilkan form edit sprint</i>	Berhasil / Gagal
		Pesan form error	<i>User dapat melihat pesan error jika form tidak sesuai</i>	
		<i>Submit button: edit sprint</i>	<i>User dapat mengklik submit button untuk proses edit sprint</i>	
11	<i>Delete sprint</i>	<i>Pop-up: hapus sprint</i>	<i>User dapat melihat pop-up hapus sprint</i>	Berhasil / Gagal
		<i>Submit button: hapus sprint</i>	<i>User dapat mengklik submit button untuk proses hapus sprint</i>	
		<i>Back button: batal hapus sprint</i>	<i>User dapat mengklik back button untuk membatalkan hapus sprint</i>	
12	<i>View tim</i>	Menu tim	<i>User dapat memilih menu tim pada halaman dashboard</i>	Berhasil / Gagal
		Halaman tim	<i>User dapat menampilkan halaman tim</i>	

		Halaman <i>detail</i> tim	<i>User</i> dapat menampilkan halaman tim yang dipilih	
		<i>List button: view</i> nilai tim	<i>User</i> dapat mengklik <i>list button</i> untuk menampilkan halaman nilai tim	
13	<i>View mahasiswa</i>	Menu mahasiswa	<i>User</i> dapat memilih menu mahasiswa pada halaman <i>dashboard</i>	Berhasil / Gagal
		Halaman mahasiswa	<i>User</i> dapat menampilkan halaman mahasiswa	
		<i>List button: view</i> nilai mahasiswa	<i>User</i> dapat mengklik <i>list button</i> untuk menampilkan halaman nilai mahasiswa	
14	<i>View nilai</i>	Halaman nilai tim	<i>User</i> dapat menampilkan halaman form nilai tim	Berhasil / Gagal
		Halaman nilai mahasiswa	<i>User</i> dapat menampilkan halaman nilai mahasiswa	

4.3.2 *User Acceptance Testing*

Penelitian yang dilakukan tidak terlepas dari keterlibatan *end-user* dalam menjadikan aplikasi yang telah dikembangkan dapat berjalan sesuai dengan kebutuhan. Adapun isi dari *user acceptance testing* ini dapat dilihat pada tabel-tabel berikut:

A. *User sebagai Product Owner*

Tabel 4.4 UAT user product owner

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	Berhasil / Gagal	
		<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>		

		<i>User dapat mengklik submit button untuk proses validasi</i>		
		<i>User berhasil login, menampilkan splash-screen dan halaman dashboard</i>		
		<i>User dapat melihat pesan error jika form tidak sesuai atau validasi gagal</i>		
2	<i>View profile</i>	<i>User dapat memilih menu profile pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman profile</i>		
3	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>		
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project dan profile user</i>	Berhasil / Gagal	
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman list project</i>		
		<i>User dapat menampilkan halaman project yang dipilih</i>		

B. User sebagai Dosen

Tabel 4.5 UAT user dosen

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	User dapat menampilkan halaman <i>login</i>	Berhasil / Gagal	
		User dapat mengetikkan <i>username</i> dan <i>password</i>		
		User dapat mengklik <i>submit button</i> untuk proses validasi		
		User berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>		
		User dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal		
2	<i>View profile</i>	User dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	Berhasil / Gagal	
		User dapat menampilkan halaman <i>profile</i>		
3	<i>Logout user</i>	User dapat mengklik <i>icon logout</i> pada halaman <i>dashboard</i> untuk proses <i>logout</i>	Berhasil / Gagal	
		User dapat menampilkan halaman <i>login</i> setelah proses <i>logout</i> selesai		
4	<i>Dashboard</i>	User dapat menampilkan halaman <i>dashboard</i> yaitu manajemen <i>project</i> , <i>sprint</i> , tim, mahasiswa dan <i>profile user</i>	Berhasil / Gagal	
5	<i>View project</i>	User dapat memilih menu <i>project</i> pada halaman <i>dashboard</i>	Berhasil / Gagal	
		User dapat menampilkan halaman <i>project</i>		

		<i>User dapat menampilkan halaman project yang dipilih</i>		
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman sprint</i>		
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>		
7	<i>View tim</i>	<i>User dapat memilih menu tim pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman tim</i>		
		<i>User dapat menampilkan halaman tim yang dipilih</i>		
		<i>User dapat mengklik list button untuk menampilkan halaman nilai tim</i>		
8	<i>View mahasiswa</i>	<i>User dapat memilih menu mahasiswa pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman mahasiswa</i>		
		<i>User dapat mengklik list button untuk menampilkan halaman nilai mahasiswa</i>		
9	<i>View nilai</i>	<i>User dapat menampilkan halaman nilai tim</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman nilai mahasiswa</i>		

C. User sebagai Scrum Master

Tabel 4.6 UAT user scrum master

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	Berhasil / Gagal	
		<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>		
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi		
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>		
		<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal		
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	Berhasil / Gagal	
		<i>User</i> dapat menampilkan halaman <i>profile</i>		
3	<i>Logout user</i>	<i>User</i> dapat mengklik <i>icon logout</i> pada halaman <i>dashboard</i> untuk proses <i>logout</i>	Berhasil / Gagal	
		<i>User</i> dapat menampilkan halaman <i>login</i> setelah proses <i>logout</i> selesai		
4	<i>Dashboard</i>	<i>User</i> dapat menampilkan halaman <i>dashboard</i> yaitu manajemen <i>project</i> , <i>sprint</i> , tim, mahasiswa dan <i>profile user</i>	Berhasil / Gagal	
5	<i>View project</i>	<i>User</i> dapat memilih menu <i>project</i> pada halaman <i>dashboard</i>	Berhasil / Gagal	
		<i>User</i> dapat menampilkan halaman <i>project</i>		

		<i>User</i> dapat menampilkan halaman <i>project</i> yang dipilih		
6	<i>View sprint</i>	<i>User</i> dapat memilih menu <i>sprint</i> pada halaman <i>dashboard</i>	Berhasil / Gagal	
		<i>User</i> dapat menampilkan halaman <i>sprint</i>		
		<i>User</i> dapat menampilkan halaman <i>detail sprint</i> yang dipilih		
		<i>User</i> dapat mengklik <i>add button</i> untuk menampilkan halaman tambah <i>sprint</i>		
		<i>User</i> dapat mengklik <i>delete button</i> untuk menampilkan <i>pop-up</i> hapus <i>sprint</i>		
7	<i>Add sprint</i>	<i>User</i> dapat menampilkan <i>form</i> tambah <i>sprint</i>	Berhasil / Gagal	
		<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai		
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses tambah <i>sprint</i>		
8	Memulai <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses mulai <i>sprint</i>	Berhasil / Gagal	
9	Mengakhiri <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses akhiri <i>sprint</i>	Berhasil / Gagal	
10	<i>Edit sprint</i>	<i>User</i> dapat menampilkan <i>form edit sprint</i>	Berhasil / Gagal	
		<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai		
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses <i>edit sprint</i>		

11	<i>Delete sprint</i>	<i>User dapat melihat pop-up hapus sprint</i>	Berhasil / Gagal	
		<i>User dapat mengklik submit button untuk proses hapus sprint</i>		
		<i>User dapat mengklik back button untuk membatalkan hapus sprint</i>		
12	<i>View tim</i>	<i>User dapat memilih menu tim pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman tim</i>		
		<i>User dapat menampilkan halaman tim yang dipilih</i>		
13	<i>View mahasiswa</i>	<i>User dapat memilih menu mahasiswa pada halaman dashboard</i>	Berhasil / Gagal	
		<i>User dapat menampilkan halaman mahasiswa</i>		

4.3.3 Kuesioner

Tahapan terakhir dalam bab ini adalah mengumpulkan kuesioner penggunaan aplikasi *link-match* dari setiap responden dengan sistem penilaian yang akan digunakan adalah skala likert yang memiliki bobot sebagai berikut:

Tabel 4.7 Bobot nilai jawaban

Jawaban	Bobot
Sangat setuju	4
Setuju	3
Tidak setuju	2
Sangat tidak setuju	1

Tabel 4.8 Pertanyaan kuesioner

No.	Pertanyaan
1	Apakah tampilan aplikasi ini <i>user-friendly</i> ?

2	Apakah menu-menu aplikasi ini mudah dipahami?
3	Apakah fitur pada aplikasi berfungsi dengan baik?
4	Apakah fitur pada aplikasi membantu dalam program <i>link-match</i> ?
5	Apakah aplikasi ini sudah cukup baik?

BAB V

IMPLEMENTASI DAN EVALUASI

Pada bab ini akan dijelaskan mengenai analisis dan rancangan dari sistem yang akan dibangun berdasarkan pengamatan. Selanjutnya, tahapan-tahapan yang akan dilakukan antara lain yaitu menganalisa sistem, membuat pemodelan sistem dengan *Unified Modelling Language (UML)*, merancang *user story* yang kemudian akan dikumpulkan menjadi *product backlog*, dan dilanjutkan dengan perancangan antar muka aplikasi.

5.1 Implementasi Scrum

5.1.1 Scrum Team

Scrum team adalah seluruh *stakeholder* yang terlibat dalam proses pembuatan aplikasi *link-match* ini.

Tabel 5.1 Scrum team

Posisi	Nama
<i>Product owner</i>	Sirojul Munir
<i>Scrum master</i>	Atmam Alfaruqi
<i>Developer</i>	Ahmad Fathan Syakir
	Ardith Lutfiawan
	Fatkurrahman Jundi
	Huda Izzatulhaq
	Muhammad Azhar Rasyad

5.1.2 Sprint Planning

Pada pengembangan aplikasi *link-match* ini dilaksanakan *sprint planning* dalam rangka menentukan target-target yang akan dicapai dalam proses pengembangan aplikasi dan dilakukan sebelum setiap *sprint* dimulai. Berikut adalah rincian pelaksanaan *scrum*:

A. *Sprint 1*

Pada *sprint 1* ini belum melakukan penulisan kode program, melainkan berisi tahap pertama sebelum pembuatan aplikasi yaitu perancangan *use case* dan *activity diagram* yang penulis cantumkan hasilnya pada bab sebelumnya. Target minimal pencapaian (*Minimum Value Product*) pada *sprint 1* adalah membuat *use case diagram* dan *activity diagram*. Berikut adalah tabel *sprint 1*:

Tabel 5.2 Sprint 1

Kode	User story	Target	Realisasi
MD-001	Membuat <i>use case diagram</i>	<i>Sprint 1</i>	<i>Sprint 1</i>
MD-002	Membuat daftar fitur utama	<i>Sprint 1</i>	<i>Sprint 1</i>
MD-003	Membuat <i>activity diagram</i> modul dosen	<i>Sprint 1</i>	<i>Sprint 1</i>
MD-004	Membuat <i>activity diagram: user management</i>	<i>Sprint 1</i>	<i>Sprint 1</i>
MD-005	Membuat <i>activity diagram: team management</i>	<i>Sprint 1</i>	<i>Sprint 1</i>
MD-006	Membuat <i>activity diagram: project management</i>	<i>Sprint 1</i>	<i>Sprint 1</i>

B. *Sprint 2*

Pada *sprint* ini penulis berada di tahapan pembuatan *mockup* aplikasi *mobile link-match*. Hasil daripada *mockup* ini telah penulis cantumkan pada dalam bab sebelumnya. Target minimal pencapaian (*Minimum Value Product*) pada *sprint 2* adalah membuat tampilan *mockup* aplikasi. Berikut adalah apa saja yang dilakukan pada *sprint* ini:

Tabel 5.3 Sprint 2

Kode	User story	Target	Realisasi
MD-007	Membuat <i>mockup</i> sistem: <i>user management</i>	<i>Sprint 2</i>	<i>Sprint 2</i>

MD-008	Membuat <i>mockup</i> sistem: <i>team management</i>	<i>Sprint 2</i>	<i>Sprint 2</i>
MD-009	Membuat <i>mockup</i> sistem: <i>project management</i>	<i>Sprint 2</i>	<i>Sprint 2</i>

C. *Sprint 3*

Pada *sprint 3* ini telah dimulai untuk penulisan kode program untuk masing-masing fiturnya. Target minimal pencapaian (*Minimum Value Product*) pada *sprint 3* adalah membuat menu *login* dengan validasi dan tampilan *dashboard*.

Berikut adalah apa saja yang dilakukan pada *sprint* ini:

Tabel 5.4 Sprint 3

Kode	User story	Target	Realisasi	API & Response
				baseurl: <code>https://linkmatchsttnfapi.herokuapp.com</code>
MD-010	<i>Login user</i>	<i>Sprint 3</i>	<i>Sprint 3</i>	<code>\$baseurl/api/login (POST)</code> RESPONSE: 200 (BERHASIL)
MD-011	<i>View profile</i>	<i>Sprint 3</i>	<i>Sprint 3</i>	<code>\$baseurl/api/profile (GET)</code> RESPONSE: 200 (BERHASIL)
MD-012	<i>Logout user</i>	<i>Sprint 3</i>	<i>Sprint 4</i>	<code>\$baseurl/api/login (DELETE)</code> RESPONSE: 200 (BERHASIL)
MD-013	Halaman <i>dashboard</i>	<i>Sprint 3</i>	<i>Sprint 4</i>	<code>\$baseurl/api/user (GET)</code> RESPONSE: 200 (BERHASIL)
MD-014	Fitur melihat daftar <i>project</i>	<i>Sprint 3</i>	<i>Sprint 4</i>	<code>\$baseurl/api/project (GET)</code> RESPONSE: 200 (BERHASIL)
MD-015	Fitur melihat <i>detail project</i>	<i>Sprint 3</i>	<i>Sprint 4</i>	<code>\$baseurl/api/project (GET)</code> RESPONSE: 200 (BERHASIL)

MD-016	Fitur melihat daftar <i>sprint</i>	<i>Sprint 3</i>	<i>Sprint 3</i>	<pre>\$baseurl/api/sprint (GET) RESPONSE: 200 (BERHASIL)</pre>
MD-017	Fitur melihat <i>detail sprint</i>	<i>Sprint 3</i>	<i>Sprint 3</i>	<pre>\$baseurl/api/sprint (GET) RESPONSE: 200 (BERHASIL)</pre>

Proses penggerjaan pada *sprint* ini memiliki ketidak sesuaian dengan *sprint planning* dimana target dan realisasi yang berbeda.

- Implementasi BLoC

Tabel 5.5 BLoC *sprint* 3

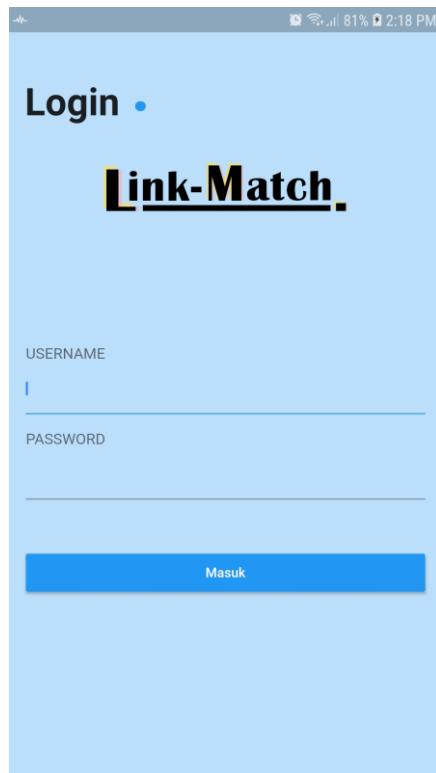
Kode	User story	UI	BLoC	Repository	Network Provider
MD-010	<i>Login user</i>	<i>LoginScreen (stateless)</i>	<i>LoginBloc</i>	<i>postLoginUser</i>	<i>UserApiProvider, ApiAuthProvider</i>
MD-011	<i>View profile</i>	<i>ProfilePage (stateful)</i>	<i>UsersBloc</i>	<i>fetchAllUsers</i>	<i>UserApiProvider</i>
MD-012	<i>Logout user</i>	<i>MyHomePage (stateful)</i>	<i>AuthorizationBloc</i>	-	<i>UserApiProvider</i>
MD-013	Halaman dashboard	<i>MyHomePage (stateful)</i>	<i>HomeBloc</i>	-	<i>UserApiProvider</i>
MD-014	Fitur melihat daftar project	<i>ProjectList (stateful)</i>	<i>ProjectsBloc</i>	<i>fetchAllProjects</i>	<i>ProjectApiProvider</i>
MD-015	Fitur melihat detail project	<i>ProjectDetail (stateful)</i>	<i>ProjectsBloc</i>	<i>fetchAllProjects</i>	<i>ProjectApiProvider</i>
MD-016	Fitur melihat daftar sprint	<i>SprintList (stateful)</i>	<i>SprintsBloc</i>	<i>fetchAllSprints</i>	<i>SprintApiProvider</i>
MD-017	Fitur melihat	<i>SprintDetail (stateful)</i>	<i>SprintsBloc</i>	<i>fetchAllSprints</i>	<i>SprintApiProvider</i>

	<i>detail</i> <i>sprint</i>				
--	--------------------------------	--	--	--	--

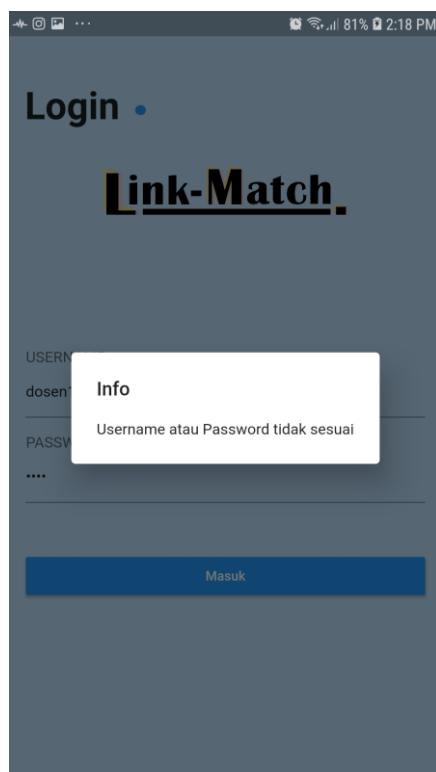
- Tampilan Antarmuka Aplikasi

Berikut adalah tampilan antarmuka aplikasi pada *sprint 3*:

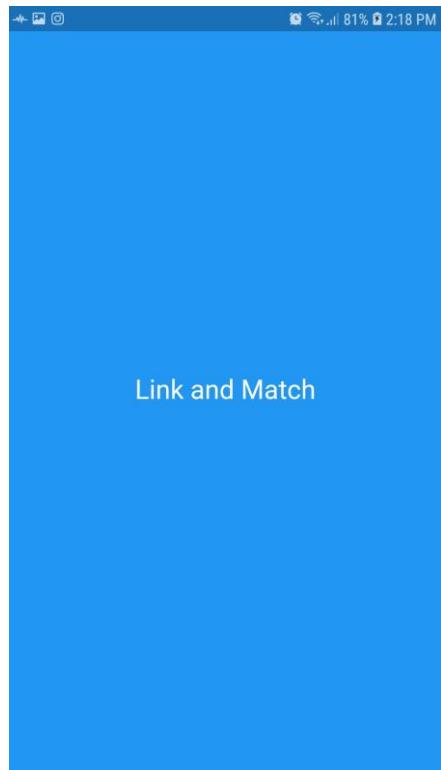
a) Halaman *login user*



Gambar 5.1 *Login user*

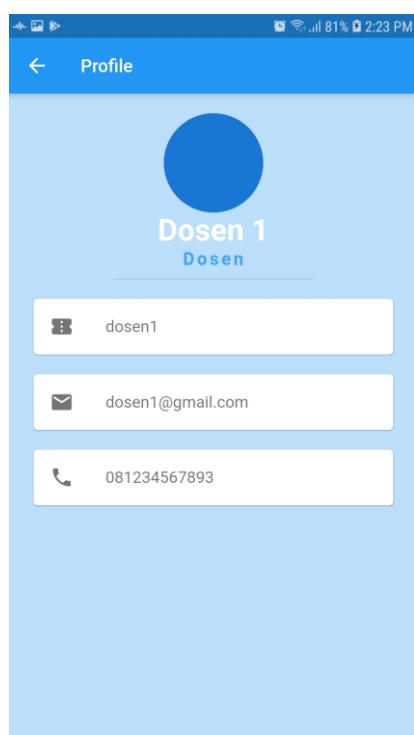


Gambar 5.2 Validasi *login user*



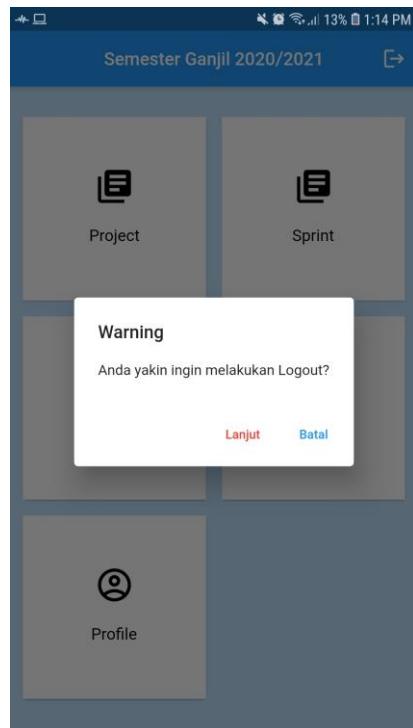
Gambar 5.3 *Splash screen*

b) *View profile*



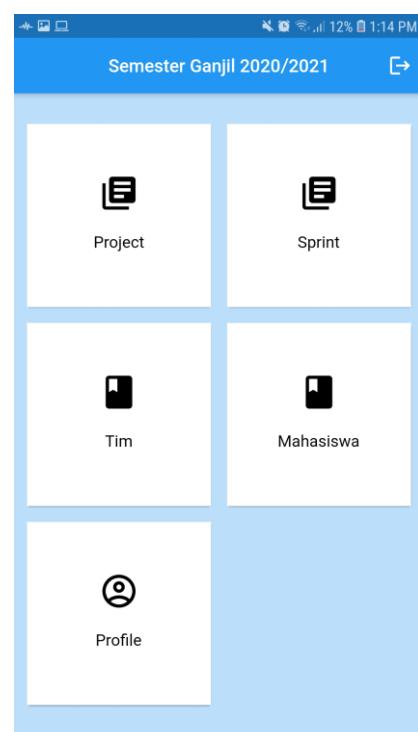
Gambar 5.4 *View profile*

c) *Logout user*

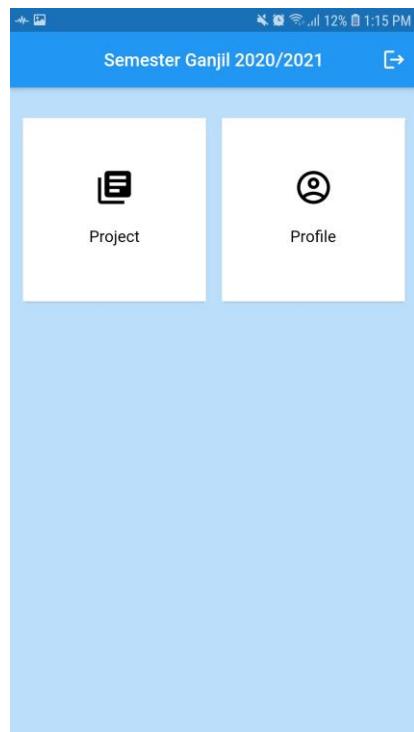


Gambar 5.5 *Logout user*

d) Halaman *dashboard*

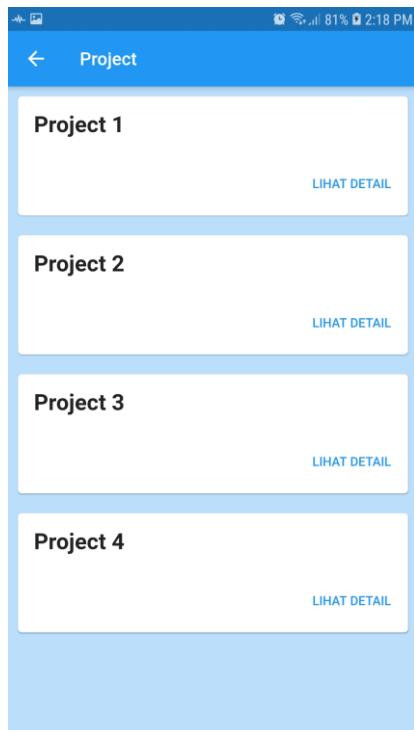


Gambar 5.6 *Dashboard scrum master dan dosen*



Gambar 5.7 Dashboard product owner

e) Halaman *list project*



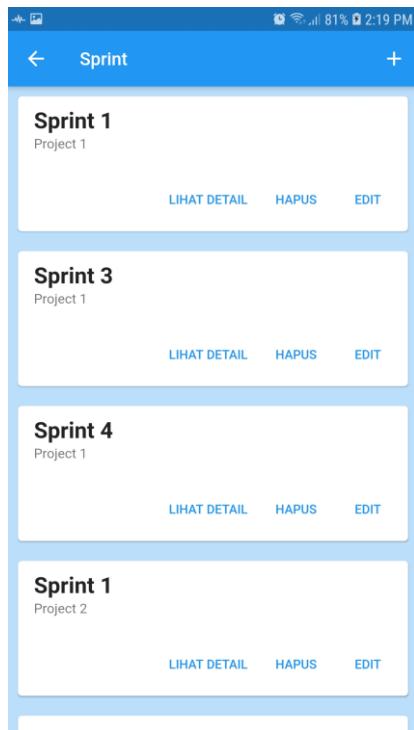
Gambar 5.8 List project

f) Halaman *detail project*



Gambar 5.9 Detail project

g) Halaman *list sprint*



Gambar 5.10 *List sprint*

h) Halaman *detail sprint*



Gambar 5.11 *Detail sprint*

D. Sprint 4

Pada *sprint 4* Target minimal pencapaian (*Minimum Value Product*) pada *sprint 4* adalah melakukan CRUD pada *sprint* dan melihat daftar tim. Pada *sprint* ini terdapat beberapa fitur yang dikerjakan, diantaranya:

Tabel 5.6 Sprint 4

Kode	User story	Target	Realisasi	API & Response
MD-018	Fitur menambah sprint	Sprint 4	Sprint 3	<pre>\$baseurl: https://linkmatchsttnfapi.herokuapp.com</pre> <pre>\$baseurl/api/sprint/{id} (POST)</pre> <p>RESPONSE: 200 (BERHASIL)</p>
MD-019	Fitur memulai sprint	Sprint 4	Sprint 6 (perbaikan)	<pre>\$baseurl/api/sprint/{id} (PUT)</pre> <p>RESPONSE: 200 (BERHASIL)</p>
MD-020	Fitur mengakhiri sprint	Sprint 4	Sprint 6 (perbaikan)	<pre>\$baseurl/api/sprint/{id} (PUT)</pre> <p>RESPONSE: 200 (BERHASIL)</p>
MD-021	Fitur mengedit sprint	Sprint 4	Sprint 5	<pre>\$baseurl/api/sprint/{id} (PUT)</pre> <p>RESPONSE: 200 (BERHASIL)</p>
MD-022	Fitur menghapus sprint	Sprint 4	Sprint 5	<pre>\$baseurl/api/sprint//{id} (DELETE)</pre> <p>RESPONSE: 200 (BERHASIL)</p>
MD-023	Fitur melihat daftar tim	Sprint 4	Sprint 4	<pre>\$baseurl/api/team (GET)</pre> <p>RESPONSE: 200 (BERHASIL)</p>
MD-024	Fitur melihat detail tim	Sprint 4	Sprint 4	<pre>\$baseurl/api/team (GET)</pre> <p>RESPONSE: 200 (BERHASIL)</p>

MD-025	Fitur melihat daftar mahasiswa	<i>Sprint 4</i>	<i>Sprint 4</i>	<pre>\$baseurl/api/mahasiswa (GET)</pre> <p style="text-align: center;">RESPONSE: 200 (BERHASIL)</p>
--------	--------------------------------	-----------------	-----------------	--

- Implementasi BLoC

Tabel 5.7 BLoC sprint 4

Kode	User story	UI	BLoC	Repository	Network Provider
MD-018	Fitur menambah sprint	<i>AddSprint (stateful)</i>	<i>SprintsBloc</i>	<i>createNewSprint (projectID, namaSprint, tglMulai, tglAkhir)</i>	<i>SprintApiProvider</i>
MD-019	Fitur memulai sprint	<i>SprintDetail (stateful)</i>	<i>SprintsBloc</i>	<i>updateStatusSprint (id, status)</i>	<i>SprintApiProvider</i>
MD-020	Fitur mengakhiri sprint	<i>SprintDetail (stateful)</i>	<i>SprintsBloc</i>	<i>updateStatusSprint (id, status)</i>	<i>SprintApiProvider</i>
MD-021	Fitur mengedit sprint	<i>UpdateSprint (stateful)</i>	<i>SprintsBloc</i>	<i>updateSprint (id, projectID, namaSprint, tglMulai, tglAkhir)</i>	<i>SprintApiProvider</i>
MD-022	Fitur menghapus sprint	<i>SprintList (stateful)</i>	<i>SprintsBloc</i>	<i>deleteSprint(id)</i>	<i>SprintApiProvider</i>
MD-023	Fitur melihat daftar tim	<i>TeamList (stateful)</i>	<i>TeamsBloc</i>	<i>fetchAllTeams</i>	<i>TeamApiProvider</i>

MD-024	Fitur melihat detail tim	<i>TeamDetail</i>	<i>TeamsBloc</i>	<i>fetchAllTeams</i>	<i>TeamApiProvider</i>
MD-025	Fitur melihat daftar mahasiswa	<i>MahasiswaList</i>	-	-	<i>JsonMahasiswaAdapter</i>

- Tampilan Antarmuka Aplikasi

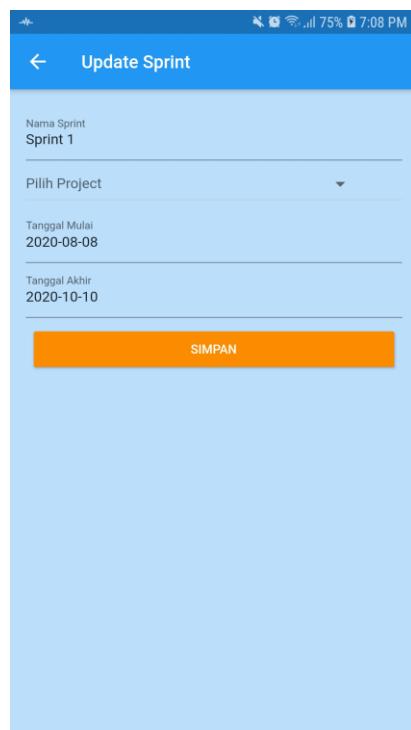
Berikut adalah tampilan antarmuka aplikasi pada *sprint 4*:

a) Halaman menambah *sprint*



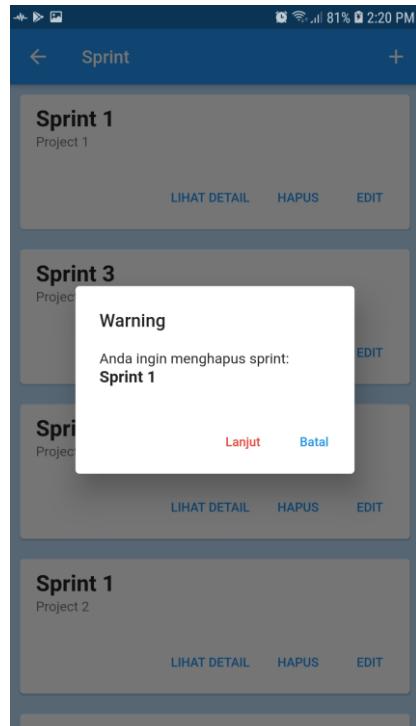
Gambar 5.12 Add Sprint

b) Halaman *edit sprint*



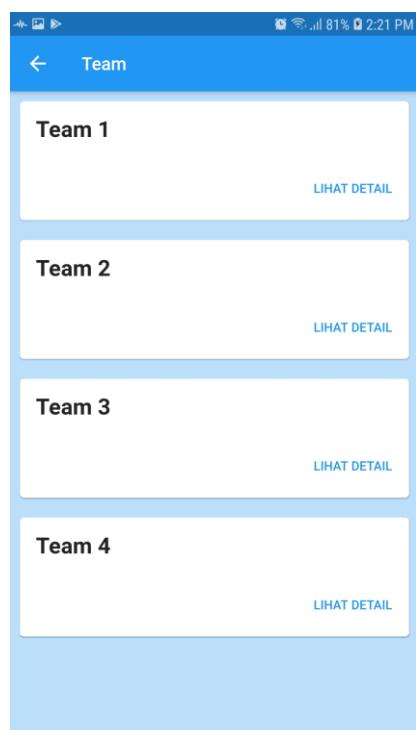
Gambar 5.13 Edit sprint

c) Menghapus *sprint*



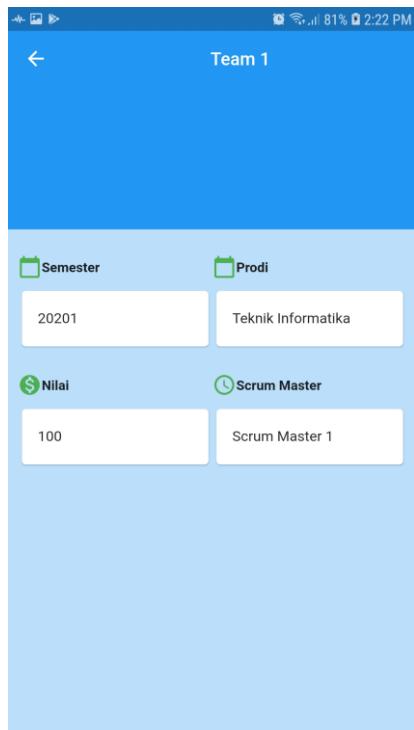
Gambar 5.14 Delete sprint

d) Halaman *list tim*



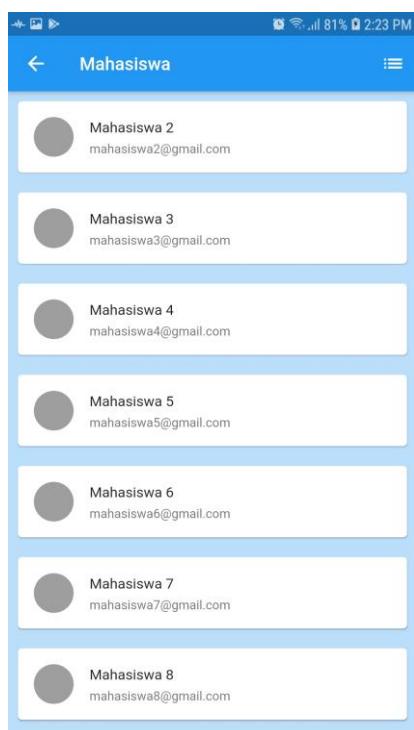
Gambar 5.15 List tim

e) Halaman *detail* tim



Gambar 5.16 *Detail* tim

f) Halaman *list* mahasiswa



Gambar 5.17 *List* mahasiswa

E. Sprint 5

Pada *sprint 5* terdapat beberapa fitur yang dikerjakan dan rancangan pengujian aplikasi. Target minimal pencapaian (*Minimum Value Product*) pada *sprint 5* adalah melihat daftar nilai mahasiswa dan tim. Berikut adalah apa saja yang dilakukan pada *sprint 5* ini:

Tabel 5.8 Sprint 5

Kode	User story	Target	Realisasi	API & Response
MD-026	Fitur melihat daftar nilai tim	<i>Sprint 5</i>	Belum selesai	(belum dibuat)
MD-027	Fitur melihat daftar nilai mahasiswa	<i>Sprint 5</i>	Belum selesai	(belum dibuat)
MD-028	Rancangan <i>Black-box Testing</i>	<i>Sprint 5</i>	<i>Sprint 4</i>	-
MD-029	Rancangan <i>UAT</i>	<i>Sprint 5</i>	<i>Sprint 4</i>	-

- Implementasi BLoC

Tabel 5.9 BLoC sprint 5

Kode	User story	UI	BLoC	Repository	Network Provider
MD-026	Fitur melihat daftar nilai tim	<i>NilaiMahasiswaList (stateful)</i>	-	-	-

MD-027	Fitur melihat daftar nilai mahasiswa	-	-	-	-
--------	--------------------------------------	---	---	---	---

5.1.3 Sprint Review

Berikut ini adalah *sprint review* pada pengembangan aplikasi *link-match* modul dosen berbasis Android yang penulis jelaskan dalam sebuah tabel.

Tabel 5.10 Sprint review

No	Sprint	Task	Sprint Review
1	<i>Sprint 0</i>	-	Persiapan sebelum <i>sprint</i> dimulai bersama dengan <i>product owner</i> dan <i>development team</i> .
2	<i>Sprint 1</i>	<ul style="list-style-type: none"> • Membuat <i>user story</i> • Membuat <i>use case diagram</i> • Membuat daftar fitur utama • Membuat <i>activity diagram</i> 	<ul style="list-style-type: none"> • Rancangan sistem sudah cukup baik • Tambahkan tabel member_tim_nilai dan <i>sprint</i> • Dibuatkan <i>login</i> masing-masing <i>user role</i> • Aplikasi <i>mobile</i> hanya fokus pada transaksi yang tidak merubah <i>master data</i> • Ambil data dari <i>web service</i>
3	<i>Sprint 2</i>	<ul style="list-style-type: none"> • Membuat <i>mockup user management</i> • Membuat <i>mockup team management</i> • Membuat <i>mockup project management</i> 	<i>User interface</i> sudah cukup hanya perlu tambahkan judul pada App bar
4	<i>Sprint 3</i>	<ul style="list-style-type: none"> • Halaman <i>login</i> • Halaman <i>logout</i> • Halaman <i>dashboard</i> • Halaman <i>profile</i> • Halaman <i>daftar project</i> dan <i>detail project</i> 	<ul style="list-style-type: none"> • Tambahkan rancangan ERD sebelum <i>use case diagram</i> • Semua halaman sudah sesuai dengan rancangan sistem • Pada halaman <i>dashboard</i> tambahkan judul semester saat ini

		<ul style="list-style-type: none"> • Halaman daftar <i>sprint</i> dan <i>detail sprint</i> 	
5	<i>Sprint 4</i>	<ul style="list-style-type: none"> • Menambah <i>sprint</i> • Memulai <i>sprint</i> • Mengakhiri <i>sprint</i> • Mengubah <i>sprint</i> • Menghapus <i>sprint</i> • Halaman daftar tim dan detail <i>sprint</i> • Halaman daftar mahasiswa 	<ul style="list-style-type: none"> • Beberapa fitur belum dibuat • Gunakan validasi pada <i>login</i> • Dibuatkan filter by user pada halaman dashboard
6	<i>Sprint 5</i>	<ul style="list-style-type: none"> • Halaman daftar nilai tim • Halaman daftar nilai mahasiswa • Hasil <i>black-box testing</i> • Hasil <i>user acceptance test</i> 	<ul style="list-style-type: none"> • API sudah bisa diakses pada aplikasi mobile • <i>Black-box testing</i> penguji oleh <i>developer</i> • UAT
7	<i>Sprint 6</i>	-	Dipastikan lagi semua fitur untuk siap uji

5.1.4 Evaluasi Scrum

Scrum team pada penelitian ini menentukan banyaknya *sprint* adalah 5 *sprint* dan pada setiap *sprint* memiliki waktu penggerjaan 25-30 hari. Tetapi pada pelaksanaannya, penulis menambahkan satu *sprint* baru dikarenakan belum mencapai target minimal berfungsinya fitur pada aplikasi.

Kemudian pada implementasi tahapan *scrum*, *scrum team* telah melaksanakan *sprint planning* dan *sprint review*, namun untuk tahapan seperti *daily meeting* dan *sprint retrospective* tidak terlaksana. Hal ini dikarenakan *scrum team* belum terbentuk secara utuh dan baru menentukan *scrum master* saat *sprint 3* sedang berjalan.

5.2 Pengujian Sistem

5.2.1 Hasil *Black-box Testing*

Pada pengujian *black-box testing* dilakukan hanya untuk mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian ini dilakukan oleh penguji sebagai pengembang daripada aplikasi. Hasil *black-box testing* selanjutnya akan dijabarkan pada tabel berikut:

Tabel 5.11 Hasil *black-box testing*

No.	Pengujian	User Story	Hasil yang diharapkan	Hasil
1	<i>Login user</i>	Halaman <i>login</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	Berhasil
		<i>Form login</i> berisi <i>username</i> dan <i>password</i>	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	Berhasil
		<i>Submit button: login</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	Berhasil
		<i>Splash-screen</i> dan halaman <i>dashboard</i>	<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	Berhasil
		Pesan validasi <i>error</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	Berhasil
2	<i>View profile</i>	Menu <i>profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	Berhasil
		Halaman <i>profile</i>	<i>User</i> dapat menampilkan halaman <i>profile</i>	Berhasil
3	<i>Logout user</i>	<i>Logout user</i>	<i>User</i> dapat mengklik <i>icon logout</i> pada halaman <i>dashboard</i> untuk proses <i>logout</i>	Berhasil
		Halaman <i>login</i>	<i>User</i> dapat menampilkan halaman <i>login</i> setelah proses <i>logout</i> selesai	Berhasil

4	Halaman <i>dashboard</i>	Halaman <i>dashboard</i>	<i>User</i> dapat menampilkan halaman <i>dashboard</i> sesuai dengan <i>role</i> tertentu	Berhasil
5	<i>View project</i>	Menu <i>project</i>	<i>User</i> dapat memilih menu <i>project</i> pada halaman <i>dashboard</i>	Berhasil
		Halaman <i>project</i>	<i>User</i> dapat menampilkan halaman <i>project</i>	Berhasil
		Halaman <i>detail project</i>	<i>User</i> dapat menampilkan halaman <i>project</i> yang dipilih	Berhasil
6	<i>View sprint</i>	Menu <i>sprint</i>	<i>User</i> dapat memilih menu <i>sprint</i> pada halaman <i>dashboard</i>	Berhasil
		Halaman <i>sprint</i>	<i>User</i> dapat menampilkan halaman <i>sprint</i>	Berhasil
		Halaman <i>detail sprint</i>	<i>User</i> dapat menampilkan halaman <i>detail sprint</i> yang dipilih	Berhasil
		<i>Add button:</i> tambah <i>sprint</i>	<i>User</i> dapat mengklik <i>add button</i> untuk menampilkan halaman tambah <i>sprint</i>	Berhasil
		<i>Delete button:</i> hapus <i>sprint</i>	<i>User</i> dapat mengklik <i>delete button</i> untuk menampilkan <i>pop-up</i> hapus <i>sprint</i>	Berhasil
7	<i>Add sprint</i>	Halaman tambah <i>sprint</i>	<i>User</i> dapat menampilkan <i>form</i> tambah <i>sprint</i>	Berhasil
		Pesan <i>form error</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai	Berhasil
		<i>Submit button:</i> tambah <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses tambah <i>sprint</i>	Berhasil
8	Memulai <i>sprint</i>	<i>Submit button:</i> mulai <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses mulai <i>sprint</i>	Berhasil
9	Mengakhiri <i>sprint</i>	<i>Submit button:</i> akhiri <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses akhiri <i>sprint</i>	Berhasil
10	<i>Edit sprint</i>	Halaman <i>edit sprint</i>	<i>User</i> dapat menampilkan <i>form edit sprint</i>	Berhasil

		Pesan <i>form error</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai	Berhasil
		<i>Submit button: edit sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses <i>edit sprint</i>	Berhasil
11	<i>Delete sprint</i>	<i>Pop-up: hapus sprint</i>	<i>User</i> dapat melihat <i>pop-up hapus sprint</i>	Berhasil
		<i>Submit button: hapus sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses hapus <i>sprint</i>	Berhasil
		<i>Back button: batal hapus sprint</i>	<i>User</i> dapat mengklik <i>back button</i> untuk membatalkan hapus <i>sprint</i>	Berhasil
12	<i>View tim</i>	Menu tim	<i>User</i> dapat memilih menu tim pada halaman <i>dashboard</i>	Berhasil
		Halaman tim	<i>User</i> dapat menampilkan halaman tim	Berhasil
		Halaman <i>detail</i> tim	<i>User</i> dapat menampilkan halaman tim yang dipilih	Berhasil
		<i>List button: view nilai tim</i>	<i>User</i> dapat mengklik <i>list button</i> untuk menampilkan halaman nilai tim	Gagal
13	<i>View mahasiswa</i>	Menu mahasiswa	<i>User</i> dapat memilih menu mahasiswa pada halaman <i>dashboard</i>	Berhasil
		Halaman mahasiswa	<i>User</i> dapat menampilkan halaman mahasiswa	Berhasil
		<i>List button: view nilai mahasiswa</i>	<i>User</i> dapat mengklik <i>list button</i> untuk menampilkan halaman nilai mahasiswa	Berhasil
14	<i>View nilai</i>	Halaman nilai tim	<i>User</i> dapat menampilkan halaman nilai tim	Gagal
		Halaman nilai mahasiswa	<i>User</i> dapat menampilkan halaman nilai mahasiswa	Gagal

Hasil pengujian *black-box testing* menggambarkan seberapa baik aplikasi ini secara fungsionalitas yang telah dieksekusi oleh penulis dan dapat ditarik kesimpulan sebagai berikut:

Dari hasil 14 pengujian tercatat sebanyak 12 pengujian berhasil dan 2 pengujian tidak atau belum berhasil, ini berarti menunjukkan fungsi daripada aplikasi adalah 85.7% dapat berjalan dengan baik. Sedangkan 14.3% belum dapat dikategorikan sesuai dengan fungsi yang diharapkan. Pengujian dengan catatan kurang baik ini antara lain pada *user story* (fitur) *list button* nilai tim, melihat nilai tim dan melihat nilai mahasiswa.

5.2.2 Hasil *User Acceptance Test*

User Acceptance Test merupakan suatu rangkaian dari tahap pengujian yang dilakukan *end-user* atau pengguna aplikasi. Pada pengujian UAT penelitian ini melibatkan 3 *user role* yaitu *product owner*, dosen dan *scrum master*. Selanjutnya dari *role-role* tersebut pengujian dilakukan kepada 1 *product owner*, 3 dosen dan 3 *scrum master*. Adapun rumus dari perhitungan UAT ini yaitu:

$$\begin{aligned}\textbf{Total Uji} &= \text{Responden} \times \text{Butir Uji} \\ \textbf{Jumlah} &= \text{Total Berhasil} / \text{Total Uji} * 100\end{aligned}$$

Hasil daripada pengujian UAT sebagaimana terlampir adalah sebagai berikut:

Tabel 5.12 Hasil pengujian UAT *user product owner*

Responden	UAT		Total Berhasil	Total Gagal	Jumlah
	Butir Uji	Total Uji			
1	13	13	13	0	100%

Pada pengujian UAT *user product owner* tercatat bahwa tingkat keberhasilan aplikasi mencapai 100% dengan rincian yaitu 13 butir uji berhasil dan tanpa pengujian gagal.

Tabel 5.13 Hasil pengujian UAT *user dosen*

Responden	UAT		Total Berhasil	Total Gagal	Jumlah
	Butir Uji	Total Uji			
4	25	100	100	0	100%

Hasil pengujian UAT *user dosen* memiliki tingkat keberhasilan aplikasi mencapai 100% dengan rincian yaitu 100 butir total pengujian yang berhasil dan tanpa pengujian gagal.

Tabel 5.14 Hasil pengujian UAT *user scrum master*

Responden	UAT		Total Berhasil	Total Gagal	Jumlah
	Butir Uji	Total Uji			
3	34	102	95	7	93%

Hasil pengujian UAT *user scrum master* memiliki tingkat keberhasilan aplikasi mencapai 95% dengan rincian yaitu 95 butir total pengujian yang berhasil dan dengan 7 pengujian gagal.

5.2.3 Hasil Kuesioner

Adapun dalam menghitung tingkat kepuasan terhadap aplikasi ini penulis menggunakan indeks persentase sebagai berikut untuk menilai seberapa baik aplikasi yang telah dibangun.

Tabel 5.15 Indeks persentase

Indeks Persentase	Penilaian
0% - 24,99%	Tidak Baik
25% - 49,99%	Kurang Baik
50% - 74,99 %	Baik
75% - 100%	Sangat Baik

Pada kuesioner aplikasi *link-match* modul dosen berbasis Android ini penulis melakukan perhitungan pada 5 pertanyaan yang diajukan dan didapatkan nilai rata-rata mencapai 77%. Hasil daripada kuesioner ini adalah sebagai berikut:

Tabel 5.16 Hasil skala likert

SKALA LIKERT		JUMLAH	PERSENTASE
NO. 1 Apakah tampilan aplikasi ini <i>user-friendly</i> ?	Sangat Setuju	4	0
	Setuju	3	7
	Tidak Setuju	2	1
	Sangat Tidak Setuju	1	0
NO. 2 Apakah menu-menu aplikasi ini mudah dipahami?	Sangat Setuju	4	2
	Setuju	3	6
	Tidak Setuju	2	0
	Sangat Tidak Setuju	1	0
NO. 3 Apakah fitur pada aplikasi berfungsi dengan baik?	Sangat Setuju	4	3
	Setuju	3	4
	Tidak Setuju	2	1
	Sangat Tidak Setuju	1	0
NO. 4 Apakah fitur pada aplikasi membantu dalam program <i>link-match</i> ?	Sangat Setuju	4	1
	Setuju	3	7
	Tidak Setuju	2	0
	Sangat Tidak Setuju	1	0
NO. 5 Apakah aplikasi ini sudah cukup baik?	Sangat Setuju	4	1
	Setuju	3	6
	Tidak Setuju	2	1
	Sangat Tidak Setuju	1	0

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Pada pengembangan aplikasi *link-match* modul dosen berbasis Android diawali dengan menganalisa kebutuhan sistem pada program *link-match* STT NF berdasarkan hasil wawancara kepada *stakeholder* internal kampus. Dilanjutkan dengan membentuk *development team* dalam rangka melakukan penelitian dan pembuatan aplikasi dalam menjawab persoalan melalui pendekatan *scrum*. Diawali dengan perancangan *product backlog* dan *sprint planning* untuk menyederhanakan kebutuhan sistem. Dari hasil perancangan, implementasi dan evaluasi sistem kemudian didapatkan kesimpulan sebagai berikut:

1. Aplikasi yang penulis kembangkan menggunakan *Flutter Framework* ini tidak mengubah mekanisme dari proses pembelajaran program *link-match*, melainkan dapat membantu dalam manajemen *project* sebab banyak aplikasi pihak ketiga yang digunakan maupun *platform* yang sulit diakses saat berjalannya program *link-match*.
2. Fitur aplikasi yang berhasil dibuat pada penelitian ini antara lain: *login*, *logout*, melihat *profile user*, melihat informasi *project* dan *sprint*, melakukan CRUD pada *sprint*, melihat tim dan mahasiswa.
3. Berdasarkan hasil pengujian UAT aplikasi ini memiliki tingkat keberhasilan mencapai 100% pada *user product owner*, 100% pada *user dosen* dan 93% *user scrum master*. Kemudian untuk tingkat kepuasaan pengguna terhadap aplikasi yang diperoleh melalui kuesioner yaitu sebesar 77.5%. Hal ini memberikan kesimpulan bahwa aplikasi *link-match* modul dosen ini cukup baik dan dapat membantu pelaksanaan program *link-match* di STT Terpadu Nurul Fikri.

6.2 Saran

Pada penelitian ini penulis mendapatkan saran yang didapat dari hasil pengujian pada aplikasi untuk bisa dipertimbangkan dalam tercapainya tujuan dan dapat mendukung program link-match di STT Terpadu Nurul Fikri ke depannya. Beberapa

fitur pada aplikasi perlu untuk diperbaiki karena pada dasarnya belum mencapai keberhasilan sebagaimana yang diharapkan pada *output* pengujian di *user story*, fitur-fitur tersebut diantaranya: *add sprint*, *edit sprint*, melihat nilai tim dan melihat nilai mahasiswa. Masalah yang penulis alami sehingga belum maksimal dalam proses pembuatan fitur-fitur tersebut adalah waktu yang terbatas serta belum berkoordinasi dengan baik dengan *developer team* terkait pembuatan API untuk digunakan pada aplikasi ini.

Penulis juga mendapatkan beberapa saran untuk penyempurnaan aplikasi, berikut adalah saran untuk aplikasi:

- Tampilan UI hendaknya dibuat lebih proporsional
- Dibuat agar loading aplikasi tidak lama
- Harusnya dapat melihat nama anggota tim pada halaman *detail* tim
- Nilai mahasiswa langsung saja tampilkan daftar nilainya di halaman awal
- Buat tombol yang dapat *view password* pada halaman *login*
- Tampilkan foto *user*
- *Sprint* dikelompokkan bedasarkan *project* saja
- Tampilkan indikator proses berhasil atau gagal
- Halaman mahasiswa diganti menjadi nilai mahasiswa saja
- Tampilkan notifikasi tidak terhubung ke internet

DAFTAR PUSTAKA

- Ardiansyah, H. I. (2017). *Pengembangan REST API Sistem Informasi Marketplace Pariwisata Indoexplore.id.*
- Badan Pusat Statistik. (2019). *Statistik Karakteristik Usaha 2019 1.*
- Bloc, a predictable state management library for Dart.* (n.d.).
<https://bloclibrary.dev/#/>
- Christmantara, V. (2013). Metodologi scrum. In *iSTTS* (pp. 1–6).
- Dharwiyanti, S., & Wahono, R. S. (2003). Pengantar Unified Modeling LAnguage (UML). *IlmuKomputer.Com*, 1–13. <http://www.unej.ac.id/pdf/yanti-uml.pdf>
- Disas, E. P. (2018). Link and Match sebagai Kebijakan Pendidikan Kejuruan. *Indonesia, Universitas Pendidikan Barat, Jawa*, 18(2), 233.
- Helmi, T., Munjin, R. A., & Purnamasari, I. (2017). Kualitas Pelayanan Publik Dalam Pembuatan Izin Trayek Oleh Dllaj Kabupaten Bogor. *Jurnal Governansi*, 2(1), 51. <https://doi.org/10.30997/jgs.v2i1.209>
- Herman, I. (2016). *Pengembangan Sumber Daya Umat Di Era Globalisasi Dan Modernisasi.* 2.
- J. F. DiMarzio. (2016). *Beginning Android® Programming with Android Studio.*
- Johan, A. B., PH, S., & Widodo. (2019). *Evaluation Of Education Implementation Of Link And Match Systems Of The Industrial And Vocational School In Yogyakarta Province.* 7(2), 216–222.
- Joyo, D., Rajabi, N., Anggara, R., & Rahmani, A. (2020). *Perbandingan Penggunaan Reactive Programming dan Object- Oriented Programming pada Kinerja Aplikasi Sederhana Berbasis Android.* 26–27.
- Munir, S., Adidaya, W., & Sasmita, H. (2016). Perancangan Sistem Informasi Akademik Berbasis Web Menggunakan Framework MVC Pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. *Jurnal Informatika Terpadu*, 2(1), 1–8.
<http://journal.nurulfikri.ac.id/index.php/JIT/article/view/50/36>
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). *PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN).* 1(3), 31–36.

- Ni Ayu Krisna Dewi, Zukhri, A., & I Ketut Dunia. (2014). Analisis Faktor-Faktor Penyebab Anak Putus Sekolah Usia Pendidikan Dasar di kecamatan Gerokgak Tahun 2012/2013. *Pendidikan Ekonomi Universitas Pendidikan Ganesha*, 4(1), 1–14.
- Prajyot Mainkar, & Giordano, S. (2019). Google Flutter Mobile Development Quick Start Guide. In *Packt Publishing* (Vol. 53, Issue 9). Packt.com.
<https://doi.org/10.1017/CBO9781107415324.004>
- Program Class Link & Match di STT Terpadu Nurul Fikri*. (2019). STT Terpadu Nurul Fikri. <https://nurulfikri.ac.id/program-class-link-match-di-stt-terpadu-nurul-fikri/>
- Pujihastuti, I. (2010). Prinsip Penulisan Kuesioner Penelitian. *Jurnal Agribisnis Dan Pengembangan Wilayah*, 2(1), 43–56.
- Putra, B. W., Saputra, A., & Sanjaya, R. (2019). *Implementasi Framework CodeIgniter dan Restful API pada Sistem Informasi Manajemen Tugas Akhir*. 5(1), 307–313.
- Raharjo, B. (2019). *Pemrograman Android Dengan Flutter*. Informatika Bandung.
- Riansyah, E. (2015). *Analisis Dan Perancangan Sistem Informasi Akademik Stt-Nf Modul Rencana Studi Berbasis Web Menggunakan Mvc Framework*.
- Saeprrahman, Setiawan, A., & Handriani, I. (2019). Sistem Pencatatan Dan Pendataan Manajemen Sumber Daya Manusia Dengan Model Scrum (Studi Kasus: Pt Bintang Trans Khatulistiwa). *JSAI (Journal Scientific and Applied Informatics)*, 2(1), 105–115. <https://doi.org/10.36085/jsai.v2i1.138>
- Schwaber, K., & Sutherland, J. (2017). *Panduan Definitif untuk Scrum: Aturan Main*. 2(12), 1–19. <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>
- Wang, V. V., Sukamto, A. S., & Pratama, E. E. (2019). Sistem Pendukung Keputusan Seleksi Mahasiswa Penerima Beasiswa BBP-PPA dengan Metode TOPSIS pada Fakultas Teknik UNTAN. *Jurnal Sistem Dan Teknologi Informasi (JUSTIN)*, 7(2), 105. <https://doi.org/10.26418/justin.v7i2.29656>
- Wardhono, W. S., Kusuma, L. P., & Wardhono, W. S. (2015). Evaluasi User Acceptance Augmented Reality Triage Mobile Pada Sistem Kedaruratan Medis.

Jurnal Sentar, 978–979. <http://research-report.umm.ac.id/index.php/sentra/article/viewFile/2878>

LAMPIRAN

Lampiran 1 Hasil Wawancara

Nama : Aditya Fitriadi (mahasiswa)

Kelas : TI 2017

Pertanyaan Umum:

1. Program Link and Match yang dikerjakan adalah *website* untuk mengelola *sprint* dan *task*
2. Metode yang dikerjakan pada program *link-match* adalah *Scrum*
3. *Tools* yang digunakan pada *link-match*:
 - a. Dokumentasi:
Trello, Google Spreadsheet, Google Docs, Google Drive, Google Form
 - b. *Daily meeting*:
Secara langsung diskusi dan menggunakan Google Form
 - c. Mengetahui perkembangan antar anggota tim dengan whatsApp

A. Modul Absensi

1. Absensi pada proses absensi dengan Google Form walaupun ada dengan *fingerprint* tapi tidak digunakan optimal.
2. Absensi datang pada jam 08.00 dan pulang jam 17.00.
3. Metode absensi dengan Google Form tidak efisien, tidak *real-time* atau tidak akurat karena dapat dengan kecurangan kehadiran dengan Google Form.
4. Untuk melihat kehadiran tidak bisa melihat secara langsung karena tidak dapat akses atau *platform* untuk melihat kehadiran yang telah masuk selama menjalankan program *link-match*.
5. Menurut saya sistem absensi yang lebih baik dengan cara menggunakan *fingerprint* agar lebih akurat karena tidak bisa dimanipulasi atau dengan titip absen ke teman.

B. Modul Mahasiswa

1. Kegiatan yang dilakukan setelah absensi dengan penyampaian materi dari dosen kemudian *daily meeting* dengan membahas pekerjaan yang telah dilakukan, pekerjaan yang akan dilakukan di hari itu dan kendala.
2. Kesulitan selama melakukan *daily meeting* adalah ketika anggota tidak hadir ketika *link-match*, kemudian kesulitan dalam melihat *user story* yang dimaksud ketika melakukan laporan harian dan tidak bisa melihat laporan harian atau sprint di tiap anggota kelompok.
3. Kesulitannya ketika mengelola *task* atau mengontrol tugas pada *user story* karena tidak ada yang fokus dalam mengontrol *task* atau tugas sehingga masing-masing perlu mengelolanya sendiri.
4. Kesulitan dalam dokumentasi adalah karena masih manual dengan Google Spreadsheet.
5. Kesulitan dalam mengetahui perkembangan proyek dari masing-masing anggota tidak bisa melihat laporan perkembangan seperti *daily meeting* atau laporan sprint atau tidak ada platformnya. Sehingga untuk mengetahuinya manual seperti bertanya secara langsung dan sebagainya.

C. Modul *Mobile*

1. Dokumentasi lebih menggunakan laptop seperti *update user story*, Trello dan lainnya karena tidak ada *platform mobile* yang memudahkan dalam pengelolaan dokumentasi dalam menjalankan program *link-match*. Penggunaannya yang masih menggunakan manual (banyak *tools* yang digunakan).
2. Jika ada platform yang dapat mengelola dokumentasi tersebut sangat membantu, nyaman dan memudahkan digunakan karena bisa fleksibel dan lain sebagainya.

D. Modul Umum

1. Kekurangan dari *link-match* dari sistem yang ribet seperti serba manual dan menggunakan banyak *tools* yang digunakan, jika ada platform yang ada dapat meningkatkan proses pembelajaran *link-match*.
 2. Sistem yang perlu ditambahkan satu platform yang satu kesatuan di dalam proses pembelajaran *link-match* untuk bisa integrasi satu dengan yang lainnya sehingga mengurangi *tools* yang digunakan.
-
-

Nama : Jennie Imelia Fitri Sundari (mahasiswi)

Kelas : SI 2017

Pertanyaan Umum:

1. Program Link and Match yang dikerjakan adalah dashboard *link-match* yang menampilkan absensi yang terhubung dengan Google Form, laporan projek dan lainnya.
2. Metode yang dikerjakan pada program *link-match* adalah *Scrum*
3. Tools yang digunakan pada *link-match*:
Trello, Google Spreadsheet, Google Docs, Google Drive, Google Form

A. Modul Absensi

1. Platform yang digunakan adalah Google Form, dimana pengisian secara manual termasuk jam kehadiran dan kepulangan.
2. Jam masuk 08.00 dan jam pulang 17.00.
3. Metode pada absensi dengan Google Form masih kurang baik, kurang efektif karena tidak menghemat waktu dan tenaga karena absensi dibutuhkan secara *real-time* dan mudah digunakan serta tidak bisa di manipulasi.
4. Tidak dapat melihat daftar kehadiran selama program *link-match*.
5. Absensi yang digunakan baiknya dengan *fingerprint* untuk menghemat waktu dan tenaga serta dapat secara *real-time* dengan kehadiran.

B. Modul Mahasiswa

1. Kegiatan yang dilakukan absensi melakukan *daily meeting* dengan me-review kembali tugas yang dikerjakan dan me-review tugas yang akan dikerjakan selanjutnya serta kendala yang dihadapi bersama *scrum master*.
2. Kendala dalam laporan harian dan *sprint* yaitu notulensi yang dilakukan secara manual.
3. Kesulitan ketika mengelola *task* karena suka terlupakan dalam mengelola *task*.
4. Sering terjadi komunikasi antar tim dan perkembangan masing-masing anggota karena seringnya ketidak sesuaian dalam pembagian tugas dan pelaporan yang tidak bisa dilihat oleh semua anggota tim. Sering juga anggota yang tidak melaporkan progress dan tidak ada platform untuk melihat *details* tugas yang mengerjakan itu siapa sehingga ada 1 *task* dikerjakan 2 orang secara bersamaan.

5. Terdapat kendala pada *sprint review* yaitu tidak bisa melihat progress perkembangan *sprint* dari masing-masing anggota tim.

C. Modul *Mobile*

1. Dokumentasi *link-match* baik juga dengan *device mobile* untuk melihat perkembangan atau mengontrol proyek lebih fleksibel sehingga bisa dilakukan dimanapun dan kapanpun.
 2. Program *link-match* perlu ditingkatkan terutama dalam *platform* yang bisa integrasi dengan semua dokumentasi yang dikerjakan selama *link-match* dan *real-time* sehingga tidak memerlukan banyak *tools* untuk pendokumentasian *link-match*.
-
-

Nama : Justis Aulia Pratomo (mahasiswa)

Kelas : TI 2018

Pertanyaan Umum:

1. Program Link and Match yang dikerjakan adalah fitur tambahan untuk AIS yaitu absensi
2. Metode yang dikerjakan pada program *link-match* adalah *Scrum*
3. *Tools* yang digunakan pada *link-match*:
Trello, Google Spreadsheet, dan aplikasi pihak ketiga lainnya

A. Modul Absensi

1. Jadwal *link-match* sesuai jam mata kuliah.
2. Sistem absensi yang dilakukan dengan Google Form.
3. Untuk presensi cukup untuk menggunakan Google Form, walaupun ada yang telat mengisi otomatis tidak hadir karena ada waktu untuk pengaksesan Google Form dan kode rahasianya.
4. Untuk *check* absensi selama *link-match* tidak bisa harus izin ke *scrum master* atau dirilis absensi di pertengahan semester dan di akhir.
5. Sistem absensi untuk program *link-match* lebih baik dengan *tools* formulir lewat Google sudah efektif. Tetapi jika ada platform absensi dengan *fingerprint* mengurangi kecurangan absensi dan lebih efisien.

B. Modul Mahasiswa

1. Setelah absensi kemudian melakukan *daily meeting* yang membahas *requirement* yang perlu dilengkapi jika kurang jelas maka bisa hubungi

- lagi ke *scrum master* atau *product owner* dan diskusi dengan anggota tim.
2. *Tools* yang digunakan untuk laporan harian itu dengan grup WhatsApp dan rekapan laporan dengan Google Spreadsheet.
 3. Kesulitan dalam menggunakan WhatsApp untuk laporan dengan sistem lainnya sama saja kendalanya seperti jaringannya dan diskusinya sangat kurang.
 4. Google Spreadsheet menggunakan progress proyek yang dapat dilihat dosen dan *scrum master*.
 5. Cukup untuk *tools* yang digunakan dalam dokumentasi tapi untuk mengetahui progress tim mengalami kesulitan apalagi dengan sistem *online*.

C. Modul *Mobile*

1. Untuk pengerjaan proyek lebih dominan laptop dan *mobile*, untuk pengerjaan proyek dengan laptop dan laporan proyek menggunakan *smartphone*.
 2. Untuk program *link-match* sudah baik, paling kendalanya karna *online* sehingga kurangnya komunikasi.
 3. Untuk sistem konsep *link-match* sudah baik, tetapi saran untuk *link-match* lebih baik *offline* karena kalo *online* lebih banyak kendala *missed* komunikasi.
-
-

Nama : Muhammad Ibnu Al-Fida (mahasiswa)

Kelas : SI 2018

Pertanyaan Umum:

1. Program Link and Match yang dikerjakan adalah membuat *startup*
2. Metode yang dikerjakan pada program *link-match* adalah *Waterfall*
3. *Tools* yang digunakan pada *link-match*:
Figma dan *tools* desainnya lainnya, Google Drive

A. Modul Absensi

1. *Tools* absensi dengan manual yaitu panggilan dari dosen.

2. Absensi mengikuti waktu kuliah yang sesuai jadwal karena ada beberapa teman yang tidak mengikuti mata kuliah terkait program pra *link-match*.
3. Metode *fingerprint* jika ada lebih mudah dan simpel tanpa harus menunggu waktu kehadiran dosen terkait mata kuliah untuk hadir terlebih dahulu.
4. Tidak bisa melihat *update* absensi karna tidak *real-time* dari AIS dengan absensi metode manual ini.

B. Modul Mahasiswa

1. Tidak mengalami kesulitan dalam membuat laporan progress kepada dosen.
2. Sebelum mengerjakan *task* atau tugas perlu mengelola tugas yang akan dikerjakan baru di lakukan *task* yang telah diatur.
3. Kesulitan dalam mengerjakan *task* adalah pengetahuan, sarana pra *link-match* yang belum mumpuni untuk pengelolaan *tasknya*.
4. Kesulitan ketika berdiskusi antar anggota tim diantaranya komunikasi, kesibukan masing-masing anggota.

C. Modul *Mobile*

1. Untuk penggunaan *device* dalam program *link-match* lebih sering menggunakan laptop.
2. Untuk konsepnya cukup baik hanya saja pra *link-match* sering terjadi *missed* komunikasi antar dosen terkait aplikasi yang digunakan dalam program pra *link-match*. Tetapi secara keseluruhan sudah baik.
3. Konsep yang cocok untuk program *link-match* nanti ditambahkan absensi yang lebih mudah, desain ruangan untuk penggerjaan proyeknya. Untuk konsep keseluruhan sudah baik tinggal fasilitas *link-match* yang lebih baik.
4. Saran dari saya fasilitas adalah *platform* yang dapat terintegrasi antara dosen dan mahasiswa dalam menjalankan program *link-match* nanti.

Nama : Pak Sirojul Munir (dosen)

Tambahan Fitur Modul Dosen:

1. Tambahkan *dashboard*
2. *Product owner* ditambahkan di dalam modul dosen
3. *Product owner* cukup melihat *project*
4. Akun yang berbeda walaupun orangnya sama (dosen dan *product owner* walau sama tapi memiliki dua akun untuk masing-masing *role*)
5. Tampilan informasi *project* yang lengkap beserta tim yang mengerjakan dan jumlah sprint

Lampiran 2

User Acceptance Test

Penguji: Pak Rojul (*product owner*)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	Login user		User dapat menampilkan halaman <i>login</i>	✓	
		Username: productowner1 Password: password	User dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		User dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		User berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		Form login	User dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	View profile	User dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	
		User dapat menampilkan halaman <i>profile</i>	✓	

3	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	✓	
		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project dan profile user</i>	✓	Tampilan UI nya iconnya kecil, baiknya proporsional
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	
		<i>User dapat menampilkan halaman list project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	

Penguji: Bu Tifani (dosen)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	✓	Loading cukup lama
		<i>Username:</i> dosen1 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	Sudah cukup baik
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	
3	<i>Logout user</i>	<i>User</i> dapat mengklik <i>icon logout</i> pada halaman <i>dashboard</i> untuk proses <i>logout</i>	✓	Baik

		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project, sprint, tim, mahasiswa dan profile user</i>	✓	Loading cukup lama
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	Baik
		<i>User dapat menampilkan halaman project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	✓	Belum bisa edit dan selesai sprint
		<i>User dapat menampilkan halaman sprint</i>	✓	
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	✓	
7	<i>View tim</i>	<i>User dapat memilih menu tim pada halaman dashboard</i>	✓	Harusnya dapat melihat nama anggota tim

		<i>User</i> dapat menampilkan halaman tim	✓	
		<i>User</i> dapat menampilkan halaman tim yang dipilih	✓	
		<i>User</i> dapat mengklik <i>list button</i> untuk menampilkan halaman nilai tim	✓	
8	<i>View mahasiswa</i>	<i>User</i> dapat memilih menu mahasiswa pada halaman <i>dashboard</i>	✓	<i>User</i> akan mengira nilai bisa dilihat dengan langsung mengklik nama mahasiswa di dashboard
		<i>User</i> dapat menampilkan halaman mahasiswa	✓	
		<i>User</i> dapat mengklik <i>list button</i> untuk menampilkan halaman nilai mahasiswa	✓	
9	<i>View nilai</i>	<i>User</i> dapat menampilkan halaman nilai tim	✓	Nilai mahasiswa kalau tidak bisa dilihat dari <i>icon</i> mahasiswa (yg di- <i>list</i>), langsung saja tampilkan daftar nilainya di halaman awal
		<i>User</i> dapat menampilkan halaman nilai mahasiswa	✓	

Penguji: Pak Hilmy (dosen)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	✓	Loading agak lama, perlu ada fasilitas <i>view password</i>
		<i>Username:</i> dosen2 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	Data belum muncul semua (foto). Tidak ada fasilitas <i>update profile</i>
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	

3	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	✓	Baik
		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project, sprint, tim, mahasiswa dan profile user</i>	✓	<i>Font dan icon kurang proporsional. Design dibuat lebih rapih</i>
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	<i>Font terlalu besar, tidak perlu ada tulisan lihat detail. Cukup di klik saja list-nya. Bisa juga ditambahkan info detail seperti project owner dan durasi pada list, sehingga tidak hanya nama project</i>
		<i>User dapat menampilkan halaman project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	✓	Dikelompokkan berdasarkan project.

		<i>User dapat menampilkan halaman sprint</i>	✓	Atau ada fasilitas <i>filter</i> berdasarkan <i>project</i> . Menu <i>edit</i> ada, tapi tidak dapat <i>edit</i> dan <i>delete</i> . Atau <i>loading</i> agak lama?
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	✓	Tidak ada <i>progress/loading spinner</i> dan notifikasi sehingga tidak tahu apakah proses berhasil/tidak
7	View tim	<i>User dapat memilih menu tim pada halaman dashboard</i>	✓	UI dipercantik, <i>scrum master/ketua kelompok</i> ditampilkan di <i>list</i> . Untuk lihat <i>detail</i> cukup di <i>click</i> , tidak perlu "lihat <i>detail</i> "
		<i>User dapat menampilkan halaman tim</i>	✓	
		<i>User dapat menampilkan halaman tim yang dipilih</i>	✓	
		<i>User dapat mengklik list button untuk menampilkan halaman nilai tim</i>	✓	

8	View mahasiswa	<i>User dapat memilih menu mahasiswa pada halaman dashboard</i>	✓	Halaman mahasiswa gunanya apa ya? tidak ada info penting dan tidak bisa diapa-apakan. Halaman nilai mahasiswa lebih informatif
		<i>User dapat menampilkan halaman mahasiswa</i>	✓	
		<i>User dapat mengklik <i>list button</i> untuk menampilkan halaman nilai mahasiswa</i>	✓	
9	View nilai	<i>User dapat menampilkan halaman nilai tim</i>	✓	
		<i>User dapat menampilkan halaman nilai mahasiswa</i>	✓	

Penguji: Pak Nasrul (dosen)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	✓	Baik
		<i>Username:</i> dosen2 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	Tampilkan foto dosen
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	
3	<i>Logout user</i>	<i>User</i> dapat mengklik <i>icon logout</i> pada halaman <i>dashboard</i> untuk proses <i>logout</i>	✓	Baik

		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project, sprint, tim, mahasiswa dan profile user</i>	✓	Baik
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	Baik
		<i>User dapat menampilkan halaman project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	✓	Baik
		<i>User dapat menampilkan halaman sprint</i>	✓	
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	✓	
7	<i>View tim</i>	<i>User dapat memilih menu tim pada halaman dashboard</i>	✓	Baik

		<i>User dapat menampilkan halaman tim</i>	✓	
		<i>User dapat menampilkan halaman tim yang dipilih</i>	✓	
		<i>User dapat mengklik <i>list button</i> untuk menampilkan halaman nilai tim</i>	✓	
8	<i>View mahasiswa</i>	<i>User dapat memilih menu mahasiswa pada halaman <i>dashboard</i></i>	✓	Baik
		<i>User dapat menampilkan halaman mahasiswa</i>	✓	
		<i>User dapat mengklik <i>list button</i> untuk menampilkan halaman nilai mahasiswa</i>	✓	
9	<i>View nilai</i>	<i>User dapat menampilkan halaman nilai tim</i>	✓	Baik
		<i>User dapat menampilkan halaman nilai mahasiswa</i>	✓	

Penguji: Bu Ami (dosen)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>	<i>User</i> dapat menampilkan halaman <i>login</i>	✓	
		<i>Username:</i> dosen2 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	

3	<i>Logout user</i>	<i>User</i> dapat mengklik <i>icon logout</i> pada halaman <i>dashboard</i> untuk proses <i>logout</i>	✓	Kenapa harus ke pengujian <i>logout</i> sebelum ke pengujian menu lainnya?
		<i>User</i> dapat menampilkan halaman <i>login</i> setelah proses <i>logout</i> selesai	✓	
4	<i>Dashboard</i>	<i>User</i> dapat menampilkan halaman <i>dashboard</i> yaitu manajemen <i>project</i> , <i>sprint</i> , tim, mahasiswa dan <i>profile user</i>	✓	
5	<i>View project</i>	<i>User</i> dapat memilih menu <i>project</i> pada halaman <i>dashboard</i>	✓	Baik
		<i>User</i> dapat menampilkan halaman <i>project</i>	✓	
		<i>User</i> dapat menampilkan halaman <i>project</i> yang dipilih	✓	
6	<i>View sprint</i>	<i>User</i> dapat memilih menu <i>sprint</i> pada halaman <i>dashboard</i>	✓	Informasi <i>sprint</i> menempel pada <i>project</i> yg mana? Tidak ada di <i>detail sprint</i> ?
		<i>User</i> dapat menampilkan halaman <i>sprint</i>	✓	
		<i>User</i> dapat menampilkan halaman <i>detail sprint</i> yang dipilih	✓	

7	View tim	User dapat memilih menu tim pada halaman <i>dashboard</i>	✓	Informasi nama-nama anggota tim tidak ada dan <i>project</i> yang mana juga tidak ditampilkan
		User dapat menampilkan halaman tim	✓	
		User dapat menampilkan halaman tim yang dipilih	✓	
		User dapat mengklik <i>list button</i> untuk menampilkan halaman nilai tim	✓	
8	View mahasiswa	User dapat memilih menu mahasiswa pada halaman <i>dashboard</i>	✓	Mahasiswa berelasi sama <i>project</i> apa?
		User dapat menampilkan halaman mahasiswa	✓	
		User dapat mengklik <i>list button</i> untuk menampilkan halaman nilai mahasiswa	✓	
9	View nilai	User dapat menampilkan halaman nilai tim	✓	Ditulis di judul “Lihat Nilai Tim”
		User dapat menampilkan halaman nilai mahasiswa	✓	

Penguji: Hamdan Ainul Atmam Al Faruq (*scrum master*)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>		<i>User</i> dapat menampilkan halaman <i>login</i>	✓	
		<i>Username:</i> scrummaster2 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	

3	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	✓	
		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project, sprint, tim, mahasiswa dan profile user</i>	✓	
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	
		<i>User dapat menampilkan halaman project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	✓	<i>Update daftar sprint terlalu lama</i>
		<i>User dapat menampilkan halaman sprint</i>	✓	
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	✓	

			<i>User</i> dapat mengklik <i>add button</i> untuk menampilkan halaman tambah <i>sprint</i>	✓	
			<i>User</i> dapat mengklik <i>delete button</i> untuk menampilkan <i>popup hapus sprint</i>	✓	
7	<i>Add sprint</i>	<i>User</i> dapat menampilkan <i>form</i> tambah <i>sprint</i>	✓	- pesan error yang tampil sebatas tidak mengisi <i>fill</i> - tidak ada penjelasan format tanggal - tidak ada pesan kesalahan ketika format gambar di isi sembarang.
		<i>Form add sprint</i> (namaSprint, projectId, tglMulai, tglAkhir)	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai	✗	- tombol memproses meski yang di isi sembarang
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses tambah <i>sprint</i>	✓	Meskipun tidak ter-update, ini tidak <i>user friendly</i>

8	Memulai <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses mulai <i>sprint</i>	✓	Di awal saya pikir gagal, status tidak segera ter-update
9	Mengakhiri <i>sprint</i>	<i>User</i> dapat mengklik <i>submit button</i> untuk proses akhiri <i>sprint</i>	✓	Di awal saya pikir gagal, status tidak segera ter-update
10	<i>Edit sprint</i>	<i>User</i> dapat menampilkan <i>form edit sprint</i>	✓	Hampir sama dengan yang tambah
		<i>Form update sprint</i> (namaSprint, projectID, tglMulai, tglAkhir)	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai	✗	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses <i>edit sprint</i>	✓	
11	<i>Delete sprint</i>	<i>User</i> dapat melihat <i>pop-up</i> hapus <i>sprint</i>	✓	Data yang ter-delete tidak segera ter-update
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses hapus <i>sprint</i>	✓	
		<i>User</i> dapat mengklik <i>back button</i> untuk membatalkan hapus <i>sprint</i>	✓	
12	<i>View tim</i>	<i>User</i> dapat memilih menu tim pada halaman <i>dashboard</i>	✓	

		<i>User dapat menampilkan halaman tim</i>	✓	
		<i>User dapat menampilkan halaman tim yang dipilih</i>	✓	
13	<i>View mahasiswa</i>	<i>User dapat memilih menu mahasiswa pada halaman dashboard</i>	✓	
		<i>User dapat menampilkan halaman mahasiswa</i>	✓	

Penguji: Annisa Tahira (*scrum master*)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>		<i>User</i> dapat menampilkan halaman <i>login</i>	✓	
		<i>Username:</i> scrummaster2 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	

3	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	✓	
		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project, sprint, tim, mahasiswa dan profile user</i>	✓	
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	
		<i>User dapat menampilkan halaman project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	✓	
		<i>User dapat menampilkan halaman sprint</i>	✓	
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	✓	

			<i>User dapat mengklik add button untuk menampilkan halaman tambah sprint</i>	✓	
			<i>User dapat mengklik delete button untuk menampilkan pop-up hapus sprint</i>	✓	
7	<i>Add sprint</i>	<i>User dapat menampilkan form tambah sprint</i>	✓	Pesan <i>error</i> tidak muncul ketika saya tidak memilih project
		<i>Form add sprint (namaSprint, projectID, tglMulai, tglAkhir)</i>	<i>User dapat melihat pesan <i>error</i> jika form tidak sesuai</i>	✓	
		<i>User dapat mengklik submit button untuk proses tambah sprint</i>	✓	
8	<i>Memulai sprint</i>	<i>User dapat mengklik submit button untuk proses mulai sprint</i>	✓	

9	Mengakhiri sprint	User dapat mengklik <i>submit button</i> untuk proses akhiri <i>sprint</i>	✓	
10	<i>Edit sprint</i>	User dapat menampilkan <i>form edit sprint</i>	✓	Pesan <i>error</i> tidak muncul ketika saya tidak memilih <i>project</i>
		<i>Form update sprint</i> (namaSprint, projectID, tglMulai, tglAkhir)	User dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai	✗	
		User dapat mengklik <i>submit button</i> untuk proses <i>edit sprint</i>	✓	
11	<i>Delete sprint</i>	User dapat melihat <i>pop-up hapus sprint</i>	✓	
		User dapat mengklik <i>submit button</i> untuk proses <i>hapus sprint</i>	✓	
		User dapat mengklik <i>back button</i> untuk membatalkan <i>hapus sprint</i>	✓	
12	<i>View tim</i>	User dapat memilih menu tim pada halaman <i>dashboard</i>	✓	
		User dapat menampilkan halaman tim	✓	
		User dapat menampilkan halaman tim yang dipilih	✓	

13	<i>View mahasiswa</i>	<i>User</i> dapat memilih menu mahasiswa pada halaman <i>dashboard</i>	✓	
		<i>User</i> dapat menampilkan halaman mahasiswa	✓	

Penguji: Syaiful Ramadhan (*scrum master*)

No.	Pengujian	Data Masukan	Hasil yang diharapkan	Hasil	Komentar
1	<i>Login user</i>		<i>User</i> dapat menampilkan halaman <i>login</i>	✓	Tambahkan notifikasi jika gagal karena tidak ada koneksi, bukan hanya <i>user</i> dan <i>password</i> yang salah
		<i>Username:</i> scrummaster2 <i>Password:</i> password	<i>User</i> dapat mengetikkan <i>username</i> dan <i>password</i>	✓	
		<i>User</i> dapat mengklik <i>submit button</i> untuk proses validasi	✓	
		<i>User</i> berhasil <i>login</i> , menampilkan <i>splash-screen</i> dan halaman <i>dashboard</i>	✓	
		<i>Form login</i>	<i>User</i> dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai atau validasi gagal	✓	
2	<i>View profile</i>	<i>User</i> dapat memilih menu <i>profile</i> pada halaman <i>dashboard</i>	✓	Tambahkan fitur <i>edit profile</i>
		<i>User</i> dapat menampilkan halaman <i>profile</i>	✓	

3	<i>Logout user</i>	<i>User dapat mengklik icon logout pada halaman dashboard untuk proses logout</i>	✓	
		<i>User dapat menampilkan halaman login setelah proses logout selesai</i>	✓	
4	<i>Dashboard</i>	<i>User dapat menampilkan halaman dashboard yaitu manajemen project, sprint, tim, mahasiswa dan profile user</i>	✓	
5	<i>View project</i>	<i>User dapat memilih menu project pada halaman dashboard</i>	✓	Jika bisa kolom sprint bisa langsung link ke menu sprint
		<i>User dapat menampilkan halaman project</i>	✓	
		<i>User dapat menampilkan halaman project yang dipilih</i>	✓	
6	<i>View sprint</i>	<i>User dapat memilih menu sprint pada halaman dashboard</i>	✓	Tidak bisa lihat daftar sprint langsung, harus ke dashboard dulu baru balik lagi ke menu sprint untuk melihat sprint yang ditambahkan.
		<i>User dapat menampilkan halaman sprint</i>	✓	
		<i>User dapat menampilkan halaman detail sprint yang dipilih</i>	✓	

			<i>User dapat mengklik add button untuk menampilkan halaman tambah sprint</i>	✓	Sebaiknya ketika disimpan halaman <i>sprint</i> langsung <i>refresh</i> otomatis
			<i>User dapat mengklik delete button untuk menampilkan pop-up hapus sprint</i>	✗	
7	<i>Add sprint</i>	<i>User dapat menampilkan form tambah sprint</i>	✓	
		<i>Form add sprint (namaSprint, projectId, tglMulai, tglAkhir)</i>	<i>User dapat melihat pesan error jika form tidak sesuai</i>	✗	
		<i>User dapat mengklik submit button untuk proses tambah sprint</i>	✓	
8	<i>Memulai sprint</i>	<i>User dapat mengklik submit button untuk proses mulai sprint</i>	✓	

9	Mengakhiri sprint	User dapat mengklik <i>submit button</i> untuk proses akhiri <i>sprint</i>	✓	
10	<i>Edit sprint</i>	User dapat menampilkan <i>form edit sprint</i>	✓	Pesan <i>error</i> tidak muncul ketika saya tidak memilih <i>project</i>
		<i>Form update sprint</i> (namaSprint, projectID, tglMulai, tglAkhir)	User dapat melihat pesan <i>error</i> jika <i>form</i> tidak sesuai	✓	
		User dapat mengklik <i>submit button</i> untuk proses <i>edit sprint</i>	✓	
11	<i>Delete sprint</i>	User dapat melihat <i>pop-up hapus sprint</i>	✓	
		User dapat mengklik <i>submit button</i> untuk proses hapus <i>sprint</i>	✗	
		User dapat mengklik <i>back button</i> untuk membatalkan hapus <i>sprint</i>	✓	
12	<i>View tim</i>	User dapat memilih menu tim pada halaman <i>dashboard</i>	✓	
		User dapat menampilkan halaman tim	✓	
		User dapat menampilkan halaman tim yang dipilih	✓	

13	<i>View mahasiswa</i>	<i>User</i> dapat memilih menu mahasiswa pada halaman <i>dashboard</i>	✓	
		<i>User</i> dapat menampilkan halaman mahasiswa	✓	