



PROPOSAL TUGAS AKHIR - IF184702

PENGUJIAN OTOMATIS TERHADAP SITUS WEB MYITS THESIS MENGGUNAKAN KATALON STUDIO

VYRA FANIA ADELINA

NRP 05111940000109

Dosen Pembimbing

Ir. Siti Rochimah, MT., Ph.D.

NIP 196810021994032001

Rizky Januar Akbar, S.Kom., M.Eng.

NIP 198701032014041001

HALAMAN JUDUL

Program Studi S1 Teknik Informatika

Departemen Teknik Informatika

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya

2022

LEMBAR PENGESAHAN

PENGUJIAN OTOMATIS TERHADAP SITUS WEB MYITS THESIS MENGUNAKAN KATALON STUDIO

PROPOSAL TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat
Memperoleh gelar Sarjana Komputer pada
Program Studi S-1 Teknik Informatika
Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Oleh : **VYRA FANIA ADELINA**

NRP. 05111940000109

Disetujui oleh Tim Penguji Proposal Tugas Akhir:

- | | | |
|----|------------------------------------|---------------|
| 1. | Ir. Siti Rochimah, MT., Ph.D. | Pembimbing |
| 2. | Rizky Januar Akbar, S.Kom., M.Eng. | Ko-pembimbing |

**SURABAYA
Desember, 2022**

APPROVAL SHEET

AUTOMATION TESTING OF MYITS THESIS WEBSITE USING KATALON STUDIO

FINAL PROJECT PROPOSAL

Submitted to fulfill one of the requirements
for obtaining a degree Bachelor of Computer Science at
Undergraduate Study Program of Informatics Engineering
Department of Informatics Engineering
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember

By: **VYRA FANIA ADELINA**

NRP. 05111940000109

Approved by Final Project Proposal Examiner Team:

- | | | |
|----|------------------------------------|------------|
| 1. | Ir. Siti Rochimah, MT., Ph.D | Advisor |
| 2. | Rizky Januar Akbar, S.Kom., M.Eng. | Co-Advisor |

**SURABAYA
December, 2022**

PENGUJIAN OTOMATIS TERHADAP SITUS WEB MYITS THESIS MENGUNAKAN KATALON STUDIO

Nama Mahasiswa / NRP : Vyra Fania Adelina / 05111940000109
Departemen : Teknik Informatika FTEIC - ITS
Dosen Pembimbing I : Ir. Siti Rochimah, MT., Ph.D.
Dosen Pembimbing II : Rizky Januar Akbar, S.Kom., M.Eng.

ABSTRAK

Abstrak

Direktorat Pengembangan Teknologi dan Sistem Informasi (DPTSI) ITS mengembangkan salah satu sistem yang mempermudah mahasiswa dan dosen dalam proses pelaksanaan Tugas Akhir/Tesis/Disertasi, bernama myITS Thesis. Untuk melakukan pengembangan dan pemeliharaan perangkat lunak berjangka panjang, dibutuhkan pengujian otomatis. Dengan adanya pengujian otomatis tersebut, perangkat lunak dapat diuji secara berkala dan menghasilkan perangkat lunak yang baik kepada pengguna. Selain itu, terdapat proses integrasi pengujian otomatis pada *CI/CD pipeline* untuk mempermudah dan mempercepat proses perilis perangkat lunak yang cepat dan aman bagi pengembang. Dalam proses implementasi pengujian otomatis, digunakan alat bantu Katalon Studio untuk pembangunan pengujian otomatis fungsional antarmuka pengguna serta Jenkins untuk mengintegrasikan pengujian otomatis yang dapat mempercepat pengembangan sistem. Dengan begitu, akan dihasilkan implementasi terbaik untuk pengujian otomatis sistem myITS Thesis serta bagaimana cara melakukan pemeliharaan perangkat lunak dalam jangka panjang.

Kata kunci: DPTSI, Jenkins, Katalon, Pengembangan, Pengujian, Otomatis.

AUTOMATION TESTING OF MYITS THESIS WEBSITE USING KATALON STUDIO

Student Name / NRP : Vyra Fania Adelina / 05111940000109
Department : Informatics Engineering FTEIC - ITS
Advisor I : Ir. Siti Rochimah, MT., Ph.D.
Advisor II : Rizky Januar Akbar, S.Kom., M.Eng.

Abstract

The Directorate of Technology and Information System Development (DPTSI) ITS, currently developing a system to ease the process of Final Project/Thesis/Disertation for the students and professors, called myITS Thesis. To carry out long-term software development and maintenance, the implementation of software automation testing is required. With the implementation, software can be tested regularly and produce good quality software for users. In addition, integration process of automated testing in the CI/CD pipeline is needed to simplify and accelerate the process of software release for developers. In the process of implementing automated testing, Katalon Studio is used for the functional automation testing of the user interface and Jenkins to integrate automated testing which can speed up the system development. With that, the best implementation of automation testing for myITS Thesis will be produced as well as how to maintain the software in long term.

Kata kunci: Automation, Developer, DPTSI, Jenkins, Katalon, Testing.

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	ii
ABSTRAK	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL.....	x
BAB I PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan Permasalahan.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	2
1.5 Manfaat.....	2
BAB II TINJAUAN PUSTAKA.....	3
2.1 Penelitian Terkait.....	3
2.2 Dasar Teori	3
2.2.1 <i>User Story</i>	3
2.2.2 Kasus Pengujian	4
2.2.3 <i>Functional Testing</i>	4
2.2.4 <i>Automation Testing</i>	4
2.2.5 <i>Continuous Integration & Continuous Delivery (CI/CD)</i>	4
2.2.6 Katalon Studio	5
2.2.7 Jenkins	5
BAB III METODOLOGI.....	7
3.1 Metode yang dirancang	7
3.1.1 Analisis Situs Web myITS Thesis Sebagai Studi Kasus	8
3.1.1.1 Analisis Permasalahan	8
3.1.1.2 Deskripsi Umum Sistem	8
3.1.1.3 Aktor/ <i>User</i>	9
3.1.1.4 Fitur.....	11
3.1.2 Perancangan.....	29
3.1.2.1 Perancangan Kasus Pengujian	29
3.1.3 Implementasi	31
3.1.3.1 Pembuatan Pengujian Otomatis	31
3.1.3.2 Integrasi Pengujian Otomatis pada CI/CD pipeline	32

3.2	Peralatan pendukung.....	33
3.3	Rencana Implementasi dan Uji Coba	34
JADWAL KEGIATAN.....		35
DAFTAR PUSTAKA		36

DAFTAR GAMBAR

Gambar 2.2.1 Alur kerja CI/CD pipeline	5
Gambar 3.1.1 Alur pengujian otomatis MyITS Thesis	7
Gambar 3.1.2 Use Case diagram sistem myITS Thesis	9
Gambar 3.1.3 Tampilan menu manajemen tugas akhir mahasiswa	12
Gambar 3.1.4 Tampilan menu usulan tugas akhir	13
Gambar 3.1.5 Tampilan detail tugas akhir	14
Gambar 3.1.6 Tampilan fitur tugas akhir administrator	15
Gambar 3.1.7 Tampilan Halaman detail tugas akhir administrator.....	15
Gambar 3.1.8 Tampilan halaman Kelola periode sidang administrator	16
Gambar 3.1.9 Tampilan halaman detail periode sidang administrator	16
Gambar 3.1.10 Tampilan halaman sidang dosen.....	17
Gambar 3.1.11 Tampilan halaman daftar ajuan sidang	18
Gambar 3.1.12 Tampilan halaman detail ajuan sidang dosen penguji	18
Gambar 3.1.13 Tampilan halaman daftar revisi	18
Gambar 3.1.14 Tampilan halaman detail revisi penelitian	19
Gambar 3.1.15 Tampilan halaman jadwal sidang mahasiswa.....	20
Gambar 3.1.16 Tampilan halaman detail ajuan sidang mahasiswa	20
Gambar 3.1.17 Tampilan halaman revisi tugas akhir mahasiswa	21
Gambar 3.1.18 Tampilan popup detail revisi mahasiswa.....	21
Gambar 3.1.19 Tampilan halaman pembimbingan dosen administrator	22
Gambar 3.1.20 Tampilan halaman perubahan kuota bimbingan.....	22
Gambar 3.1.21 Tampilan Halaman Bimbingan Saya dosen pembimbing	23
Gambar 3.1.22 Tampilan daftar tugas akhir bimbingan dosen.....	24
Gambar 3.1.23 Tampilan detail tugas akhir dosen pembimbing	24
Gambar 3.1.24 Tampilan halaman detail logbook progress dosen pembimbing	25

Gambar 3.1.25 Tampilan halaman detail usulan perubahan dosen	25
Gambar 3.1.26 Tampilan popup form penilaian sidang	26
Gambar 3.1.27 Tampilan Popup Berikan Revisi dosen.....	26
Gambar 3.1.28 Tampilan halaman penawaran topik dosen.....	27
Gambar 3.1.29 Tampilan halaman membuat penawaran topik	28
Gambar 3.1.30 Tampilan halaman detail penawaran topik	28
Gambar 3.1.31 Tampilan halaman penawaran topik mahasiswa	29
Gambar 3.1.32 Alur Pengujian pada Jenkins	33

DAFTAR TABEL

Tabel 3.1.1 Aktor dan hak akses pada myITS Thesis	10
Tabel 3.1.2 Tabel Kasus Pengujian	29

BAB I PENDAHULUAN

1.1 Latar belakang

Direktorat Pengembangan Teknologi dan Sistem Informasi (DPTSI) ITS saat ini sedang mengembangkan sistem yang dapat mendigitalisasi proses manajemen tugas akhir dan memudahkan perantara antara dosen pembimbing dan mahasiswa, bernama myITS Thesis. Tentunya, dalam proses pengembangan perangkat lunak tersebut tidaklah mudah, dalam mengembangkan suatu perangkat lunak terdapat proses dan metode tertentu yang dilakukan oleh tim pengembang dalam daur hidup pengembangan perangkat lunak (*Software Development Life Cycle*, disingkat SDLC). SDLC sendiri merupakan suatu metodologi yang digunakan untuk mengembangkan sistem rekayasa perangkat lunak dan proses untuk membuat, mendesain, mengubah, dan menguji sistem serta model agar sesuai dengan bisnis yang diinginkan (Ragunath et al, 2010). Dari berbagai fase yang dilalui, pengujian perangkat lunak merupakan salah satu fase yang penting dan memiliki peran vital dalam suatu pengembangan perangkat lunak.

Pengujian perangkat lunak merupakan suatu rangkaian kegiatan yang dilakukan untuk menemukan kesalahan dan mengukur kualitas pada perangkat lunak tersebut (Tuteja & Dubey, 2012). Perangkat lunak dinyatakan lolos verifikasi dan validasi proses pengujian jika memenuhi persyaratan yang telah disepakati dalam perancangan pengembangan perangkat lunak dan bekerja seperti yang diharapkan sesuai dengan kebutuhan yang ditetapkan. Terdapat berbagai macam tipe pengujian, diantaranya adalah pengujian manual dan pengujian otomatis. Pengembangan myITS Thesis telah hampir mencapai tahap akhirnya. Setelah menguji kualitasnya secara manual, myITS Thesis siap untuk dirilis dan digunakan oleh mahasiswa dan dosen ITS. Tetapi, untuk pengembangan dan pemeliharaan jangka panjang ada beberapa hal yang harus ditambahkan, salah satunya merupakan dirancangnya dan diterapkannya pengujian otomatis yang dapat membantu, tidak hanya tim penjaminan mutu, tetapi juga tim pengembang.

Pengujian otomatis yang dikembangkan akan menguji kasus yang krusial seperti fitur masuk akun, keluar akun atau melakukan pencarian. Dengan adanya pengujian otomatis tersebut, tim penjaminan mutu dapat fokus untuk menguji fitur yang sedang dikembangkan. Pengujian otomatis yang tereksekusi akan secara otomatis menguji sesuai dengan kasus uji pada program lalu menemukan kesalahan yang terjadi sebelum dilakukan perilisan. Jika hasil yang didapatkan pada pengujian ‘gagal’ maka hasil perubahan tersebut tidak layak untuk dilanjutkan ke tahap setelahnya yaitu, perilisan. Hal tersebut dapat memudahkan tim pengembang dalam melakukan pemeliharaan pada sistem, seperti saat memperbaiki kerusakan, meningkatkan kinerja sistem, ataupun mengembangkan fitur baru.

Dalam pembuatan pengujian otomatis terdapat berbagai alat bantu gratis yang dapat digunakan, salah satunya merupakan Katalon Studio. Selain memiliki komunitas yang besar, Katalon Studio juga mendukung pengujian otomatis pada situs web dengan menggunakan bahasa Java (Prakoso & Sujarwo, 2022). Pengujian otomatis yang telah dibuat akan diintegrasikan dengan *pipeline* yang mengakibatkan terjadinya perubahan dan penambahan

alur kerja pada bagian CI/CD (*Continuous Integration – Continuous Delivery*). Penggunaan CI/CD sendiri dapat mempercepat dan memudahkan perilisan perangkat lunak yang dikembangkan oleh pengembang.

1.2 Rumusan Permasalahan

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana proses pembuatan pengujian otomatis pada sistem myITS Thesis?
2. Bagaimana cara mengintegrasikan pengujian otomatis pada CI/CD *pipeline* sistem myITS Thesis?
3. Bagaimana mengevaluasi hasil pengujian otomatis?
4. Bagaimana cara tim pengembang melakukan pemeliharaan dan pengembangan fitur baru pada sistem myITS Thesis dalam jangka panjang?

1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan antara lain:

1. Proses pelaksanaan pengujian otomatis, dilakukan dari tahap pembuatan kasus uji, pembuatan skenario uji, perancangan dan pengimplementasian pengujian otomatis menggunakan katalon, hingga pengintegrasian kode pada CI/CD.
2. Pengujian otomatis dibatasi hanya pada modal Admin, Beranda, DosenPembimbing, Pembimbing, Tesis, dan *Setting*.
3. Pengujian otomatis fokus pada pengujian fungsional sistem, termasuk *Create, Read, Update, dan Delete* (CRUD).

1.4 Tujuan

Tujuan dari pembuatan tugas akhir ini adalah untuk melakukan pengujian otomatis terhadap situs web myITS Thesis menggunakan alat bantu Katalon Studio dan mengimplementasikannya pada CI/CD *pipeline* untuk pemeliharaan perangkat lunak jangka panjang.

1.5 Manfaat

Hasil pembuatan tugas akhir ini memiliki manfaat khususnya pada DPTSI untuk memudahkan tim pengembang dalam mengembangkan dan memelihara situs web myITS Thesis dalam jangka panjang.

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Dalam proses pengerjaan tugas akhir ini, dilakukan pencarian penelitian-penelitian yang sudah dilakukan untuk menjelaskan posisi penelitian ini. Terdapat beberapa penelitian terdahulu mengenai pengujian otomatis terhadap perangkat lunak menggunakan Katalon Studio sebagai referensi untuk membantu proses pengerjaan tugas akhir ini.

Seperti pada jurnal tugas akhir yang dilakukan dengan judul **'Implementasi Automated Functional UI Testing Untuk Mencapai Konsep Ci/Cd Terintegrasi Jenkins Menggunakan Framework Serenity Dengan Page Object Model Design Pattern Pada Ppdb Jatim'** (Prasetyo, 2021). Pada jurnal ini, dilakukan pengujian otomatis terhadap situs web PPDB Jatim menggunakan alat bantu Serenity. Pengujian yang dilakukan meliputi pembuatan pengujian otomatis, pengujian fungsionalitas situs web secara otomatis, hingga mengimplementasikan konsep CI/CD dengan mengintegrasikan pengujian otomatis pada Jenkins. Setelah dilakukan pembuatan pengujian otomatis menggunakan Serenity, hasil dari pengujian otomatis yang telah didapatkan lalu dibandingkan dengan ekspektasi hasil uji menggunakan Hamcrest, dengan begitu dapat diketahui apakah hasil yang didapat telah sesuai dengan ekspektasi atau tidak. Pada integrasi Jenkins digunakan Apache dari XAMPP. Jenkins dijalankan pada localhost dengan port 8080, serta diperlukan Java dan Maven untuk menjalankan program pengujian.. Pada implementasi ini, terdapat tiga *Job* yang dibangun. *Job* pertama berguna untuk mendeteksi adanya perubahan yang dilakukan oleh pengembang pada situs web. *Job* kedua berfungsi untuk menjalankan pengujian otomatis yang telah diintegrasikan. Dan yang terakhir, *Job* ketiga digunakan untuk mengotomatisasi proses *deploy* situs web. Dengan dilakukannya implementasi pengujian otomatis menggunakan Serenity serta Jenkins yang mendukung konsep CI/CD, didapatkan beberapa poin penting, yaitu: waktu pengujian yang lebih singkat, laporan yang terbentuk secara otomatis dan terstruktur dari pengujian otomatis, serta pengembangan sistem yang lebih aman dan cepat.

Terdapat beberapa jurnal ilmiah lainnya diantaranya berjudul **'Perancangan Automated Testing Pada Studi Kasus Website Indicar'** (Prakoso dan Sujarwo, 2022). Dimana dilakukan perancangan pengujian otomatis pada situs web menggunakan beberapa alat bantu pengujian otomatis, dan salah satunya merupakan Katalon Studio. Serta **'Automating Web Application Testing Using Katalon Studio'** (Ereiz, 2019), dengan melakukan pengujian otomatis fungsional situs web menggunakan mode *record* dan *replay* yang disediakan oleh Katalon Studio.

2.2 Dasar Teori

2.2.1 User Story

User story merupakan salah satu metode populer untuk mewakili kebutuhan secara sederhana, contohnya: "*Sebagai seorang [role], saya ingin, [goal], agar [manfaat]*" (Lucassen et al, 2016). Dengan begitu, *User story* menjelaskan tentang kegunaan yang memiliki peran penting untuk pengguna atau pembeli sebuah sistem perangkat lunak. *User story* ditentukan melalui tiga aspek, yaitu: (Cohn, 2004)

1. Deskripsi tertulis mengenai *story* yang digunakan untuk perencanaan dan mengingat.
2. Percakapan mengenai *story* yang menjelaskan secara detail.
3. Menyatakan pengujian dan detail dokumen yang dapat digunakan untuk menentukan kapan *story* selesai.

2.2.2 Kasus Pengujian

Sedangkan Kasus pengujian merupakan pengujian yang menjalankan sesuai dengan tujuan yang telah dirancang dengan baik, seperti perilaku dari suatu fitur dalam kondisi yang spesifik (Nguyen, 2001). Kasus pengujian juga memiliki serangkaian masukkan yang diberikan ke program, hasil yang diharapkan, serta prasyaratnya (Jorgensen, 2013). Contohnya: ‘*When user login with unregistered email, system will show alert email/password is incorrect*’. Ketika pengujian dilakukan, penguji akan membandingkan hasil yang diperoleh dan hasil yang diharapkan. Jika kedua hal tersebut berbeda maka pengujian tersebut dinyatakan gagal dan harus dicatat dengan baik untuk menemukan penyebabnya (Sharma, 2014).

2.2.3 Functional Testing

Pada pengembangan perangkat lunak, pengujian fungsional merupakan salah satu bagian yang sangat penting. Pengujian fungsional mengharuskan penguji memilih data uji berdasarkan sifat penting dari elemen dalam domain variabel *input* dan *output* program (Howden, 1980). Kasus uji untuk *functional testing* diperoleh dengan menganalisis spesifikasi dan pengetahuan umum terhadap aplikasi (Bühler, 2008). Dengan adanya pengujian tersebut, dapat ditemukan kesalahan dan perilaku yang tidak diduga oleh perangkat lunak pada tahap awal sebelum dilakukannya perilisan (Ereiz, 2019).

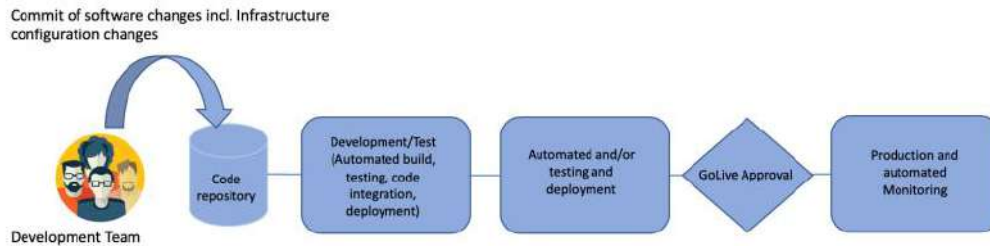
2.2.4 Automation Testing

Automation testing atau pengujian otomatis merupakan program yang dibuat oleh penguji menggunakan alat bantu pilihannya untuk melakukan pengujian pada perangkat lunak, lebih singkatnya, proses yang melibatkan otomasi terhadap proses manual (Sharma & Angmo, 2014). Saat digunakan, pengujian otomatis dapat menghemat waktu dan menjalankan pengujian jumlah besar secara efisien dalam waktu yang singkat (Mohammad, 2015). Tujuan digunakannya pengujian otomatis adalah untuk meminimalisir jumlah kasus pengujian yang akan dijalankan secara manual serta tidak menghilangkan pengujian manual secara bersamaan. Dalam pelaksanaan pengujian otomatis pada perangkat lunak, dilibatkan pengembangan *script* pengujian menggunakan Bahasa seperti Python, JavaScript, atau *Tool Command Language* (TCL) (Sharma, 2014).

2.2.5 Continuous Integration & Continuous Delivery (CI/CD)

Continuous integration merupakan praktik pengembangan perangkat lunak dimana pengembang dan penguji mengintegrasikan pekerjaan mereka dengan repositori kode sumber secara teratur dan mengotomatiskan pembuatan, pengujian, dan validasi perangkat lunak. Hal tersebut dapat memudahkan tim dalam menemukan dan memperbaiki kesalahan dengan cepat dan meningkatkan kualitas perangkat lunak.

Continuous Delivery adalah kemampuan untuk mendapatkan segala jenis perubahan dari fitur baru, perubahan konfigurasi, dan perbaikan kesalahan. Manfaat yang didapatkan dengan menggunakan continuous delivery yaitu adanya meningkatkan produktivitas dan efisiensi, perilsan yang terpercaya dan mempercepat waktu pemasaran (Arachchi & Perera, 2018). Pipeline Continuous delivery mengantarkan produk perangkat lunak berkualitas dengan cepat dari pengujian ke staging environment dan akhirnya ke produksi secara otomatis (Guşeilă & Moraru, 2019).



Gambar 2.2.1 Alur kerja CI/CD pipeline

2.2.6 Katalon Studio

Salah satu alat bantu *open-source* yang sering digunakan oleh tim penjaminan perangkat lunak untuk memudahkan pembuatan pengujian otomatis adalah Katalon Studio. Katalon Studio merupakan alat bantu pengujian otomatis yang dirilis oleh Katalon LLC. pada tahun 2015. Menggunakan Bahasa pemrograman Groovy, Java, serta JavaScript yang membuat Katalon Studio mampu untuk melakukan pengujian pada situs web, API, dan Mobile (Andorid, dan iOS) (Tjandra & Theopilus, 2021). Terdapat 3 cara pembuatan tes pada Katalon Studio, yaitu (Ereiz, 2019):

1. *Record* dan *Replay*, atau merekam dan mengulangi, dimana penguji akan merekam (*record*) langkah yang dijalankan pada perangkat lunak, setelah data tersebut tersimpan, akan dilakukan pengujian ulang (*replay*) jika diperlukan. Penguji dapat menambahkan data yang berisi berbagai informasi yang dapat digunakan oleh Katalon untuk proses pengujian
2. Manual mode, atau mode manual, pada mode ini, penguji dapat memasukkan langkah dan kata kunci yang digunakan pada kasus uji.
3. *Scripting* mode, atau mode *Scripting*, yang memungkinkan penguji menulis kode program menggunakan java dan atau Groovy untuk melakukan pengujian dan merekamnya.

2.2.7 Jenkins

Jenkins merupakan alat bantu *open source* untuk proses *Continuous Integration* (CI), yang ditulis menggunakan Bahasa pemrograman Java. Jenkins biasa digunakan untuk proyek dengan variasi Bahasa dan teknologi luas, seperti .NET, Ruby, Groovy, PHP, Java, dan masih banyak lagi (Smart, 2011). Tugas dasar Jenkins berfokus pada pengembangan sistem perangkat lunak dan menjalankan beberapa langkah yang telah didefinisikan sebagai *Job* untuk memicu suatu aktivitas. Selain itu, Jenkins menyediakan fitur untuk

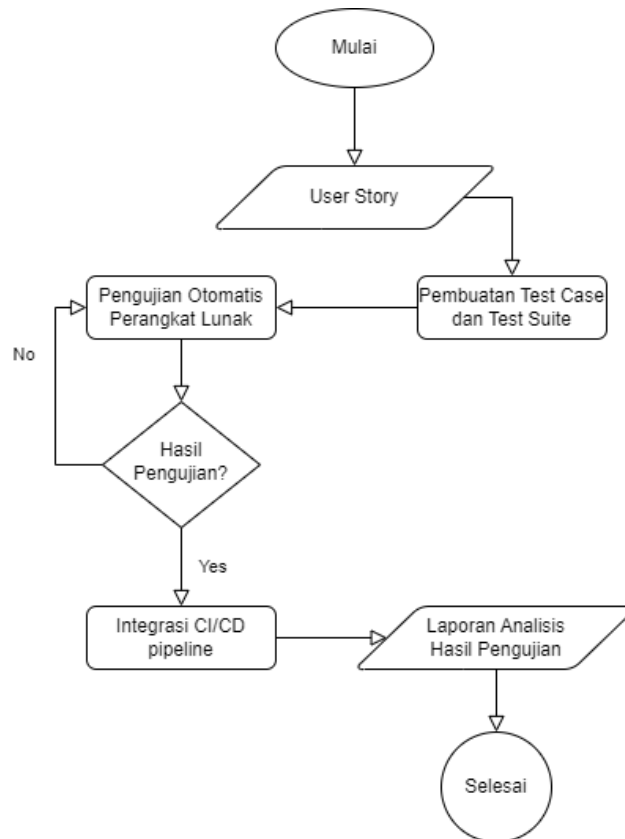
memelihara laporan pengujian, mengirimkan pemberitahuan mengenai status *build* kepada tim terkait, pemantauan proses, serta alat validasi bagi pengguna (Seth & Khare, 2015). Jenkins juga dapat dengan mudah diperluas dan diadaptasi untuk memproses rangkaian tugas komputasional dengan kompleksitas yang berubah-ubah (Moutsatsos, 2017).

Paraf Pembimbing I: 

Paraf Pembimbing II:

BAB III METODOLOGI

3.1 Metode yang dirancang



Gambar 3.1.1 Alur pengujian otomatis MyITS Thesis

Tugas Akhir ini akan menciptakan dan melakukan pengujian otomatis terhadap situs web myITS Thesis yang kemudian dilanjutkan dengan evaluasi dan analisis hasil pengujian otomatis, bagaimana praktik terbaik dalam mengimplementasikan pengujian otomatis pada sistem serta cara tim pengembang dan penjaminan mutu mengeksekusi dan memelihara kode sumber pengujian otomatis dalam jangka panjang. Alur pengerjaan Tugas Akhir ini secara singkat sebagai berikut:

1. Analisis Studi Kasus

Analisis studi kasus bertujuan untuk mengetahui proses bisnis yang terjadi pada sistem studi kasus. Pada tugas akhir ini, studi kasus yang digunakan adalah situs web myITS Thesis. Analisis studi kasus dilakukan dengan meninjau dan mempertimbangkan *User Story* yang telah didefinisikan oleh manajer proyek myITS Thesis. Dengan hal ini, dapat ditentukan kasus uji yang akan digunakan pada proses pengujian otomatis. Penjelasan lebih lengkap mengenai analisis studi kasus terdapat pada Bab 3.1.1.

2. Pembuatan Test Case

Tahapan ini berupa persiapan untuk melakukan pengujian otomatis. Dimana, kasus uji yang telah ditentukan akan dijadikan bahan untuk membuat *test suite*

untuk pengujian otomatis. Penjelasan lebih lengkap mengenai pembuatan kasus uji terdapat pada Bab 3.1.2.

3. Implementasi Pengujian Otomatis

Setelah kasus uji dibuat, dilakukan pembuatan pengujian otomatis menggunakan alat bantu Katalon Studio dengan mode *script*. Mode tersebut membantu dalam pembuatan *script* uji dari *test case* hingga *test suite*. Bahasa yang digunakan dalam pembuatan pengujian otomatis adalah Java yang mendukung pengujian terhadap situs web. Penjelasan implementasi pengujian otomatis lebih jelas terdapat pada Bab 3.1.3.1.

4. Integrasi pada CI/CD pipeline

Setelah pengujian otomatis selesai dibuat dan hasil uji telah sesuai dengan kasus uji yang diharapkan, dilakukan integrasi *script* pengujian otomatis pada CI/CD *pipeline*. Tahapan ini dilakukan dengan tujuan setiap tim pengembang melakukan perubahan pada kode sumber, hal itu akan memicu pengujian tereksekusi secara otomatis. Penjelasan integrasi pengujian otomatis pada CI/CD *pipeline* lebih jelas terdapat pada Bab 3.1.3.2.

3.1.1 Analisis Situs Web myITS Thesis Sebagai Studi Kasus

3.1.1.1 Analisis Permasalahan

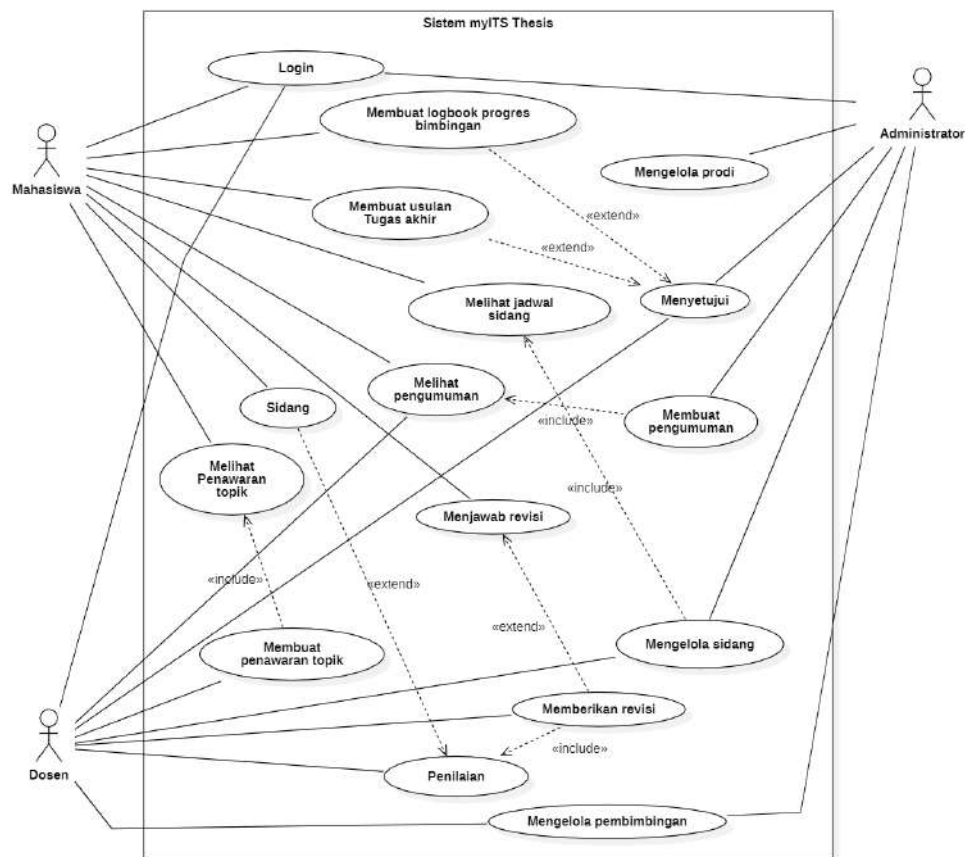
Permasalahan utama yang diangkat dalam pembuatan Tugas Akhir ini adalah bagaimana membangun dan mengimplementasikan sebuah sistem pengujian otomatis agar memudahkan proses pengembangan dan pemeliharaan sistem myITS Thesis dalam jangka waktu yang panjang. Dengan adanya penambahan proses pengujian otomatis dalam sistem myITS Thesis, terdapat perubahan alur kerja yang sudah ada serta diperlukannya pemeliharaan terkait dengan pengujian otomatis. Perubahan yang terjadi pada alur kerja pengembangan, hanya terdapat pada CI/CD *pipeline*, Ketika terjadinya perubahan pada kode program myITS Thesis.

Terdapat dua situasi perubahan kode program yang mempengaruhi pengujian otomatis, yaitu perubahan kode program dalam memelihara sistem seperti memperbaiki *bugs*, dan perubahan kode program dalam pengembangan fitur baru pada sistem. Pada situasi pertama, dengan adanya pengujian otomatis ini dapat mendeteksi apakah program yang diubah sudah berjalan sesuai proses bisnis yang diinginkan dengan cepat. Pada kondisi kedua, karena sistem myITS Thesis akan digunakan dalam jangka waktu yang panjang, akan terjadi penambahan fitur-fitur baru untuk mempermudah pengguna. Tidak hanya sistem myITS Thesis saja yang perlu diperbarui, tetapi juga sistem pengujian otomatis sesuai dengan fitur baru yang dikembangkan.

Oleh karena itu, dalam Tugas Akhir, ini penulis akan menyelesaikan masalah pada kedua situasi tersebut yang telah dipaparkan dengan memberikan praktik terbaik dalam mengimplementasikan pengujian otomatis sistem myITS Thesis, untuk membuat proses pengujian lebih cepat serta memberikan rekomendasi alur kerja baru dan cara pembaruan pengujian otomatis setiap ada fitur baru yang dirilis. Sistem pengujian otomatis juga akan menerapkan CI/CD untuk mengintegrasikan pengujian pada *pipeline* yang ada. Dengan begitu, pengujian dapat dilakukan secara otomatis dalam proses pengembangan dan dapat dieksekusi secara berkala tanpa menguras sumber daya yang ada.

3.1.1.2 Deskripsi Umum Sistem

Sistem yang digunakan sebagai bahan uji tugas akhir ini adalah situs web myITS Thesis. MyITS Thesis merupakan aplikasi untuk mendigitalisasikan proses dan manajemen tugas akhir, tesis, dan disertasi serta untuk mempermudah pemantauan dan pemerataan pembimbingan dosen terhadap mahasiswa. Secara garis besar, situs web yang dikembangkan oleh DPTSI ITS memudahkan mahasiswa, dosen serta administrator dalam pembimbingan, pengajuan sidang, dan melihat pemberitahuan mengenai tugas akhir, tesis, dan disertasi. Setiap pengguna memiliki penggunaan fitur yang berbeda. Mahasiswa, dosen, serta administrator dapat melakukan *login* menggunakan *username* dan *password* yang digunakan pada myits dashboard. Untuk lebih detailnya akan dijelaskan pada pemaparan selanjutnya.



Gambar 3.1.2 Use Case diagram sistem myITS Thesis

3.1.1.3 Aktor/User

Melalui *User Story* yang telah tersedia, ditentukan *User* pada myITS Thesis terbagi menjadi enam *role user*, diantaranya merupakan: mahasiswa, dosen, dan administrator. Untuk dosen dan administrator, dikategorikan lagi menjadi dosen pembimbing, dan dosen penguji, serta administrator prodi, dan administrator sidang. Setiap *role user* memiliki hak akses dan tugas yang berbeda. Saat *user* masuk akun atau *login*, akan ditampilkan beranda dengan konten dan fitur yang berbeda. Role administrator prodi berhak atas semua fitur yang ada pada situs web myITS Thesis. Hak akses yang diberikan ke masing-masing role melalui *User Story* akan dijelaskan lebih rinci dalam Tabel 3.1.1.

Tabel 3.1.1 Aktor dan hak akses pada myITS Thesis

No	Aktor	Hak Akses
1	Mahasiswa	Membuat usulan tugas akhir/tesis/disertasi
		Melihat penawaran topik tugas akhir/tesis/disertasi
		Mendaftar sidang pada periode yang dibuka
		Membuat rekap progres bimbingan
		Mengajukan permintaan perubahan data tugas akhir/tesis/disertasi
		Melihat riwayat perubahan tugas akhir/tesis/disertasi mahasiswa
		Melihat pengumuman yang dibuat oleh admin prodi
		Melihat dan mengunduh daftar panduan tugas akhir/tesis/disertasi
		Dapat menyetujui atau menolak bergabung menjadi anggota tim tugas akhir/tesis/disertasi
		Melihat daftar permintaan untuk bergabung pada tim Tugas Akhir/Tesis/Disertasi
		Dapat menjawab revisi tugas akhir/tesis/disertasi setelah sidang selesai
2	Dosen Pembimbing	Melihat pengumuman yang dibuat oleh admin prodi
		Melihat dan mengunduh daftar panduan tugas akhir/tesis/disertasi
		Dosen pembimbing dapat melihat halaman Bimbingan Dosen pembimbing
		Melihat daftar dan detail Permintaan Pembimbingan Tugas Akhir/Tesis/Disertasi
		Menyetujui dan menolak permintaan pembimbingan tugas akhir/tesis/disertasi
		Menyetujui pengajuan sidang tugas akhir/tesis/disertasi mahasiswa bimbingan
		Memberi nilai tugas akhir/tesis/disertasi mahasiswa
		Menyetujui progres bimbingan tugas akhir/tesis/disertasi mahasiswa
		Menyetujui perubahan data tugas akhir/tesis/disertasi mahasiswa bimbingan
		Memberi revisi terhadap tugas akhir/tesis/disertasi mahasiswa
		Melihat, menyetujui, dan menolak jawaban revisi tugas akhir/tesis/disertasi mahasiswa
3	Dosen Penguji	Memberi nilai tugas akhir/tesis/disertasi mahasiswa
		Menentukan kelulusan mahasiswa ketika sidang
		Memberi revisi terhadap tugas akhir/tesis/disertasi mahasiswa

No	Aktor	Hak Akses
		Melihat, menyetujui, dan menolak jawaban revisi tugas akhir/tesis/disertasi mahasiswa
4	Dosen	Melihat, membuat, dan menyunting penawaran topik tugas akhir/tesis/disertasi dosen
		Melihat seluruh topik tugas akhir/tesis/disertasi
5	Administrator Prodi	Mengatur boleh/tidaknya mahasiswa berkelompok untuk pengajuan tugas akhir/tesis /disertasi
		Melihat halaman Pembimbingan Dosen
		Mengatur kuota bimbingan dosen
		Melihat daftar Tugas Akhir/Tesis/Disertasi
		Membuat dan mengatur jadwal periode sidang
		Membatalkan sidang
		Membuat kriteria penilaian tugas akhir/tesis/disertasi dalam prodi
		Membuat, melihat, dan merubah pengumuman terkait dengan tugas akhir/tesis/disertasi
		Membuat, melihat, dan merubah panduan tugas akhir/tesis/disertasi terkait
6	Administrator Sidang	Mengganti dosen pembimbing

3.1.1.4 Fitur

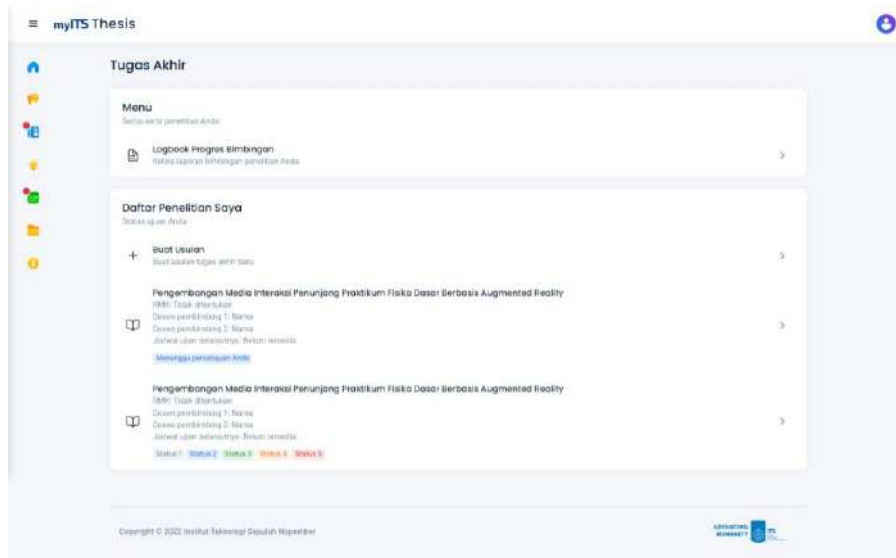
Fitur yang terdapat pada myITS Thesis meliputi kegiatan progres pembimbingan, melihat penawaran topik tugas akhir yang disediakan dosen dan pembuatan penawaran topik tugas akhir oleh dosen, pengajuan sidang tugas akhir yang dapat diakses oleh mahasiswa dan dosen pembimbing, persetujuan sidang dan tugas akhir, penilaian tugas akhir oleh dosen pembimbing atau dosen penguji, hingga melihat pengumuman terbaru oleh admin prodi. Secara garis besar, fitur yang terdapat pada sistem myITS Thesis akan dijelaskan lebih rinci sebagai berikut

3.1.1.4.1 Manajemen Tugas Akhir/Tesis/Disertasi

Fitur manajemen tugas akhir/tesis/disertasi digunakan untuk memantau dan mengatur proses tugas akhir/tesis/disertasi yang diajukan oleh mahasiswa. Fitur ini dapat diakses oleh mahasiswa, dosen pembimbing, dosen penguji, serta administrator prodi. Tetapi setiap role akan mendapatkan tampilan halaman yang berbeda sesuai dengan hak akses yang diberikan.

Pada tampilan untuk mahasiswa terdapat Menu Catatan Penelitian dan

Pembimbingan (Gambar 3.1.3). Pada menu tersebut mahasiswa dapat melihat progress pembimbingan dan catatan atau komentar dari dosen pembimbing. Selanjutnya, terdapat Menu Tugas Akhir Saya, dimana mahasiswa dapat melihat status tugas akhir yang telah diusulkan serta membuat usulan tugas akhir/tesis/disertasi dengan melakukan klik pada Buat Usulan Baru, lalu sistem akan mengalihkan pengguna ke halaman Usulan Tugas Akhir (Gambar 3.1.4). Pada halaman tersebut, mahasiswa dapat mengusulkan tugas akhir/tesis/disertasi dengan memasukkan informasi pada form, seperti Judul, Judul dalam Bahasa Inggris, Rencana Penelitian, Kata kunci, Kata kunci dalam Bahasa Inggris, serta kandidat dosen pembimbing. Setelah semua form terisi, usulan dapat disimpan dengan klik tombol Simpan Usulan. Sistem akan menampilkan *pop up* Simpan Usulan, untuk memastikan Usulan yang disimpan benar. Mahasiswa dapat klik Batal jika ingin merubah informasi atau Simpan jika ingin lanjut untuk menyimpan usulan. Selain itu, mahasiswa dapat melihat detail tugas akhir yang telah diajukan dan disimpan (Gambar 3.1.5). pada halaman detail tugas akhir ini, mahasiswa dapat melakukan konfirmasi untuk menjadi anggota kelompok penelitian yang ditawarkan.



Gambar 3.1.3 Tampilan menu manajemen tugas akhir mahasiswa

myITS Thesis

← Buat Usulan

Informasi Tugas Akhir

Judul (Inggris)

Judul dalam Bahasa Inggris

Struktur Modul
Pilih salah satu

Abstrak

Kata Kunci

Apakah bereskomprok?

☐ Ya ☐ Tidak

[Simpan sebagai Draft](#)

Copyright © 2022 Institut Teknologi Sepuluh Nopember

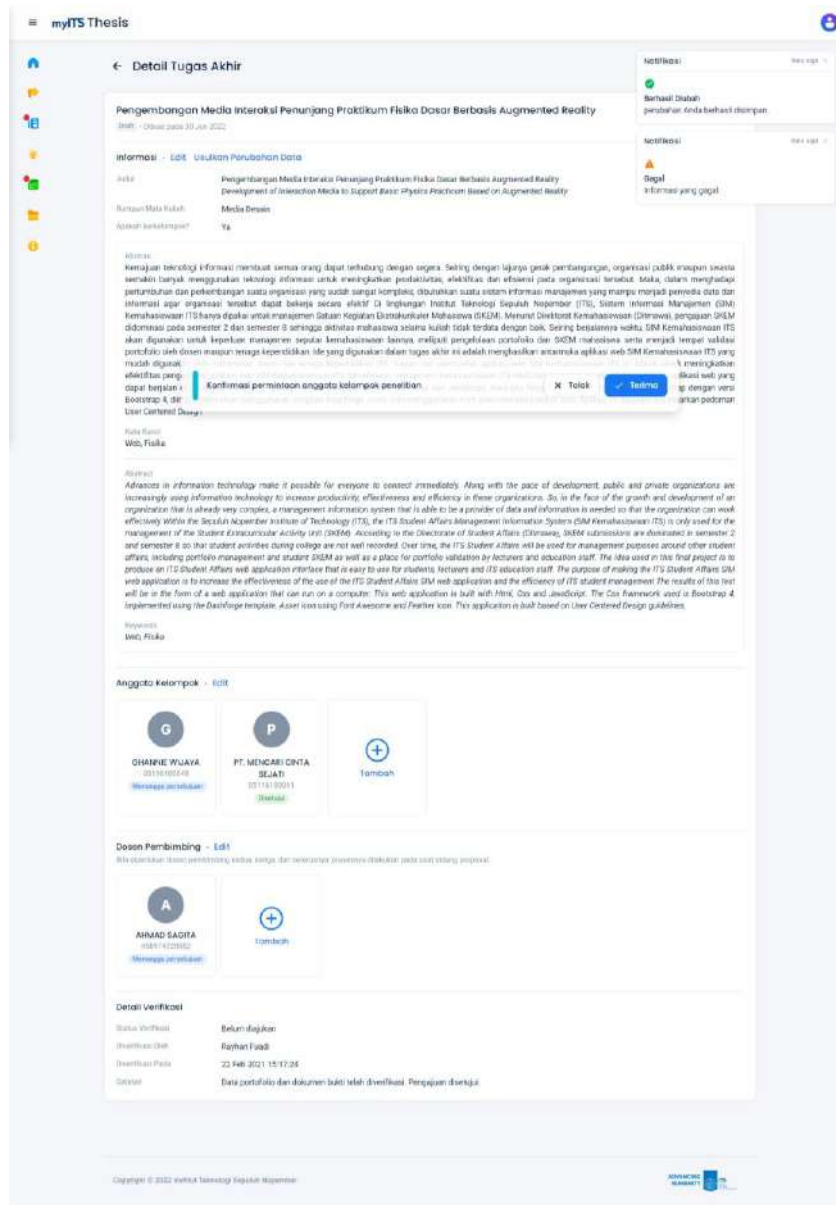
KOMUNITAS
KOMUNITAS ITS

Gambar 3.1.4 Tampilan menu usulan tugas akhir

Paraf Pembimbing I:

Sugeng

Paraf Pembimbing II:



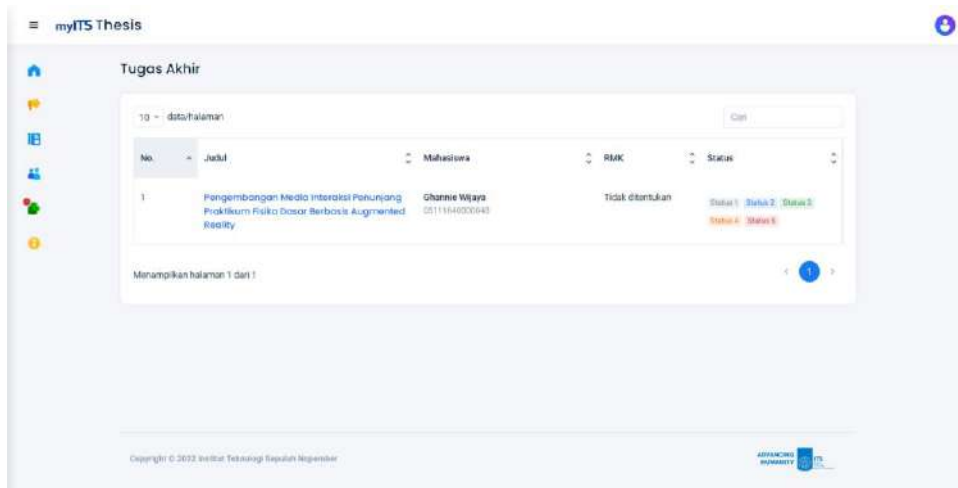
Gambar 3.1.5 Tampilan detail tugas akhir

Selanjutnya pada tampilan untuk Administrator mengakses fitur tugas akhir. Administrator dapat melihat daftar judul tugas akhir/tesis/disertasi mahasiswa serta RMK, dan status tugas akhir tersebut (Gambar 3.1.6). administrator juga dapat melihat detail tugas akhir mahasiswa (Gambar 3.1.7) dengan klik judul tugas akhir yang ingin dilihat. Pada halaman detail tugas akhir, administrator dapat melakukan persetujuan tugas akhir/tesis/disertasi mahasiswa.

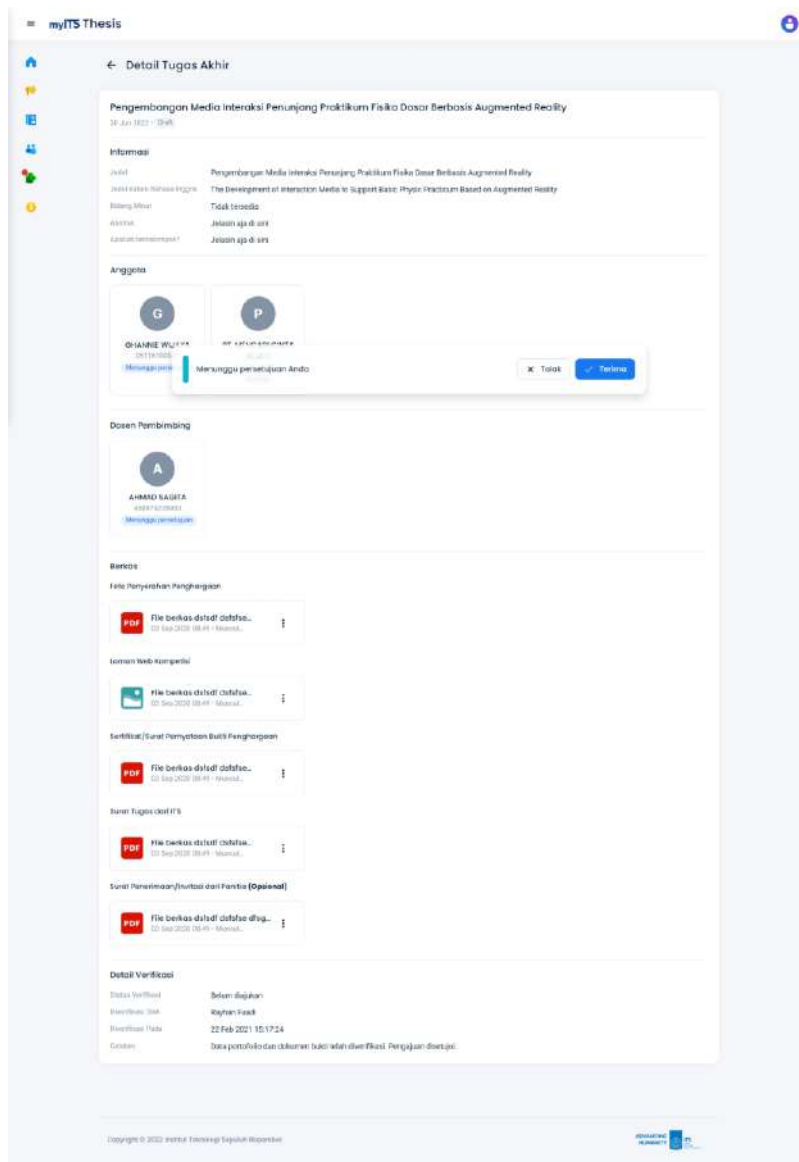
Paraf Pembimbing I:

Sugeng

Paraf Pembimbing II:



Gambar 3.1.6 Tampilan fitur tugas akhir administrator



Gambar 3.1.7 Tampilan Halaman detail tugas akhir administrator

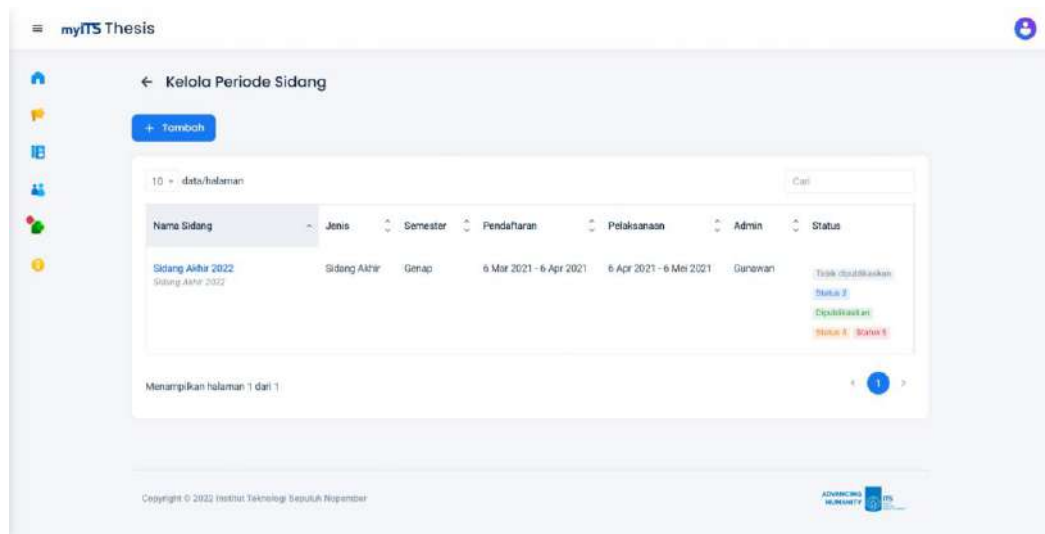
Paraf Pembimbing I:

Surya

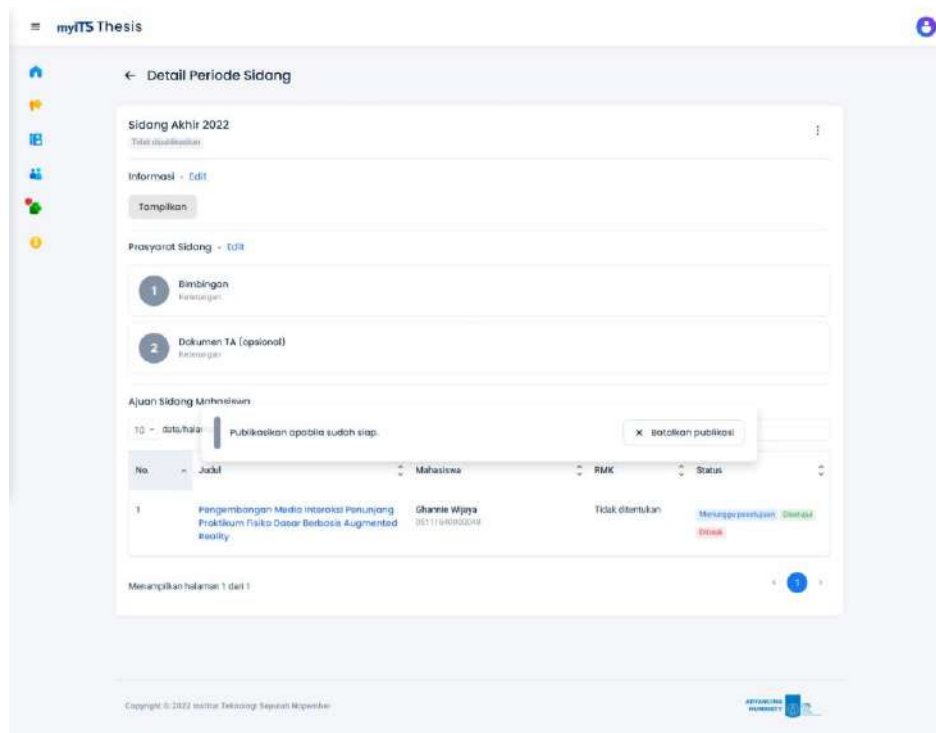
Paraf Pembimbing II:

3.1.1.4.2 Manajemen Sidang

Fitur manajemen sidang digunakan untuk mengatur jadwal periode sidang serta pendaftaran sidang mahasiswa. Fitur ini dapat diakses oleh seluruh role (mahasiswa, dosen, dan administrator) dengan haknya masing-masing. Yang pertama, administrator hanya dapat melakukan pengelolaan periode sidang (Gambar 3.1.8). Pada halaman ini, administrator dapat melihat daftar sidang yang berisi nama sidang, jenis, semester, pendaftaran, pelaksanaan, admin, serta status sidang. Administrator juga dapat melihat, mengubah, serta menghapus periode sidang pada halaman Detail periode sidang (Gambar 3.1.9).



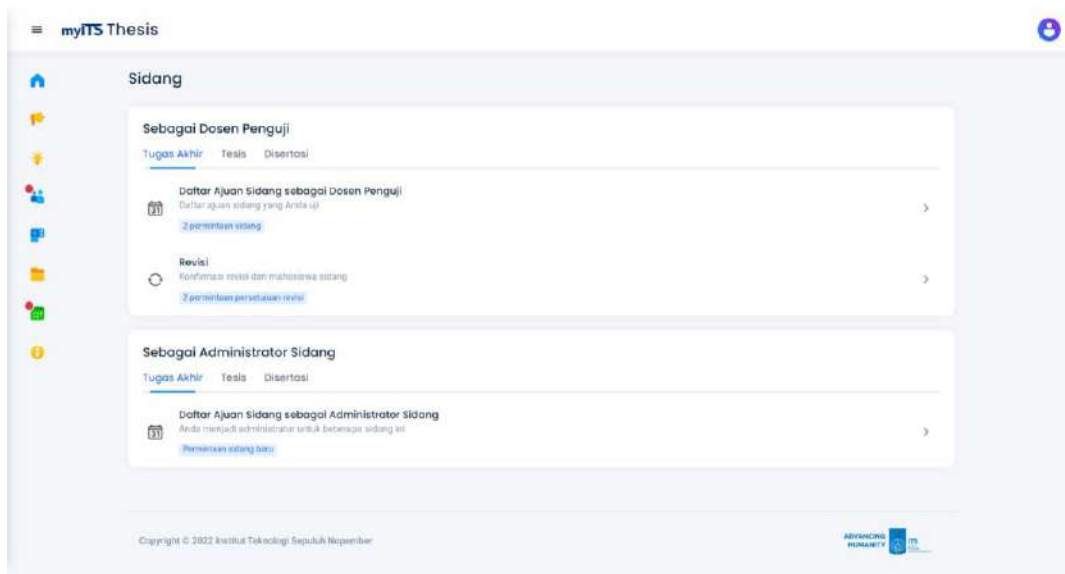
Gambar 3.1.8 Tampilan halaman Kelola periode sidang administrator



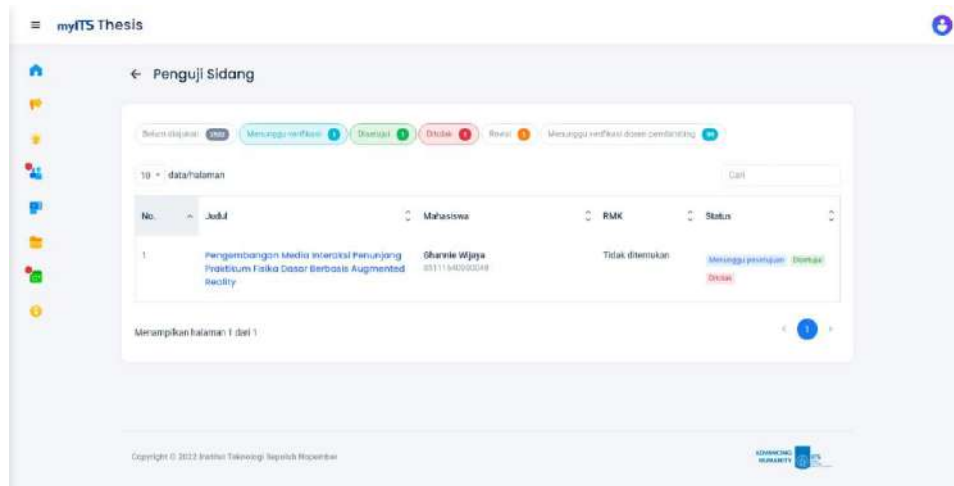
Gambar 3.1.9 Tampilan halaman detail periode sidang administrator

Selanjutnya fitur manajemen sidang pada dosen. Dosen dapat melihat halaman awal fitur Sidang (Gambar 3.1.10). Pada halaman ini, dosen dibagi menjadi dua kategori, yaitu dosen penguji dan administrator sidang. Jika dosen tersebut adalah dosen penguji, akan ditampilkan Menu Sebagai Dosen Penguji. Pada menu ini, dosen dapat mengakses daftar tugas akhir, tesis, serta disertasi yang diuji serta daftar revisi sidang. Dosen penguji dapat melihat lebih lengkap daftar ajuan sidang mahasiswa (Gambar 3.1.11) terdapat Judul, Mahasiswa yang bersangkutan, RMK serta status sidang. Dosen penguji juga dapat mengakses detail tugas akhir/tesis/disertasi mahasiswa dengan klik judul yang terdapat pada daftar, dan sistem akan menampilkan halaman detail ajuan sidang (Gambar 3.1.12). Pada halaman detail ajuan sidang, dosen penguji dapat melakukan persetujuan ajuan sidang mahasiswa, absen kehadiran, penilaian sidang, merubah status kelulusan, dan memberi revisi. Dosen penguji juga dapat melihat detail sidang, prasyarat sidang, informasi tugas akhir, anggota kelompok, dosen pembimbing, detail verifikasi, dan status sidang.

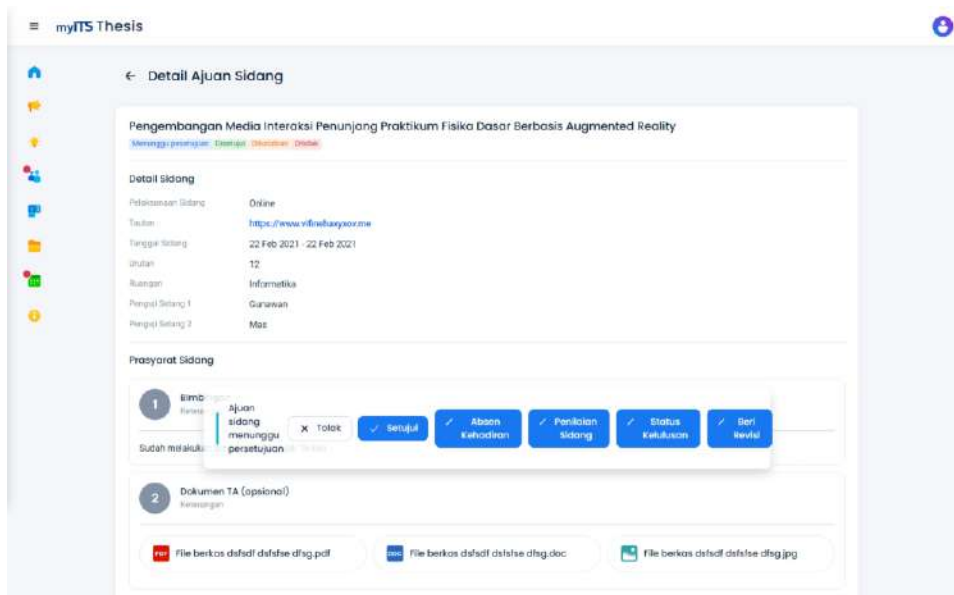
Selain daftar ajuan sidang, dosen penguji dapat melihat daftar revisi sidang (Gambar 3.1.13), dosen penguji dapat melihat judul, mahasiswa, serta status. Jika dosen melakukan klik pada judul tugas akhir yang diinginkan, sistem akan menampilkan halaman detail revisi penelitian (Gambar 3.1.14). Pada halaman ini, dosen penguji dapat melihat detail singkat tugas akhir/tesis/disertasi yang direvisi serta daftar revisi yang telah dilakukan. Dosen penguji juga dapat menentukan jika revisi tersebut diterima atau ditolak.



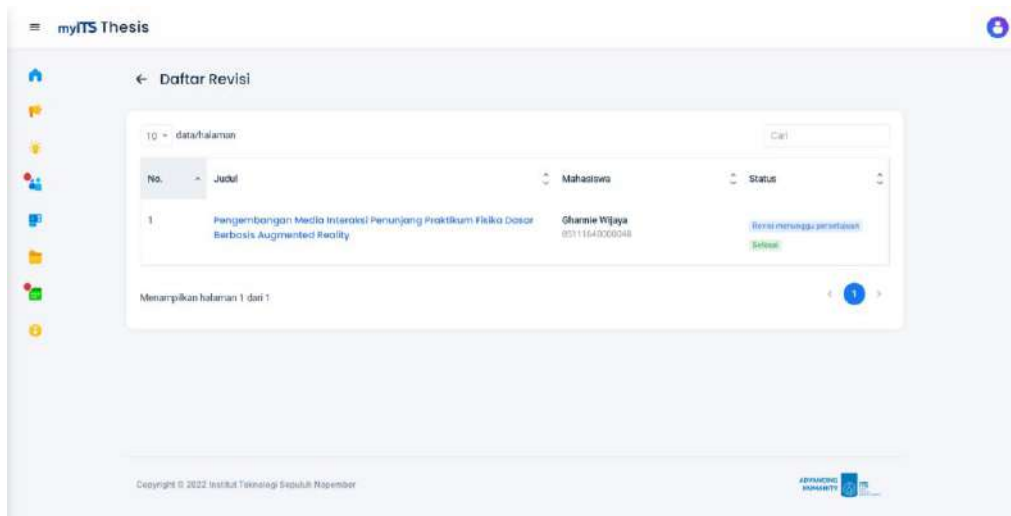
Gambar 3.1.10 Tampilan halaman sidang dosen



Gambar 3.1.11 Tampilan halaman daftar ajuan sidang



Gambar 3.1.12 Tampilan halaman detail ajuan sidang dosen penguji

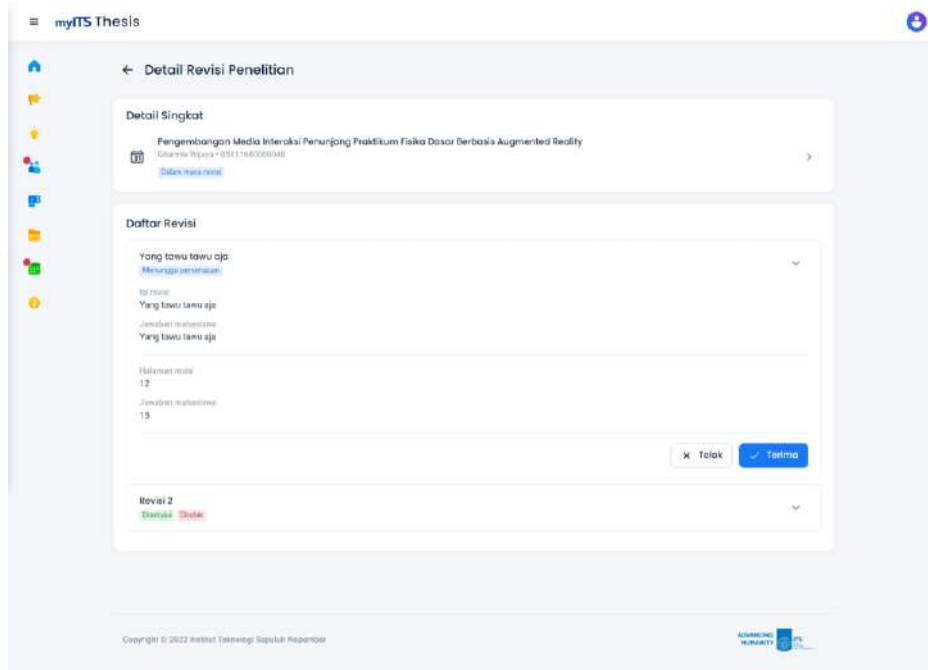


Gambar 3.1.13 Tampilan halaman daftar revisi

Paraf Pembimbing I:

Surya

Paraf Pembimbing II:

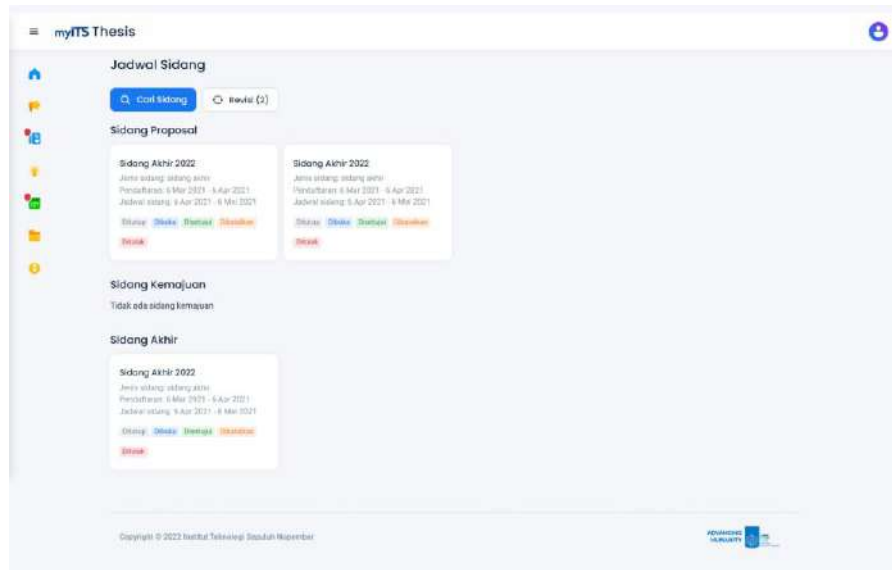


Gambar 3.1.14 Tampilan halaman detail revisi penelitian

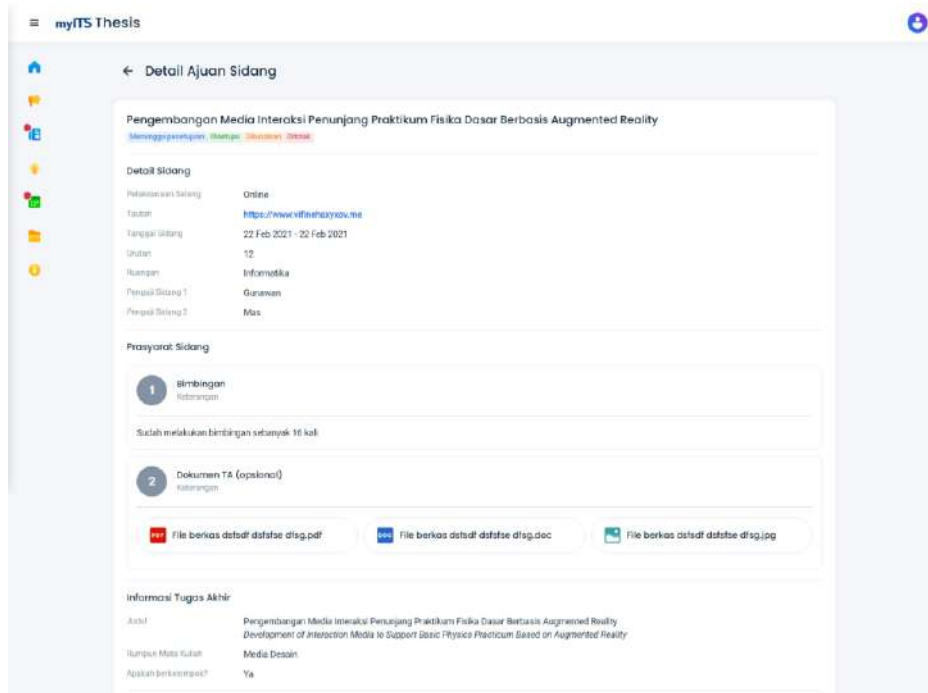
Yang kedua, jika dosen merupakan administrator sidang, maka pada halaman Sidang (Gambar 3.1.10) akan ditampilkan menu Sebagai Administrator Sidang. Pada menu ini, dosen dapat melihat daftar ajuan sidang sama seperti dosen penguji. Perbedaannya adalah pada halaman detail, administrator sidang hanya dapat melakukan persetujuan sidang.

Selanjutnya, fitur manajemen sidang pada mahasiswa. Mahasiswa dapat melihat jadwal sidang yang telah diatur oleh administrator serta revisi tugas akhir/tesis/disertasi (Gambar 3.1.15). pada halaman ini terdapat jadwal sidang untuk sidang proposal, sidang kemajuan, serta sidang akhir. Ketika mahasiswa klik jadwal yang diinginkan, sistem akan menampilkan halaman detail ajuan sidang mahasiswa (Gambar 3.1.16) yang berisi detail sidang, prasyarat sidang, informasi tugas akhir, anggota kelompok, dosen pembimbing, serta detail verifikasi.

Selain itu, mahasiswa dapat melihat daftar revisi tugas akhir yang didapatkan setelah sidang (Gambar 3.1.17). pada halaman ini, terdapat informasi mengenai penelitian yang sedang berlanjut, isi revisi, halaman revisi, dosen penguji, serta status revisi. Mahasiswa dapat menjawab revisi dengan klik isi revisi yang diinginkan, sistem akan menampilkan *popup* detail revisi kepada mahasiswa (Gambar 3.1.18). Pada *popup* ini, mahasiswa dapat menjawab revisi yang diberikan dosen penguji dan menyimpannya.



Gambar 3.1.15 Tampilan halaman jadwal sidang mahasiswa

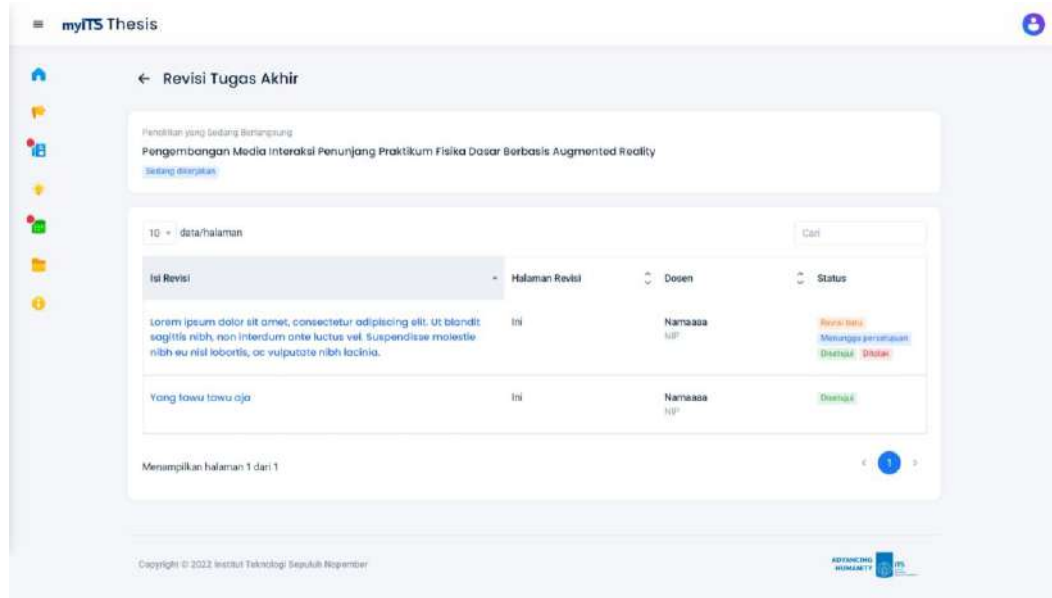


Gambar 3.1.16 Tampilan halaman detail ajuan sidang mahasiswa

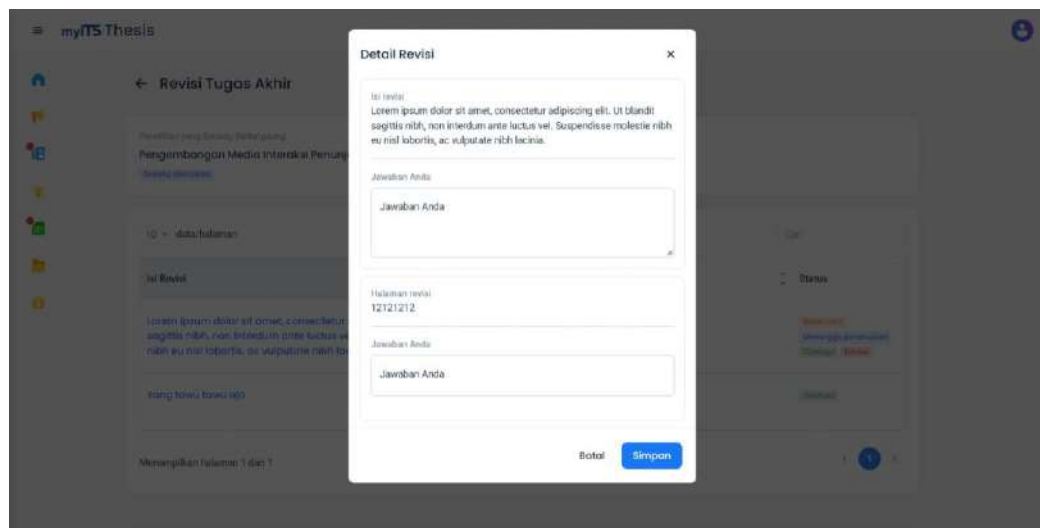
Paraf Pembimbing I:

Sugeng

Paraf Pembimbing II:



Gambar 3.1.17 Tampilan halaman revisi tugas akhir mahasiswa



Gambar 3.1.18 Tampilan popup detail revisi mahasiswa

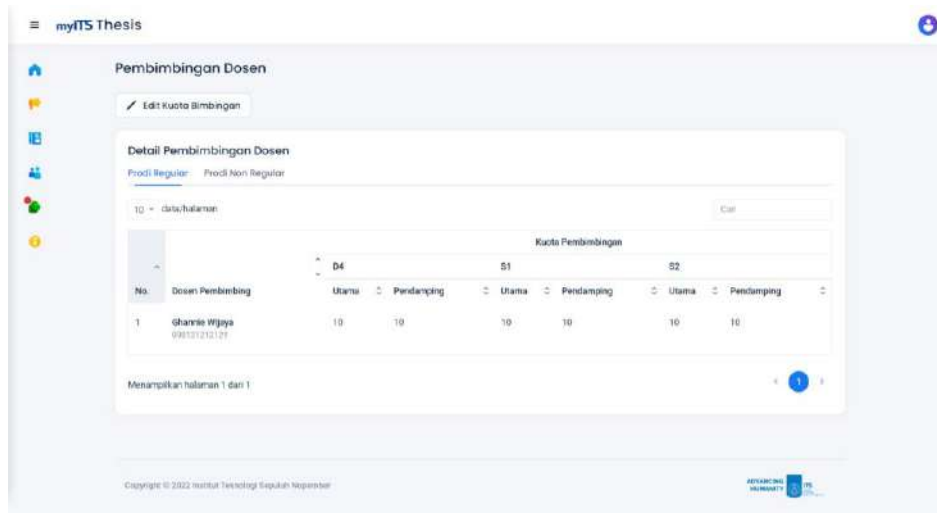
3.1.1.4.3 Manajemen Pembimbingan

Fitur manajemen pembimbingan digunakan untuk memantau dan mengatur proses pembimbingan antara dosen dengan mahasiswa. Fitur ini hanya dapat diakses oleh role dosen dan administrator. Administrator hanya dapat melihat detail pembimbingan dosen pada prodi regular dan non-regular (Gambar 3.1.19) serta melakukan perubahan kuota bimbingan dosen (Gambar 3.1.20) agar jumlah mahasiswa dan dosen penguji terbagi secara rata.

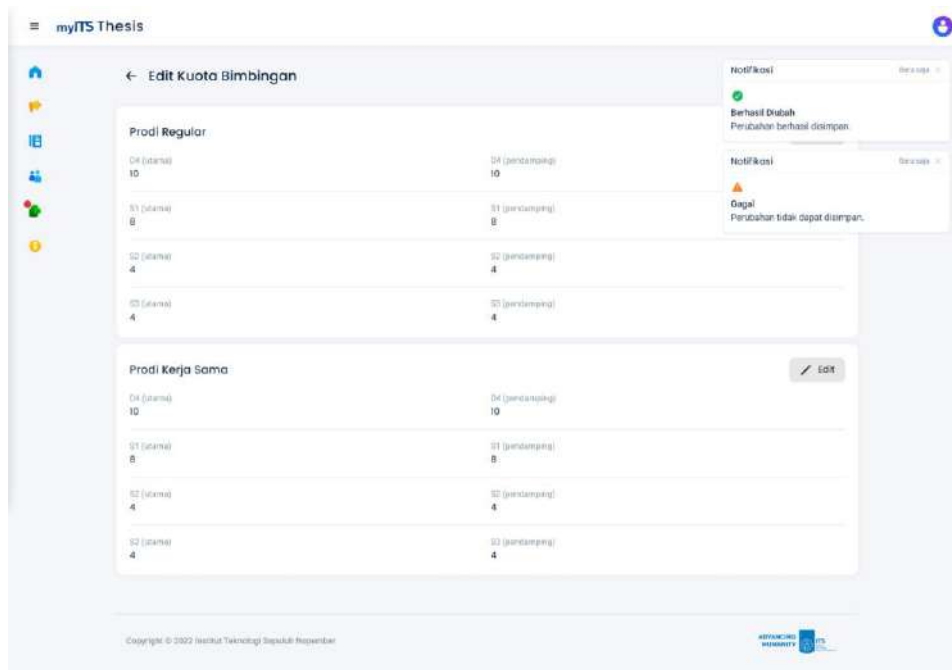
Paraf Pembimbing I:

Sugeng

Paraf Pembimbing II:



Gambar 3.1.19 Tampilan halaman pembimbingan dosen administrator



Gambar 3.1.20 Tampilan halaman perubahan kuota bimbingan

Selanjutnya pada role dosen, fitur manajemen pembimbingan hanya dapat diakses oleh dosen pembimbing. Dosen pembimbing dapat melihat halaman Bimbingan Saya (Gambar 3.1.21) yang berisi Kuota pembimbingan pada prodi regular dan prodi kerja sama, penelitian tugas akhir, tesis, serta disertasi, serta sidang. Yang pertama, pada Kelola penelitian mahasiswa bimbingan, pada bagian ini, dosen dapat melihat detail tugas akhir/tesis/disertasi, logbook proses pembimbingan, serta usulan perubahan data. Dosen dapat melihat daftar tugas akhir/tesis/disertasi bimbingan (Gambar 3.1.22) serta detail tugas akhir/tesis/disertasi (Gambar 3.1.23). Pada halaman detail ini, dosen dapat melakukan persetujuan tugas akhir/tesis/disertasi yang diajukan.

Kemudian terdapat halaman detail logbook progress bimbingan dosen

(Gambar 3.1.24). Pada halaman ini, dosen dapat melihat detail singkat mengenai penelitian yang dikerjakan mahasiswa dan daftar logbook bimbingan yang ditulis oleh mahasiswa. Daftar logbook bimbingan tersebut dapat disetujui atau ditolak oleh dosen pembimbing. Lalu yang terakhir, terdapat halaman usulan perubahan data (Gambar 3.1.25). Halaman ini sama seperti halaman detail tugas akhir, tetapi di halaman detail usulan perubahan data, dapat terlihat bagian-bagian mana saja yang mengalami perubahan. Dosen pembimbing dapat melakukan persetujuan usulan perubahan tersebut.

myITS Thesis

Bimbingan Saya

Sisa Kuota Membimbing Anda

Prodi Regular Prodi Kerja Sama

Pembimbing Utama						Pembimbing Pendamping					
D4	S1	S2	S2 Terapan	S3	S3 Terapan	D4	S1	S2	S2 Terapan	S3	S3 Terapan
0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10	0/10

Penelitian

Kelola penelitian mahasiswa bimbingan Anda

Tugas Akhir Tesis Disertasi

Daftar Tugas Akhir Mahasiswa Saya
Kelola bimbingan penelitian mahasiswa

2 permintaan bimbingan

Logbook Progres Bimbingan
Laporan bimbingan tugas akhir dari mahasiswa

2 laporan

Usulan Perubahan Data
Perubahan data dari tugas akhir mahasiswa

2 usulan perubahan

Sidang

Kelola ajuan sidang mahasiswa bimbingan Anda

Tugas Akhir Tesis Disertasi

Daftar Ajuan Sidang
Kelola ajuan sidang mahasiswa

2 permintaan sidang

Revisi
Kelola revisi pascasidang

2 jawaban revisi

Copyright © 2022 Institut Teknologi Sepuluh Nopember

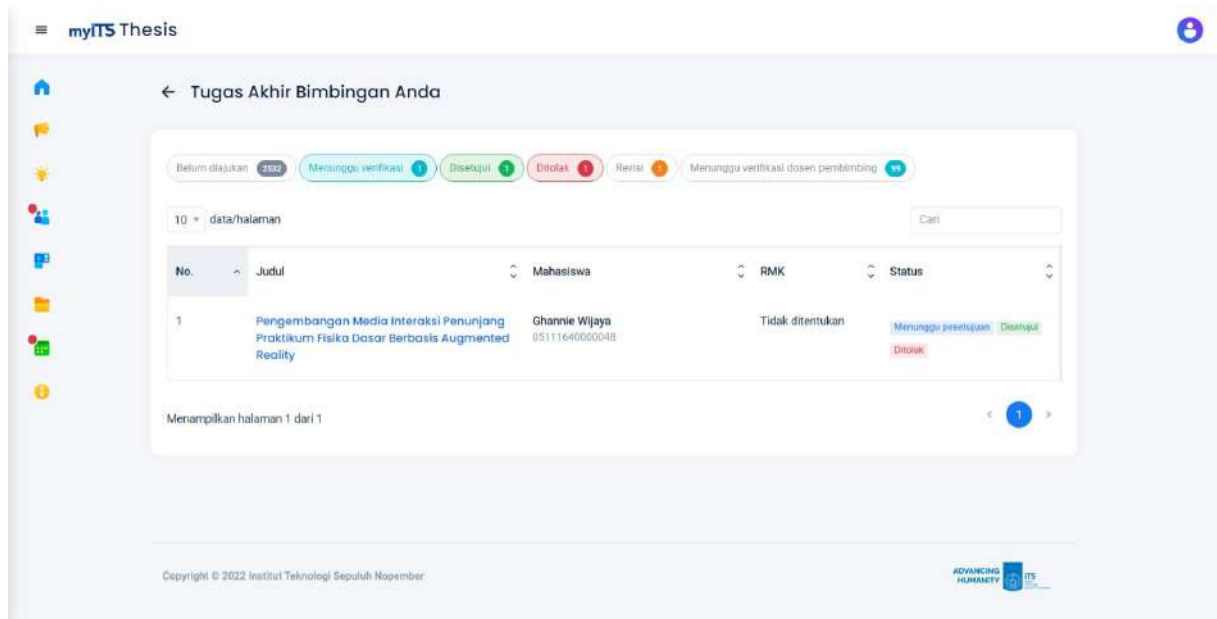
ADVANCING HUMANITY ITS

Gambar 3.1.21 Tampilan Halaman Bimbingan Saya dosen pembimbing

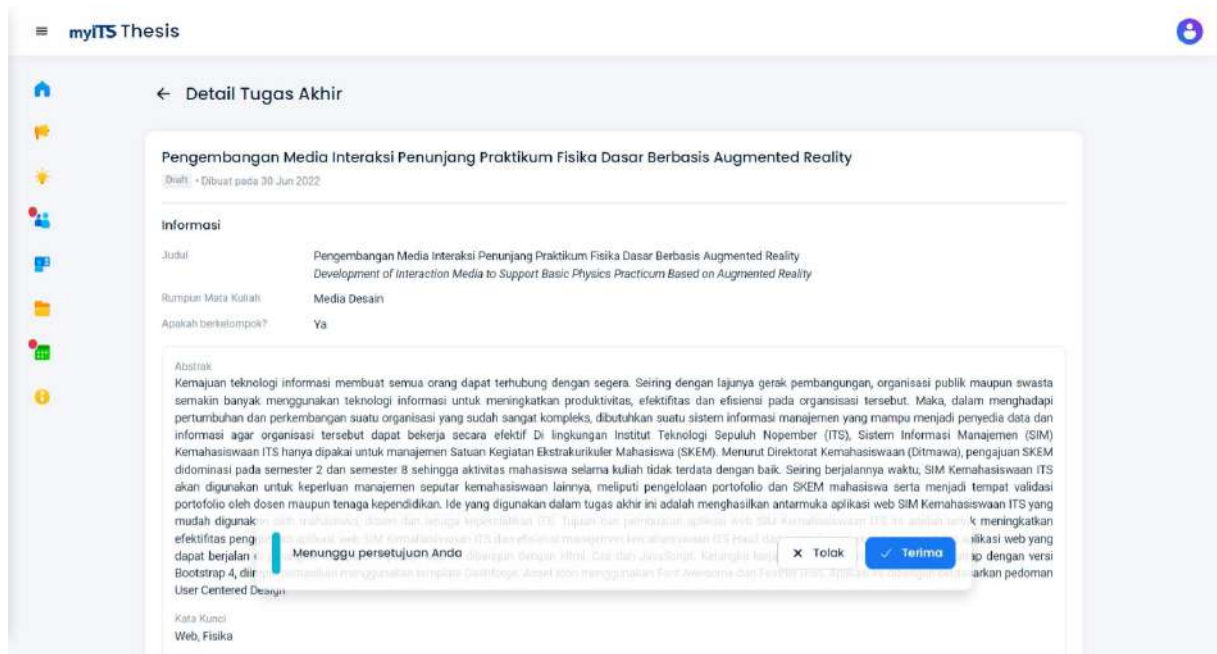
Paraf Pembimbing I:

Sugeng

Paraf Pembimbing II:



Gambar 3.1.22 Tampilan daftar tugas akhir bimbingan dosen

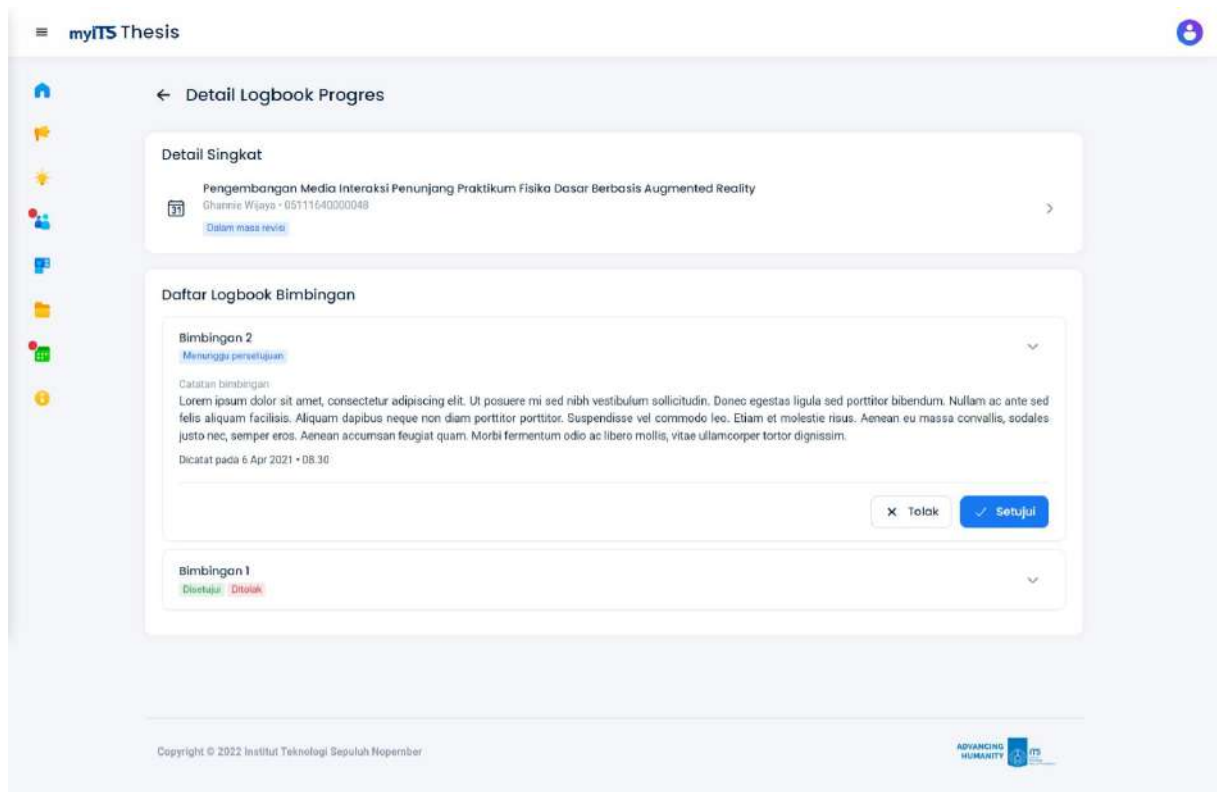


Gambar 3.1.23 Tampilan detail tugas akhir dosen pembimbing

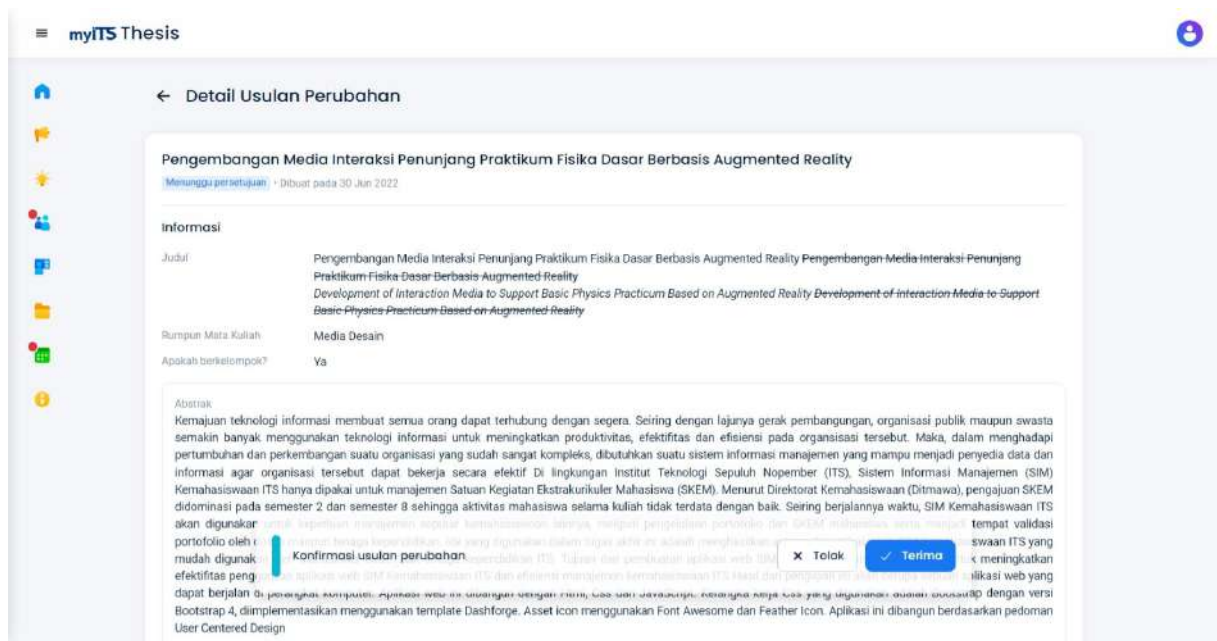
Paraf Pembimbing I:

Surya

Paraf Pembimbing II:



Gambar 3.1.24 Tampilan halaman detail logbook progress dosen pembimbing



Gambar 3.1.25 Tampilan halaman detail usulan perubahan dosen

3.1.1.4.4 Penilaian Tugas Akhir/Tesis/Disertasi

Fitur penilaian tugas akhir/tesis/disertasi hanya dapat dilakukan oleh dosen, terutama dosen pembimbing dan penguji. Dosen yang dikategorikan sebagai dosen

Paraf Pembimbing I: 

Paraf Pembimbing II:

penguji dapat mengaksesnya pada detail ajuan sidang (Gambar 3.1.12). Untuk melakukan penilaian, dosen penguji klik tombol Penilaian Sidang yang terdapat pada halaman tersebut dan sistem akan menampilkan *popup* Berikan Penilaian Sidang (Gambar 3.1.26). Dosen dapat memberikan nilai, komentar, serta catatan dosen. Setelah mengisi nilai, dosen akan diminta untuk menyimpan dengan klik tombol Simpan dan sistem akan menyimpan penilaian secara permanen dan tidak dapat diubah.

Gambar 3.1.26 Tampilan popup form penilaian sidang

3.1.1.4.5 Pemberian Revisi & Penilaian Revisi

Fitur pemberian revisi dan penilaian revisi dapat diakses oleh dosen penguji dan dosen pembimbing. Fitur pemberian revisi dapat diakses melalui halaman detail ajuan sidang (Gambar 3.1.12) dengan klik tombol Beri Revisi, sistem akan menampilkan *popup* Berikan Revisi (Gambar 3.1.27). Pada *popup* ini, dosen dapat mengisikan revisi dan halaman revisi jika ada. Dosen juga dapat menghapus revisi dengan klik icon sampah. Jika revisi yang diisikan sudah sesuai, dosen dapat menyimpannya dengan klik tombol Simpan

Gambar 3.1.27 Tampilan Popup Berikan Revisi dosen

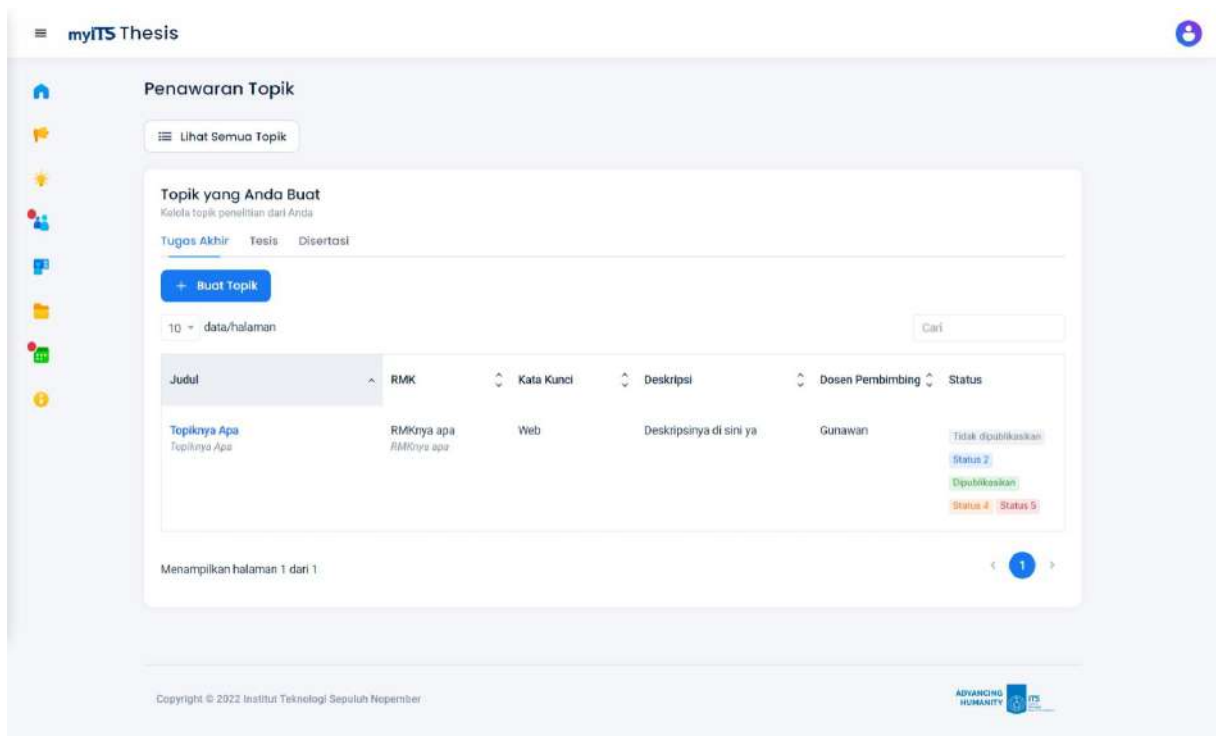
Paraf Pembimbing I:

Surya

Paraf Pembimbing II:

3.1.1.4.6 Manajemen Penawaran Topik

Fitur manajemen penawaran topik digunakan untuk mempermudah dosen untuk mencari mahasiswa yang tertarik untuk mengambil judul tugas akhir/tesis/disertasi yang dimiliki. Fitur ini dapat diakses oleh dosen dan mahasiswa. Pada role dosen, dosen dapat membuat penawaran topik tugas akhir, tesis, dan disertasi. Fitur ini dapat diakses pada halaman Penawaran Topik (Gambar 3.1.28), pada halaman ini dosen dapat melihat daftar penawaran topik yang telah dibuat. Untuk membuat penawaran topik, dosen dapat klik tombol Buat Topik dan sistem akan menampilkan halaman Buat Topik (Gambar 3.1.29). Halaman ini berisi formulir yang perlu diisi untuk menawarkan topik, berisi judul, bidang minat, deskripsi, serta kata kunci. Setelah disimpan, topik akan muncul pada daftar di halaman penawaran topik. Dosen juga dapat melihat, mengubah, serta menghapus detail penawaran topik yang telah dibuat pada halaman detail topik (Gambar 3.1.30). Selain itu, dosen juga dapat melihat daftar semua topik dari dosen lainnya dengan klik tombol Lihat semua topik pada Gambar 3.1.28.



Gambar 3.1.28 Tampilan halaman penawaran topik dosen

Paraf Pembimbing I:

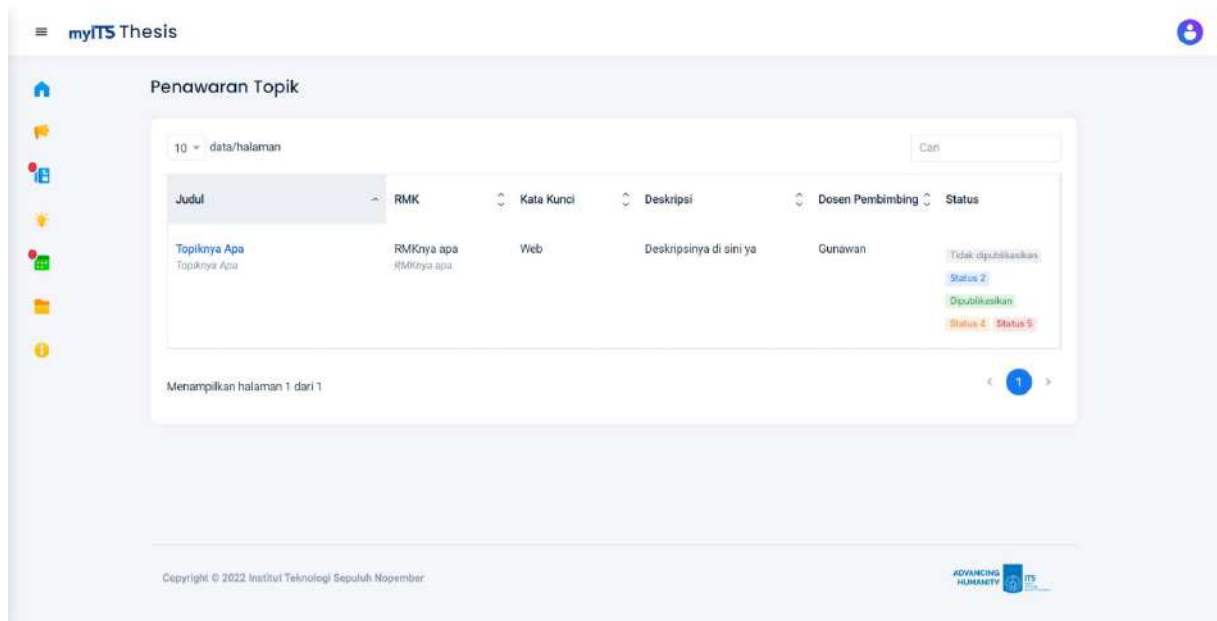
Sugeng

Paraf Pembimbing II:

Gambar 3.1.29 Tampilan halaman membuat penawaran topik

Gambar 3.1.30 Tampilan halaman detail penawaran topik

Selanjutnya untuk mahasiswa hanya diberikan akses untuk melihat daftar penawaran topik pada halaman Penawaran Topik (Gambar 3.1.31). Halaman ini dapat diakses mahasiswa melalui menu *sidebar* yang tersedia. Halaman ini, menampilkan judul, RMK kata kunci deskripsi, dosen pembimbing, sert status penawarn tersebut. Selain itu, mahasiswa juga dapat mencari judul spesifik dengan menggunakan *search bar* yang tersedia dan melihat detail topik dengan klik judul topik yang diinginkan.



Gambar 3.1.31 Tampilan halaman penawaran topik mahasiswa

3.1.2 Perancangan

3.1.2.1 Perancangan Kasus Pengujian

Tahap pertama yang dilakukan dalam mempersiapkan pengujian adalah pembuatan *test case*, yang dimana didapatkan melalui analisis User Story yang telah dirancang dan disepakati. Dengan begitu, dapat diperkirakan berapa lama waktu untuk melakukan pengujian, spesifikasinya, dan juga menghasilkan *test case* yang sesuai dengan kebutuhan bisnis dari myITS Thesis.

Tabel 3.1.2 Tabel Kasus Pengujian

No	Kasus Uji
TC-1001	Mahasiswa dapat melihat halaman beranda, terdapat informasi terkait tugas akhir/tesis /disertasi dan status, serta pengumuman terbaru
TC-1002	Mahasiswa dapat melihat sidebar menu, terdapat menu-menu yang tersedia pada sistem informasi myITS Thesis
TC-1003	Mahasiswa dapat membuat usulan tugas akhir/tesis/disertasi
TC-1004	Mahasiswa dapat mendaftar sidang pada periode yang dibuka
TC-1005	Mahasiswa dapat membuat rekap progres bimbingan
TC-1006	Mahasiswa dapat mengajukan permintaan perubahan data tugas akhir/tesis/disertasi mahasiswa
TC-1007	Mahasiswa dapat melihat riwayat perubahan tugas akhir/tesis/disertasi mahasiswa
TC-1008	Mahasiswa dapat melihat penawaran topik tugas akhir/tesis/disertasi
TC-1009	Mahasiswa dapat melihat pengumuman yang dibuat oleh admin prodi

No	Kasus Uji
TC-1010	Mahasiswa dapat melihat detail pengumuman yang dibuat oleh admin prodi
TC-1011	Mahasiswa dapat melihat dan mengunduh daftar panduan tugas akhir/tesis/disertasi yang dibuat oleh admin prodi mahasiswa
TC-1012	Mahasiswa dapat menyetujui atau menolak permintaan untuk bergabung menjadi anggota tim tugas akhir/tesis/disertasi
TC-1013	Mahasiswa dapat melihat daftar permintaan untuk bergabung menjadi anggota tim tugas akhir/tesis/disertasi pada Tugas Akhir/Tesis/Disertasi
TC-1014	Jika telah melakukan sidang, mahasiswa dapat menjawab revisi tugas akhir/tesis/disertasi
TC-2001	Dosen pembimbing dapat melihat halaman Bimbingan Dosen pembimbing
TC-2002	Dosen pembimbing dapat melihat daftar Permintaan Pembimbingan Tugas Akhir/Tesis/Disertasi
TC-2003	Dosen pembimbing dapat melihat detail permintaan pembimbingan tugas akhir/tesis/disertasi mahasiswa
TC-2004	Dosen pembimbing dapat menyetujui permintaan pembimbingan tugas akhir/tesis/disertasi
TC-2005	Dosen pembimbing dapat menolak permintaan membimbing usulan tugas akhir/tesis/disertasi
TC-2006	Dosen pembimbing dapat menyetujui pengajuan sidang tugas akhir/tesis/disertasi mahasiswa bimbingan
TC-2007	Dosen pembimbing atau dosen penguji dapat memberi nilai tugas akhir/tesis/disertasi mahasiswa
TC-2008	Dosen pembimbing dapat menyetujui progres bimbingan tugas akhir/tesis/disertasi mahasiswa
TC-2009	Dosen pembimbing dapat menyetujui perubahan data tugas akhir/tesis/disertasi mahasiswa bimbingan dosen pembimbing
TC-2010	Dosen penguji dapat menentukan kelulusan mahasiswa ketika sidang
TC-2011	Dosen pembimbing atau dosen penguji dapat memberi revisi terhadap tugas akhir/tesis/disertasi mahasiswa
TC-2012	Dosen pembimbing atau dosen penguji dapat melihat jawaban revisi tugas akhir/tesis/disertasi mahasiswa
TC-2013	Dosen dapat melihat penawaran topik tugas akhir/tesis/disertasi dosen
TC-2014	Dosen dapat membuat penawaran topik tugas akhir/tesis/disertasi
TC-2015	Dosen dapat menyunting penawaran topik tugas akhir/tesis/disertasi dosen
TC-2016	Dosen dapat melihat seluruh topik tugas akhir/tesis/disertasi
TC-2017	Dosen dapat melihat pengumuman yang dibuat oleh admin prodi
TC-2018	Dosen dapat melihat detail pengumuman yang dibuat oleh admin prodi
TC-2019	Dosen dapat melihat daftar panduan tugas akhir/tesis/disertasi yang dibuat oleh

No	Kasus Uji
	admin prodi dosen
TC-2020	Dosen pembimbing atau dosen penguji dapat menyetujui atau menolak revisi tugas akhir/tesis/disertasi mahasiswa
TC-2021	Dosen pembimbing dapat memberikan komentar dan menyetujui bimbingan
TC-3001	Admin prodi dapat melihat halaman beranda milik admin prodi
TC-3002	Admin prodi dapat mengatur boleh/tidaknya mahasiswa berkelompok untuk pengajuan tugas akhir/tesis /disertasi
TC-3003	Admin prodi dapat melihat halaman Pembimbingan Dosen
TC-3004	Admin prodi dapat mengatur kuota bimbingan dosen
TC-3005	Admin prodi dapat melihat daftar Tugas Akhir/Tesis/Disertasi
TC-3006	Admin prodi dapat membuat periode sidang
TC-3007	Admin prodi dapat mengatur jadwal periode sidang
TC-3008	Admin prodi dapat membatalkan sidang
TC-3009	Admin prodi dapat membuat kriteria penilaian tugas akhir/tesis/disertasi dalam prodi admin prodi
TC-3010	Admin prodi dapat membuat pengumuman terkait dengan tugas akhir/tesis/disertasi
TC-3011	Admin prodi dapat melihat pengumuman terkait dengan tugas akhir/tesis/disertasi yang telah admin prodi buat
TC-3012	Admin prodi dapat melihat detail pengumuman yang telah admin prodi buat sebelumnya
TC-3013	Admin prodi dapat mengedit pengumuman yang telah admin prodi buat sebelumnya
TC-3014	Admin prodi dapat membuat panduan tugas akhir/tesis/disertasi terkait
TC-3015	Admin prodi dapat melihat daftar panduan tugas akhir/tesis/disertasi yang telah admin prodi buat
TC-3016	Admin prodi dapat melihat detail panduan tugas akhir yang telah admin prodi buat sebelumnya
TC-3017	Admin prodi dapat merubah panduan tugas akhir yang telah admin prodi buat sebelumnya
TC-4001	Admin sidang dapat mengganti dosen pembimbing

3.1.3 Implementasi

3.1.3.1 Pembuatan Pengujian Otomatis

Kerangka kerja yang digunakan dalam pembangunan pengujian otomatis pada tugas akhir ini adalah Katalon Studio. Dalam pembuatan pengujian otomatis, diperlukan pembuatan *test case* pada Katalon Studio menggunakan mode *script*, *test suite* pada fitur-

fitur yang ingin diuji. Lalu membuat *Step Definition*, yang merupakan sekumpulan langkah yang harus dijalankan untuk menguji kasus uji, menggunakan Bahasa pemrograman Java yang telah disediakan di Katalon Studio. Untuk lebih lengkapnya berikut langkah-langkah yang dilakukan sebelum dan pada saat melakukan implementasi pengujian otomatis menggunakan Katalon Studio.

1. Instalasi Katalon Studio

Katalon Studio diperlukan dalam pembuatan program pengujian otomatis. Alat bantu ini membantu pengujian *end-to-end* pada situs web dengan kompatibilitasnya terhadap *browser* seperti Chrome Firefox, Safari, dan Edge serta mendukung semua kerangka *front-end*, meliputi ReactJS, AngularJS, dan VueJS. Dalam pengujian ini, digunakan Katalon Studio versi 8.5.5

2. Pembuatan *test case* menggunakan mode *script*

Pada mode *script*, telah terdapat *default package* yang di import oleh Katalon Studio. Langkah ini berdasarkan kasus uji yang telah didefinisikan sebelumnya. File *test case* yang dibuat berisikan *script* menggunakan Bahasa pemrograman Java serta kata kunci dan fungsi yang telah terpasang dan disediakan Katalon Studio. Dengan adanya kata kunci tersebut, tidak perlu melakukan konfigurasi WebDriver karena Katalon Studio menggunakan kerangka Selenium webdriver dan menangani kekompleksan melalui kata kunci yang digunakan seperti **WebUI.openBrowser()**. *Test case* ini dibuat dengan metode *data-driven testing* agar dapat mengeksekusi *test suite* yang dinamis.

3. Pembuatan *test suite*

Untuk hasil pengujian yang lebih akurat, *test suite* yang dibuat harus dinamis. *Test suite* sendiri merupakan kumpulan dari *test case* yang telah dibuat sebelumnya dan dapat di kelompokkan menjadi beberapa kategori. *Test suite* digunakan untuk menguji perangkat lunak dalam susunan perilaku tertentu. Eksekusi *test suite* akan dilakukan secara paralel agar pengujian dapat dilakukan secara bersamaan dalam lingkungan berbeda dan mempercepat waktu yang dibutuhkan untuk menguji sistem uji.

4. Pembuatan command eksekusi pengujian otomatis

Pengujian otomatis yang telah dibuat, belum memiliki *command* untuk mengeksekusi melalui *command line*. Dengan menggunakan alat bantu Build CMD pada Katalon Studio, dapat dihasilkan command untuk mengeksekusi. *Command* hanya dapat dihasilkan dari *test suite* dan kumpulan *test suite*. Selain untuk mempercepat eksekusi pengujian karena tidak ada dependensi pada antarmuka Katalon Studio, *Command* yang didapatkan akan digunakan untuk integrasi dengan sistem lainnya, salah satunya menjalankan pengujian pada Jenkins.

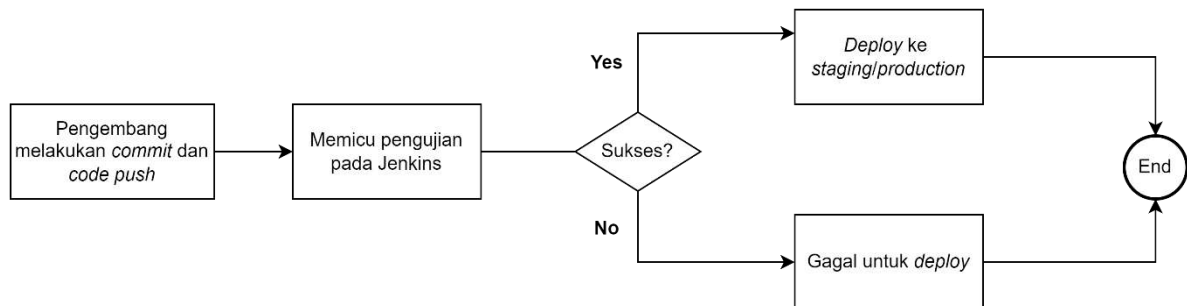
Langkah-langkah diatas merupakan implementasi pengujian otomatis yang dilakukan pada tugas akhir ini. Pengujian otomatis menggunakan Katalon Studio tentunya dapat dijalankan dengan langkah dan metode yang berbeda dalam mencapai tujuan yang sama, sesuai dengan kebutuhan dan tujuan yang diinginkan.

3.1.3.2 Integrasi Pengujian Otomatis pada CI/CD pipeline

Setelah pengujian otomatis yang dirancang telah selesai, tahap selanjutnya merupakan pengintegrasian *script* pengujian otomatis pada pipeline CI/CD menggunakan Jenkins. Agar saat tim pengembang melakukan pembangunan aplikasi baru, pengujian akan secara

otomatis tereksekusi ketika adanya perubahan pada kode program. Hal tersebut dapat dilakukan dengan menghubungkan Jenkins dengan *repository* pengujian otomatis. Pada tahap ini, akan terjadi perubahan pada *workflow CI/CD pipeline* myITS Thesis yang telah ada, dengan ditambahkannya pengujian otomatis sebelum dilakukannya perilis.

Sistem akan menguji dan mengetahui apakah terdapat kegagalan pada pengujian. Jika hasil pengujian gagal, ada kegagalan pada kode program dan tidak dapat ditampilkan atau dirilis pada lingkungan staging ataupun *production*. Pengujian otomatis yang dijalankan pada Jenkins akan meliputi semua sistem, dengan waktu yang relatif lebih singkat. Aplikasi akan teruji secara menyeluruh walaupun perubahan yang terjadi hanya pada sebagian sistem. Dengan begitu, aplikasi yang akan diterima pengguna tidak memiliki *error* atau kesalahan.



Gambar 3.1.32 Alur Pengujian pada Jenkins

Untuk mencapai alur pengujian yang diinginkan pada Jenkins, berikut Langkah-langkah yang dilakukan untuk mengintegrasikan pengujian otomatis Katalon Studio.

1. Install Jenkins

Jenkins merupakan server automasi untuk mengimplementasikan CI/CD dan memudahkan proses *testing*, *building*, dan *deployment* secara otomatis. Pada pelaksanaan tugas akhir ini, digunakan Jenkins versi 2.383. Langkah untuk mengunduh dan *setup* Jenkins dapat ditemukan di internet.

2. Membuat *Job* pada Jenkins

Pada tugas akhir ini, diperlukan pembuatan *Job* pada Jenkins untuk mendeteksi terjadinya perubahan pada kode sumber saat pengembang melakukan *commit*, dan untuk menjalankan pengujian otomatis pada Jenkins. Jenkins akan memanggil *command* yang telah dibuat untuk menjalankan pengujian otomatis.

3.2 Peralatan pendukung

Pengujian otomatis yang dilakukan menggunakan alat bantu Katalon Studio. Katalon Studio memiliki fitur yang kompleks tetapi mudah digunakan bahkan untuk seorang pemula. Selain itu, Katalon Studio sendiri menggunakan *keyword driven approach* dan *data-driven testing* sebagai metode utama dalam menuliskan pengujian otomatis dengan data *driven functionality approach* untuk melakukan eksekusi pengujian. Untuk proses integrasi pengujian otomatis pada CI/CD, digunakan alat bantu integrasi Jenkins. Dengan alat bantu tersebut, pengujian dapat dieksekusi setelah adanya proses *commit* atau sesuai jadwal yang diinginkan, dan melakukan integrasi pengujian pada proses *Continuous Integration* (CI).

3.3 Rencana Implementasi dan Uji Coba

Implementasi yang dilakukan pada tugas akhir ini berupa *script* pengujian otomatis yang berisikan *Test Case* dan *Test Suite*. *Script* tersebut lalu diintegrasikan menggunakan alat bantu Jenkins pada CI/CD *pipeline* myITS Thesis. Dengan begitu, setiap terjadi perubahan kode program yang dilakukan oleh pengembang, Jenkins akan memicu pengujian otomatis dan menjalankannya. Pengujian ini dilakukan untuk memastikan aplikasi yang akan diberikan kepada pengguna berjalan dengan baik dan tidak ada kegagalan atau *error*.

JADWAL KEGIATAN

Tahap																								
	November				Desember				Januari				Februari				Maret				April			
Penyusunan Proposal																								
Studi Pustaka																								
Analisis dan perancangan pengujian																								
Implementasi																								
Pengujian dan evaluasi																								
Penyusunan laporan Tugas Akhir																								

Paraf Pembimbing I: 

Paraf Pembimbing II:

DAFTAR PUSTAKA

- Arachchi, S. A. I. B. S., & Perera, I. (2018, May). Continuous integration and continuous delivery pipeline automation for agile software project management. In *2018 Moratuwa Engineering Research Conference (MERCon)* (pp. 156-161). IEEE.
- Bühler, O., & Wegener, J. (2008). Evolutionary functional testing. *Computers & Operations Research*, 35(10), 3144-3160.
- Cohn, M. (2004). User stories applied: For agile software development. Addison-Wesley Professional.
- Ereiz, Z. (2019). Automating Web Application Testing Using Katalon Studio. *Zbornik radova Međunarodne naučne konferencije o digitalnoj ekonomiji DIEC*, 2(2), 87-97.
- Gușeilă, L. G., Bratu, D. V., & Moraru, S. A. (2019, August). Continuous testing in the development of iot applications. In *2019 International Conference on Sensing and Instrumentation in IoT Era (ISSI)* (pp. 1-6). IEEE.
- Howden, W. E. (1980). Functional program testing. *IEEE Transactions on Software Engineering*, (2), 162-169.
- Jorgensen, P. C. (2013). *Software testing: a craftsman's approach*. Auerbach Publications.
- Lucassen, G., Dalpiaz, F., van der Werf, J. M. E., & Brinkkemper, S. (2016, March). *The use and effectiveness of user stories in practice*. In *International working conference on requirements engineering: Foundation for software quality* (pp. 205-222). Springer, Cham.
- Mohammad, S. M. (2015). Automation Testing in Information Technology. *International Journal of Creative Research Thoughts (IJCRT)*, ISSN, 2320-2882.
- Moutsatsos, I. K., Hossain, I., Agarinis, C., Harbinski, F., Abraham, Y., Dobler, L., ... & Parker, C. N. (2017). Jenkins-CI, an open-source continuous integration system, as a scientific data and image-processing platform. *SLAS DISCOVERY: Advancing Life Sciences R&D*, 22(3), 238-249.
- Nguyen, H. Q. (2001). *Testing applications on the Web: Test planning for Internet-based systems*. John Wiley & Sons.
- Prakoso, B., & Sujarwo, A. (2022). Perancangan Automated Testing Pada Studi Kasus Website Indicar. *Jurnal Ilmiah Teknologi Informasi dan Robotika*, 4(1), 29-32.
- Prasetyo, Arief. (2021). *Implementasi Automated Functional UI Testing Untuk Mencapai Konsep Ci/Cd Terintegrasi Jenkins Menggunakan Framework Serenity Dengan Page Object Model Design Pattern Pada Ppdb Jatim*. (Skripsi Sarjana, Institut Teknologi Sepuluh Nopember).
- Ragunath, P. K., Velmourougan, S., Davachelvan, P., Kayalvizhi, S., & Ravimohan, R. (2010). Evolving a new model (SDLC Model-2010) for software development life cycle (SDLC). *International Journal of Computer Science and Network Security*, 10(1), 112-119.
- Seth, N., & Khare, R. (2015, December). ACI (automated Continuous Integration) using Jenkins: Key for successful embedded Software development. In *2015 2nd International Conference on Recent Advances in Engineering & Computational Sciences (RAECS)* (pp. 1-6). IEEE.

- Sharma, R. M. (2014). Quantitative analysis of automation and manual testing. *International journal of engineering and innovative technology*, 4(1).
- Sharma, M., & Angmo, R. (2014). Web based automation testing and tools. *International Journal of Computer Science and Information Technologies*, 5(1), 908-912.
- Smart, J. F. (2011). Jenkins: The Definitive Guide: Continuous Integration for the Masses. " O'Reilly Media, Inc."
- Tjandra, S., Maryati, I., & Theopilus, J. (2021). Automated Software Testing for Multi Platform Applications using Katalon.
- Tuteja, M., & Dubey, G. (2012). A research study on importance of testing and quality assurance in software development life cycle (SDLC) models. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(3), 251-257.