**PERANCANGAN SISTEM INFORMASI PENGAJUAN JUDUL TUGAS AKHIR MENGGUNAKAN *CODEIGNITER* BERBASIS *WEBSITE* PADA UNIVERSITAS MUHAMMADIYAH KENDARI**

****

**ARMADANI**

**NIM. 21916099**

**SKRIPSI**

*Diajukan Sebagai Salah Satu Syarat*

*Untuk Memperoleh Sarjana Pendidikan*

**PROGRAM STUDI PENDIDIKAN TEKNOLOGI INFORMASI**

**FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN**

**UNIVERSITAS MUHAMMADIYAH KENDARI**

**­KENDARI**

**2023**





# HALAMAN PERSETUJUAN PEMBIMBING

Telah diperiksa dan disetujui oleh Pembimbing I dan Pembimbing II untuk dipertahankan di hadapan panitia Ujian Skripsi pada Program Studi Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nama | : | Armadani | | |
| Nim | : | 21916099 | | |
| Judul | : | Perancangan Sistem Informasi Pengajuan Judul Tugas Akhir Menggunakan Codeigniter Berbasis Website Pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari | | |
| **Menyetujui,** | | | | |
| **Pembimbing I** | | |  | **Pembimbing II** |
|  | | |  |  |
| **Alfiah Fajriani, ST., M.Eng**  **NIDN. 0919029203** | | |  | **Zila Razilu, S.Pd., M.Pd**  **NIDN. 0922059102** |
|  | | |  |  |
| **Mengetahui,** | | | | |
| **Ketua Program Studi Pendidikan Teknologi Informasi**  **Fakultas Keguruan Dan Ilmu Pendidikan**  **Universitas Muhammadiyah Kendari** | | | | |
|  | | | | |
|  | | | | |
| **Zila Razilu, S.Pd., M.Pd**  **NIDN. 0922059102** | | | | |
|  | | | | |
| **HALAMAN PENGESAHAN SKRIPSI**  **PERANCANGAN SISTEM INFORMASI PENGAJUAN JUDUL TUGAS AKHIR MENGGUNAKAN *CODEIGNITER* BERBASIS *WEBSITE* PADA UNIVERSITAS MUHAMMADIYAH KENDARI**  https://3.bp.blogspot.com/-ueDHROmHXg4/UuRPl6DRXWI/AAAAAAAAAZg/IJN3AwuMYYY/s400/Universitas+Muhammadiyah+Kendari.png**Disusun Oleh :**  **ARMADANI**  **21916099**  Telah dipertahankan dihadapan Panitia Ujian Skripsi pada Program Studi Pendidikan Teknologi Informasi pada Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari guna memperoleh gelar Sarjana Pendidikan Strata Satu (S.Pd) pada hari Selasa tanggal 22 Bulan Mei Tahun 2023 dan dinyatakan lulus dengan Pujian.  **PANITIA UJIAN**  Ketua : Hendra Nelva Saputra, S.Pd., M.Pd. (………………………)  Sekretaris : Zila Razilu, S.Pd., M.Pd (………………………)  Anggota I : Alfiah Fajriani, ST., M.Eng. (………………………)  Penguji Tamu : Darman, S.Pd.,M.Pd (………………………)  Kendari, Mei 2023  Disahkan Oleh,  Dekan Fakultas Keguruan dan Ilmu Pendidikan  Universitas Muhammadiyah Kendari  **Tri Indah Rusli, S.Pd., M.Pd**  **NIDN. 0907068602** | | | | |

**PERNYATAAN KEASLIAN TULISAN**

Yang bertanda tangan di bawah ini :

Nama : Armadani

Stambuk : 21916099

Program Studi : Pendidikan Teknologi Informasi

Fakultas : Keguruan dan Ilmu Pendidikan

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan jiplakan atau plagiat dari tulisan orang lain. Apabila dikemudian hari terbukti atau dibuktikan hasil jiplakan atau plagiat, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Kendari, Juni 2023

Yang Membuat Pernyataan

Armadani

21916099

**ABSTRAK**

**Armadani: 21916099** Perancangan Sistem Informasi Pengajuan Judul Tugas Akhir (SIMPELTA PTI) padaProgram Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari yang dibimbing oleh 1). Ibu Alfiah fajriani.,ST.M.eng, 2). Ibu Zila Razilu.,S.Pd.,M.Pd Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Muhammadiyah Kendari (UMK).

Rumusan penelitian ini adalah bagaimana membangun Sistem Informasi Pengajuan Judul Tugas Akhir (SIMPELTA PTI) padaProgram Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari berbasis *website*? Sedangkan tujuan dalam penelitian ini adalah membangun Sistem Informasi Pengajuan Judul Tugas Akhir (SIMPELTA PTI) pada Universitas Muhammadiyah Kendari berbasis *website*.

Metode yang digunakan dalam penelitian ini adalah model *waterfall* dimana dalam medel akan dilakukan sebuah perancangan sistem secara berurutan. Secara konseptual perancangan sistem informasi terdiri dari: 1).Analisis kebutuhan (*analyzing*), 2).Desain (*design),* 3).Penulisan kode (*coding*), 4).Pengujian system (*testing*), 5).Implementasi (*implementation*) (Denis, 2021).

Hasil penelitian ini adalah Sistem Informasi Pengajuan Judul Tugas Akhir berbasis web kemudian diimplementasikan di Program Studi Pendidikan Teknologi Informasi FKIP, Universitas Muhammadiyah Kendari, SIMPELTA PTI ini dinyatakan layak berdasarkan hasil pengujian yang berfokus pada fungsionalitas dari semua fitur menggunakan *Black box*.

**Kata Kunci : penelitian, pengajuan judul, simpelta pti**

***ABSTRACT***

**Armadani: 21916099** *The design of the Final Project Title Submission Information System (SIMPELTA PTI) in the Information Technology Education Study Program at Muhammadiyah Kendari University which is supervised by 1). Mrs. Alfiah fajriani, ST.M.eng, 2). Mrs. Zila Razilu, S.Pd, M.Pd, Faculty of Teacher Training and Education (FKIP), University of Muhammadiyah Kendari (UMK).*

*The formulation of this research is how to build a website-based Final Project Title Submission Information System (SIMPELTA PTI) in the Information Technology Education Study Program at Muhammadiyah Kendari University? While the purpose of this research is to build a website-based Final Project Title Submission Information System (SIMPELTA PTI) at Universitas Muhammadiyah Kendari.*

*The method used in this research is the waterfall model in which a system design will be carried out sequentially. Conceptually the design of the information system consists of: 1).Analysis of needs (analyzing), 2).Design (design), 3).Writing code (coding), 4).System testing (testing), 5).Implementation (implementation) (Denis, 2021).*

*The results of this study are a web-based Final Project Title Submission Information System then implemented in the FETT Information Technology Education Study Program, Universitas Muhammadiyah Kendari, SIMPELTA PTI is declared feasible based on the results of testing that focuses on the functionality of all features using Black box.*

***Keywords: Research, title submission, and Simpelta PTI***

**KATA PENGANTAR**

***Assalamu’alaikum Warahmatullahi Wabarakaatuh.***

Alhamdulillahhirobbil’alamin, dengan memanjatkan puji serta syukur atas kehadirat Allah SWT, karena atas segala rahmat dan karunia-Nya, Penulis masih diberikan kekuatan sekuat ahlusunnah waljamaah dan kepastian manjadda wajada, akhirnya penulis dapat menyelesaikan penyusunan skripsi ini. Selanjutnya Shalawat serta salam tak lupa penulis haturkan kepada junjungan Alam Rosulullah SAW, atas perjuangan dari beliau yang telah membawa umat islam dari zaman jahiliah menuju zaman yang penuh dengan ilmu pengetahuan seperti zaman saat ini. Alhamdulillah atas hidayah dan inayah dari ALLAH SWT, penulis dapat menyelesaikan penyusunan skripsi ini yang berjudul :

**“Perancangan Sistem Informasi Pengajuan Judul Tugas Akhir Menggunakan *Codeigniter* Berbasis *Website* Pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari”.** Yang merupakan syarat dalam rangka menyelesaikan studi untuk menempuh gelar sarjana Pendidikan di Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari.

Penulis menyadari bahwa penulisan skripsi ini masih jauh dari kata sempurna, hal itu karena keterbatasan kemampuan yang dimiliki penulis. Dalam penyusunan skripsi ini, penulis banyak mendapatkan pelajaran, dukungan motivasi, bantuan berupa bimbingan yang sangat berharga dari berbagai pihak dari pelaksanaan hingga penyusunan skripsi ini.

Pada kesempatan ini, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada orang-orang yang penulis hormati dan cintai yang membantu secara langsung maupun tidak langsung selama pembuatan skripsi ini. Terutama kepada Orangtua penulis, yaitu **Ibu Maldia** dan **Bapak Arzuman**  yang berada di Kabupaten Buton Tengah, karena atas perjuangan serta doa darinya, sehingga penulis selalu semangat, selalu tersenyum, selalu ulet, selalu tekun, selalu giat olahraga, selalu yakin pada kepastian Allah SWT Man Jadda Wajada, dan selalu berjuang dengan penuh harapan tinggi untuk membahagiakan orang tua penulis sehingga penulis berharap orang tua penulis dapat tersenyum bahagia dimasa tuanya nanti. Selanjutnya ucapan terima kasih penulis ucapkan Untuk Kakakku tercinta yang selalu memberikan do’a dan semangat yaitu Muh. Sandi, S.Pd yang selalu memberikan semangat dan ketenangan akan material untuk terus kuliah dan fokus pada perkuliahan, untuk Adikku Muhammad Fadil Arzuman dan Zulfikar yang selalu membuat senyuman bahagia ketika melihatnya.

Penulis juga mengucapan Terima kasih yang sedalam-dalamnya kepada Ibu Alfiah Fajriani, ST., M.Eng sebagai pembimbing I, karena atas waktunya untuk memberikan bimbingan, arahan, kepada penulis dalam penyelesaian skripsi ini. dan juga kepada Ibu Zila Razilu, S.Pd., M.Pd, sebagai pembimbing II, terima kasih atas waktunya untuk memberikan bimbingan, arahan, serta memberikan saran kepada penulis dalam penyelesaian skripsi ini.

Pada kesempatan ini, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu serta membimbing penulis, sehingga penyusun skripsi ini dapat terselesaikan tepat waktu, yaitu :

1. Prof. Dr. Ir. Muhammad, M.Sc., IPU., ASEAN Eng selaku Rektor Universitas Muhammadiyah Kendari
2. Tri Indah Rusli, S.Pd., M.Pd selaku Dekan FKIP Universitas Muhammadiyah Kendari.
3. Dra. Hasmira Said, M.Pd selaku Wakil Dekan FKIP Universitas Muhammadiyah kendari.
4. Zila Razilu,S.Pd., M.Pd sebagai ketua Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari.
5. Zila Razilu, S.Pd., M.Pd sebagai pembimbing akademik selama 8 semester.
6. Bapak Muhammad Akbar Yasin, ST., M.Eng. sebagai penguji sistem.
7. Para Dosen tetap pada Prodi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari.
8. Bapak Hendra Nelva Saputra, S.Pd., M.Pd selaku ketua penguji penulis yang telah memberikan saran dan masukan untuk penulis.
9. Bapak Darman, S.Pd.,M.Pd selaku penguji tamu penulis yang telah memberikan saran dan masukan untuk penulis.
10. Alm. Bapak Samusu, S.Pd.,M.Pd selaku dosen tetap Program Studi Pendidikan Teknologi Informasi yang telah memberikan saran dan masukan untuk penulis.
11. Bapak Fahsini Selaku Staf/Pengelola pada Program Studi Pendidikan Teknologi Informasi, yang telah memberikan pelayanan Administrasi yang begitu baik.
12. Seluruh Staf pegawai Akademik Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari yang telah memberikan pelayanan Administrasi yang begitu baik.
13. Ucapan Terima kasih Kepada Keluarga Besar Kakek La Bagea., Keluarga Besar Kakek Alm. La Amala., Keluarga Besar Paman Briptu Deni Dahlan, S.H., MH., Keluarga Besar Paman Dr. Ishak Bagea, S.Pd.,M.A., Keluarga Besar Paman Arfili, S.Pd., M.A., Keluarga Besar Paman Arifin Bagea, S.Pd.,M.A, dan keluarga besar lainnya lagi yang tidak dapat penulis sebutkan, penulis mengucapkan terima kasih atas pengajaran, pembelajaran arti kehidupan, nasihat, bimbingan ilmu, akhlak dan arti kekeluargaan sehingga penulis bisa menjadi seperti saat ini.
14. Terima kasih kepada Bapak Zulqifli Hedrianto Tahir, ST Selaku CEO Techno’s Studio Kendari serta seluruh karyawan, Ibu Linda Selaku CEO Dokter PC Surabaya serta seluruh karyawan, penulis mengucapkan banyak terima kasih atas pengajaran, pembelajaran, nasihat, bimbingan ilmu dan rasa kekeluargaannya sehingga penulis bisa menjadi seperti saat ini.
15. Terima kasih kepada teman-teman kelas C angkatan 2019, teman-teman Teman-teman Hippmas Raya KDI, Teman-teman Alumni SMK N 1 Mawasangka Tengah, Teman-teman GAS (Generasi Asam) dan lainnya yang tercinta yang senantiasa mendukung dan berjuang bersama, semasa perkuliahan. Terima kasih untuk kenangan indah yang telah diberikan selama empat tahun ini.
16. Ucapan terima kasih penulis ucapkan untuk sepupu-sepupu penulis antara lain: Bripda Muhammad. Izzatul Nuhsin, S.T., Rakhmat Adam, S.H., M.H., Bripda M. Azmil Ramadhan, Bripda Madin, Bripda Harmin, Helida Purwanti, S.Tr.,Kes., Irmawati, Amd.,Ft., Sarti Muliati S.Biotek. Dan lain-lainnya yang penulis tidak bisa sebut satu persatu, penulis mengucapkan terimakasih atas bantuan dan rasa kekeluargaannya.
17. Ucapan terima kasih penulis atas semua pihak yang tidak dapat penulis tuliskan namanya satu persatu yang telah berjasa dalam kehidupan penulis.

Akhir kata, penulis mengharapkan semoga tujuan dari pembuatan skripsi ini dapat tercapai sesuai dengan yang diharapkan, apabila terjadi suatu kesalahan dan kekurangan, penulis meminta maaf dan semoga kita semua dalam lindungan Allah SWT Tuhan yang maha kuasa.

***Wassalamu’alaikum warahmatullahi wabarakaatuh.***

Kendari, Mei 2023

Penulis

Armadani

Nim: 21916099

**DAFTAR ISI**

HALAMAN JUDUL i

HALAMAN LOGO ii

HALAMAN PERSETUJUAN iii

HALAMAN PENGESAHAN SKRIPSI iv

PERNYATAAN KEASLIAN TULISAN v

ABSTRAK vi

ABSTRACT vii

KATA PENGANTAR viii

DAFTAR ISI xiii

DAFTAR GAMBAR xv

DAFTAR TABEL xvii

DAFTAR LAMPIRAN xviii

**BAB I PENDAHULUAN 1**

A. Latar Belakang 1

B. Rumusan Masalah 5

C. Batasan Masalah 5

D. Tujuan 6

E. Manfaat 6

**BAB II LANDASAN TEORI 7**

A. Sistem Informasi 7

B. *Website* 10

1. *Hypertext Markup Language (HTML)* 10

` 2. *Cascading Style Sheet (CSS)* 11

3. *JavaScript* 12

4. *PHP: Hypertext Preprocessor (PHP)* 13

C. *Framework* 14

1. *Framework CodeIgniter* 15

` 2. *Framework Bootstrap* 18

D. *My Structured Query Language (MySQL)* 20

E. *Unified Modeling Language (UML)* 21

1. *Use Case* Diagram 21

` 2. *Class* Diagram 23

3. *Sequence* Diagram 24

F. Penelitian Relevan 24

**BAB III METODE PENELITIAN 30**

A. Jenis Penelitian 30

B. Prosedur Penelitian 30

1. Analisis Kebutuhan 31

` 2. Desain Sistem 36

3. Penulisan Kode 50

4. Pengujian 50

5. Implementasi 52

C. Tempat dan Waktu 52

D. Alat dan Bahan 52

**BAB IV Hasil Penelitian dan Pembahasan 54**

A. Hasil Penelitian 54

1. Implementasi SIMPELTA Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari 54

` 2.*Database)* 54

3.TampilanSIMPELTA Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari 62

B. Pengujian Sistem 80

**BAB V PENUTUP 102**

A. Kesimpulan 102

B. Saran 103

DAFTAR PUSTAKA 104

LAMPIRAN 107

DAFTAR RIWAYAT HIDUP 108

**DAFTAR GAMBAR**

**Gambar 2.1** Sistem dan Lingkungannya (Punjul, 2016) 8

**Gambar 2.2** *Application Flowchart* 17

**Gambar 2.3** Simbol *Sequence* Diagram 24

**Gambar 3.1** Metode *waterfall* (Dennis, 2021) 30

**Gambar 3.2** *Use Case**Admin* Mengelola *Menu* *Website* Pengajuan Judul 36

**Gambar 3.3** *Use Case* *Admin* Mengelola *Menu* *Website* Pengajuan Judul 37

**Gambar 3.4** *Use Case* *Admin* *Menu* Persetujuan Berkas 37

**Gambar 3.5** *Use Case* *Admin* *Menu* *View* Data 37

**Gambar 3.6** *Use Case* *Admin* Mengelola *Menu* *Repor*t 37

**Gambar 3.7** *Use Case* Kaprodi Mengelola *Menu* *Website* 38

**Gambar 3.8** *Use Case* Kaprodi Mengelola *Menu* Master Data 38

**Gambar 3.9** *Use Case* Kaprodi Mengelola *Menu* Persetujuan Berkas 38

**Gambar 3.10** *Use Case* Kaprodi *Menu* Persetujuan Judul Skripsi 38

**Gambar 3.11***Use Case* Kaprodi *Menu* *View* Data 39

**Gambar 3.12** *Use Case* Kaprodi Mengelola *Menu Report* 39

**Gambar 3.13** *Use Case* Dosen *Menu Website* 39

**Gambar 3.14** *Use Case* Dosen *Menu* Mahasiswa BimbinganData 39

**Gambar 3.15** *Use Case* Mahasiswa *Menu* Website 40

**Gambar 3.16** Use Case Mahasiswa *Menu* Pengajuan Judul Data 40

**Gambar 3.17** *Activity* Diagram *Admin* *Menu* *Login* Pengguna 40

**Gambar 3.18** *Activity* Diagram *Admin* *Menu* Pengajuan Judul 41

**Gambar 3.19** *Activity* Diagram Kaprodi *Menu* Pengajuan Judul 41

**Gambar 3.20** *Activity* Diagram Dosen *Menu* Pengajuan Judul 42

**Gambar 3.21** *Activity* Diagram Mahasiswa *Menu* Pengajuan Judul 42

**Gambar 3.22** *Sequence* Diagram *Login* Pengguna 43

**Gambar 3.23** *Sequence* Diagram *Admin* *Menu* Pengajuan Judul 43

**Gambar 3.24** *Sequence* Diagram Kaprodi *Menu* Pengajuan Judul 43

**Gambar 3.25** *Sequence* Diagram Dosen *Menu* Pengajuan Judul 44

**Gambar 3.26** *Sequence* Diagram Mahasiswa *Menu* Pengajuan Judul 44

**Gambar 3.27** *Entity Relationship Diagram* 48

**Gambar 3.28** *User Interface* Tampilan *Menu* Utama 49

**Gambar 3.29** *User Interface* Tampilan *Menu Login* 50

**Gambar 3.30** *User Interface* Tampilan *Menu* *Dashboard* 50

**Gambar 4.1** Tampilan Laragon 55

**Gambar 4.2**TampilanTabel *User* 55

**Gambar 4.3** TampilanTabel *Roles* 56

**Gambar 4.4** TampilanTabel *Lecturers* 57

**Gambar 4.5** TampilanTabel *Students* 57

**Gambar 4.6** TampilanTabel *Majors* 58

**Gambar 4.7** Tampilan Tabel *Category* 59

**Gambar 4.8** TampilanTabel *Field Of Studies* 59

**Gambar 4.9** TampilanTabel *Essays* 60

**Gambar 4.10** TampilanTabel *Titles* 61

**Gambar 4.11** Tampilan Tabel *Files* 61

**Gambar 4.12** Tampilan *Login* 62

**Gambar 4.13** Tampilan *Menu Dashboard* 63

**Gambar 4.14** Tampilan *Menu* Data *User’s* 64

**Gambar 4.15** Tampilan *Menu* DataDosen 65

**Gambar 4.16** Tampilan *Menu Data* Mahasiswa 66

**Gambar 4.17** Tampilan *Menu* DataKategori 67

**Gambar 4.18** Tampilan *Menu* DataBidang Studi 68

**Gambar 4.19** Tampilan *Menu* Data Jurusan 69

**Gambar 4.20** Tampilan *Menu* PersetujuanBerkas 70

**Gambar 4.21** Tampilan *Menu View* JudulTA *By* Kategori 71

**Gambar 4.22** Tampilan *Menu View* JudulTA *By* TahunAjuan 72

**Gambar 4.23** Tampilan *Menu View* Pembimbing 73

**Gambar 4.24** Tampilan *Menu Report* JudulDiterima 74

**Gambar 4.25** Tampilan *Menu Report* JudulDitolak 75

**Gambar 4.26** Tampilan *Menu* Pengajuan/PersetujuanJudul TA 76

**Gambar 4.27** Tampilan *Menu Beranda* Dosen 77

**Gambar 4.28** Tampilan *Menu* MahasiswaBimbingan 78

**Gambar 4.29** Tampilan *Menu* Judul TA *By* Kategori 79

**Gambar 4.30** Tampilan *Menu* Judul TA *By* Tahun Ajuan 80

**Gambar 4.31** Tampilan *Menu Beranda* Mahasiswa 80

**Gambar 4.32** Tampilan *Menu* PersyaratanBerkas 81

**Gambar 4.33** Tampilan *Menu* PengajuanJudulTAMahasiswa 82

**Gambar 4.34** Tampilan *Menu View* DataDosen 83

**Gambar 4.35** Tampilan *Menu* Judul TA *By* Kategori 84

**Gambar 4.36** Tampilan *Menu* Judul TA *By* Tahun Ajuan 85

**DAFTAR TABEL**

**Tabel 2.1** Simbol *Use Case Diagram* (Hermawan, 2017) 22

**Tabel 2.2** Simbol Pada *Class Diagram* 23

**Tabel 3.1** Evaluasi Sistem 33

**Tabel 3.2** Tabel *User* 45

**Tabel 3.3** Tabel *Roles* 45

**Tabel 3.4** Tabel *Lecturers* 45

**Tabel 3.5** Tabel*Students* 45

**Tabel 3.6** Tabel*Titles* 46

**Tabel 3.7** Tabel *Essays* 46

**Tabel 3.8** Tabel*Field Of Studies* 46

**Tabel 3.9** Tabel*Majors* 46

**Tabel 3.10** Tabel*Categories* 47

**Tabel 3.11** Tabel*Files* 47

**Tabel 3.12** Kisi-Kisi Uji *Black Box Testing* 51

**Tabel 4.1** Uji *Black Box Testing* Fitur *Login* 86

**Tabel 4.2** Uji *Black Box Testing* Fitur *Dashboard/Beranda* 86

**Tabel 4.3** Uji *Black Box Testing* Fitur User 87

**Tabel 4.4** Uji *Black Box Testing* Fitur Dosen 88

**Tabel 4.5** Uji *Black Box Testing* Fitur Mahasiswa 89

**Tabel 4.6** Uji *Black Box Testing* Fitur Kategori 91

**Tabel 4.7** Uji *Black Box Testing* Fitur Bidang Studi 92

**Tabel 4.8** Uji *Black Box Testing* Fitur Jurusan 93

**Tabel 4.9** Uji *Black Box Testing* Fitur *View* Data Judul *By* Kategori 95

**Tabel 4.10** Uji *Black Box Testing* Fitur *View* Data Judul *By* Tahun Ajuan 96

**Tabel 4.11** Uji *Black Box* Testing Fitur *View* Data Pembimbing 97

**Tabel 4.12** Uji *Black Box Testing* Fitur *Report* Judul TA Diterima 97

**Tabel 4.13** Uji *Black Box Testing* Fitur *Report* Judul TA Ditolak 98

**Tabel 4.14** Uji *Black Box Testing* Fitur Persetujuan Berkas 98

**Tabel 4.15** Uji *Black Box* *Testing* Fitur Persetujuan Judul TA 99

**Tabel 4.16** Uji *Black Box* *Testing* Fitur Mahasiswa Bimbingan 101

**Tabel 4.17** Uji *Black Box* *Testing* Fitur Pengajuan Judul TA 102

**Tabel 4.18** Uji *Black Box* *Testing* Fitur Pengajuan Berkas Persyaratan 105

**DAFTAR LAMPIRAN**

|  |  |  |
| --- | --- | --- |
| **Lampiran** | **Nama Lampiran** | **Halaman** |
| 1 | *STANDARD OPERATING PROCEDURE* (SOP) | 113 |
| 2 | SURAT IZIN PENELITIAN | 114 |
| 3 | DOKUMENTASI | 115 |

**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

Kemajuan teknologi informasi yang begitu pesatnya, dapat mempermudah serta mempercepat dalam hal penyampaian informasi yang akurat dan tepat di berbagai bidang kehidupan seperti bidang kesehatan, ekonomi serta pendidikan. Dalam pelaksanaannya pada bidang pendidikan telah memberikan kontribusi yang sangat besar untuk mengimplementasikan sistem informasi di berbagai aktivitas kehidupan lainnya. Selain itu, dengan banyaknya aktivitas di dalam lingkungan akademik yang terus bertambah maka dibutuhkan sebuah sistem informasi yang lebih efektif dan efisien untuk mengefektifkan sebuah pekerjaan *stakeholder* pada lingkungan akademik. Seperti yang dikemukakan oleh (Satyahadi, 2019) Sistem informasi pada bidang pendidikan merupakan suatu perkumpulan orang, *software, hardware*, prosedur, dan aturan yang saling bekerja sama dan terintegrasi satu sama lain untuk mencapai efektivitas proses pendidikan.

Bidang pendidikan memerlukan data dan informasi sebagai sumber daya yang sangat penting dalam pengelolaan dan manajemen data pada suatu perguruan tinggi negeri maupun swasta. Oleh karena itu efisiensi dan kecepatan dalam pengelolaan serta penyampaian data dan informasi pada perguruan tinggi adalah salah satu faktor pendukung untuk meningkatkan kualitas di dalam pelayanan akademik. Karena itulah perancangan dan pembangunan sebuah sistem informasi yang tepat sesuai dengan tahapan prosedur di dalam perguruan tinggi harus mampu meningkatkan kinerja pelayanan pada faktor pendukung lainnya sehingga dapat mewujudkan suatu peningkatan kualitas pelayanan.

Perguruan tinggi kualitas pelayanan akademik yang baik dapat diukur dari banyaknya lulusan dan ketepatan waktu dalam menyelesaikan studi. Hal ini disebabkan karena pengelolaan data dan informasi yang relevan tentang pelayanan melalui sistem informasi yang disampaikan oleh *civitas* akademik untuk mendapatkan suatu informasi akademik yang dibutuhkan bagi *civitas* akademik terkait pada institusi pendidikan tersebut. Dikarenakan Informasi yang berkualitas adalah informasi yang dapat disajikan secara relevan, akurat, dan tepat waktu sehingga dapat mempermudah *civitas* akademika.

Program Studi Pendidikan Teknologi Informasi (Prodi PTI) merupakan salah satu program studi yang ada di Universitas Muhammadiyah Kendari. Prodi PTI dalam kurikulumnya memiliki mata kuliah tugas akhir (skripsi) sebagai salah satu mata kuliah wajib yang harus ditempuh oleh setiap mahasiswa untuk bisa mendapatkan gelar strata-1 (S1), yang mana menurut (Jalal, 2012) tugas akhir merupakan karya ilmiah yang harus ditulis mahasiswa sebagai syarat untuk memperoleh gelar strata-1 (S1). Kemudian menurut (Wardani, 2013) mahasiswa diharapkan mampu merangkum dan mengaplikasikan semua pengalaman pendidikannya untuk memecahkan masalah dalam bidang keahlian tertentu secara sistematis, kritis, logis, dan berbobot, yang berlandaskan dari data serta informasi yang akurat.

Pembuatan sebuah tugas akhir memerlukan waktu yang cukup lama serta tahapan yang panjang, salah satu tahapan awalnya yaitu pengajuan judul tugas akhir di Program Studi Pendidikan Teknologi Informasi. Berdasarkan hasil observasi dan wawancara pada Ketua Program Studi Pendidikan Teknologi Informasi sendiri proses pengajuan judul tugas akhir masih belum efektif dan efisien dalam hal pelayanan kampus untuk mahasiswa tingkat akhir karena masih menggunakan beberapa *platform*. Proses pelaksanaannya sendiri sementara ini masih belum efisien, dimulai dengan menggunakan media *google form* sebagai *form* pengajuan judul dan untuk pencatatan judul yang telah di *ACC* masih menggunakan *microsoft word* maupun *excel* sehingga terkadang pengelolaan data menjadi kurang efektif dan efisien dikarenakan tidak ada suatu sistem informasi terpadu untuk pengajuan judul pada tugas akhir sehingga pada pencatatannya masih terpisah-pisah dan tidak ada interaksi antara mahasiswa dengan program studi yang memberikan kepastian informasi dalam pengajuan judul. Oleh sebab itu perlu dirancang sebuah aplikasi berbasis *website* dengan menerapkan teknologi informasi untuk mengefektifkan proses pengajuan judul tugas akhir pada Program Studi Pendidikan Teknologi Informasi. Saat ini *web* juga merupakan salah satu teknologi informasi yang banyak digunakan sebagai alat untuk mendapatkan informasi, *web* dibuat dengan tujuan agar pemakai dapat berinteraksi dengan penyedia informasi dengan mudah dan cepat. Menurut (Bekti, 2015) *Website* merupakan suatu kumpulan halaman yang mengandung data untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, *website* bisa bersifat statis maupun dinamis yang bisa dihubungkan dengan jaringan-jaringan halaman.

Sistem Informasi yang berbasis *website* merupakan langkah yang tepat untuk memberikan informasi seputar pengajuan judul tugas akhir yang dapat diakses dengan efisien oleh staf, kaprodi, dosen serta mahasiswa yang mengambil tugas akhir. *Website* merupakan salah satu teknologi yang memanfaatkan internet untuk memberikan informasi serta pertukaran data. Teknologi *website* memungkinkan kita dapat menghubungkan berbagai jenis *software* yang memiliki *platform*, basis data dan sistem operasi yang berbeda untuk saling bertukar data (Murtadho & Sanjaya, 2019).

Saat ini, tersedia berbagai macam jenis-jenis *framework* yang bisa digunakan untuk mengembangkan sistem informasi berbasis *web* diantaranya *framework* yang berbasis *PHP*, berbasis *javascript*, berbasis *CSS*. *Framework* adalah *platform* yang menyediakan kerangka kerja untuk membuat sebuah aplikasi dan *website*. Dengan menggunakan *framework* memungkinkan untuk membuat *web* dengan mudah dan cepat. Keuntungan menggunakan *framework* seperti yang dijelaskan oleh (Sidik, 2012) adalah penggunaan *library* yang *ready-to-use* sehingga memudahkan pengembang dalam pembuatan aplikasi yang tak perlu menulis *script* berulang-ulang. Penggunaan *codeIgniter* adalah salah satu *framework PHP* yang berbasis *website* yang mudah digunakan dalam pengembang sistem informasi berbasis *website*. Teknologi *web* berbasis *framework* sangatlah cepat, terlebih perkembangan *framework* dengan konsep *MVC* (*Model, View, Controler*) seperti *codeIgniter* dimana cara melakukan pemrograman menjadi lebih terstruktur yaitu *model* sebagai proses yang berinteraksi dengan databases, *view* sebagai penerima dan merepresentasikan data kepada user dan *controller* berfungsi untuk menerima *request* dan data dari *user* kemudian menentukan apa yang akan diproses oleh aplikasi menurut (Badiyanto & Murya, 2018).

Berdasarkan uraian latar belakang diatas, maka penulis tertarik untuk melakukan penelitian dengan judul: “Perancangan Sistem Informasi Pengajuan Judul Tugas Akhir Menggunakan *CodeIgniter* berbasis *website* Pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari”.

1. **Rumusan Masalah**

Berdasarkan latar belakang di atas, maka rumusan masalah penelitian ini adalah Bagaimana merancang sistem informasi pengajuan judul tugas akhir menggunakan *CodeIgniter* berbasis *website* pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari?

1. **Batasan Masalah**

Adapun batasan masalah yang terdapat dalam penelitian ini adalah sebagai berikut:

1. *Website* yang dirancang adalah *website* pengajuan judul tugas akhir yang ada di Program Studi Pendidikan Teknologi Informasi.
2. Dalam perancangannya *websit*e ini menggunakan *framework* *codeIgnite*r versi 3 dan *Bootstrap*. Dalam penelitian perancangan *website* ini menggunakan model *waterfall*.
3. Pada perancangan *website* ini menggunakan 3 diagram yang terdapat dalam *Unified Modeling Language* (UML) yaitu *use case* diagram, *activity* diagram, *sequence* diagram.
4. Hanya membahas proses yang terdapat pada sistem pengajuan judul tugas akhir mahasiswa di Program Studi Pendidikan Teknologi Informasi, yang berupa pengajuan judul tugas akhir, rekap judul yang masuk, judul diproses, diterima dan ditolak serta mahasiswa mendapatkan *notifikasi* judul yang di *ACC*.
5. Hanya membahas tentang pengajuan judul, pembagian dosen pembimbing dan rekap dosen pembimbing.
6. **Tujuan**

Menghasilkan perangkat lunak untuk sistem informasi pengajuan judul tugas akhir menggunakan *codeIgniter* berbasis *web* sehingga dapat mempermudah proses pengajuan judul tugas akhir pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari.

1. **Manfaat**

Manfaat yang diharapkan dapat diperoleh dari penelitian sistem ini adalah:

1. Manfaat teoritis yaitu sebagai sumbangan referensi bagi kepentingan ilmu pengetahuan, terutama bagi prosedur tindakan dalam perancangan sistem informasi pengajuan judul tugas akhir menggunakan *CodeIgnite*r berbasis *website*, dan dapat menjadi rujukan untuk penelitian berikutnya yang relevan dengan penelitian ini.
2. Manfaat Praktis yaitu dapat mempermudah staff, kaprodi dalam rekapitulasi judul tugas akhir yang diterima maupun disetujui di Program Studi Pendidikan Teknologi Informasi, serta untuk memudahkan mahasiswa dalam proses pengajuan judul tugas akhir di Program Studi Pendidikan Teknologi Informasi.

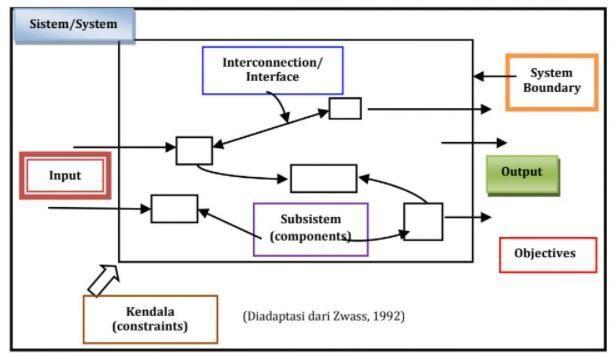
**BAB II**

**LANDASAN TEORI**

1. **Sistem Informasi**

Menurut (Punjul, 2016) Sistem merupakan suatu kumpulan dari komponen-komponen yang membentuk satu kesatuan. Sebuah organisasi dan sistem informasi untuk mencapai tujuan tertentu. Seorang manajer yang mengorganisasikan pekerjaan unitnya adalah orang yang menciptakan suatu sistem pengembangan *personal computer* (PC) menciptakan suatu sistem yang terdiri dari perangkat keras dan lunak PC ini menjadi subsistem dari suatu perangkat konferensi elektronik. Sistem fisik dan sosial adalah sistem yang abstrak (abstrak sistem) dari konsep dan ide. Contoh sistem yang abstrak adalah pengembangan sistem perangkat lunak.

Suatu sistem beroperasi di dalam lingkungan tertentu yang dibatasi oleh batasan-batasan tertentu, perhatikan gambar 2.1: sistem dan lingkungannya. Menunjukan bahwa sistem secara progresif menentukan subsistemnya dan kemudian memecahnya menjadi komponen-komponen yang lebih kecil, keterkaitan (*interconnections*) antar komponen tersebut juga akan terlihat jelas. Pertama dipastikan komponen atau subsistem apa yang akan tercakup dalam sistem tersebut, kemudian secara tidak langsung batasan sistem akan ditemukan pula. Kemudian, perlu dipikirkan secara berulang-ulang mengenai tujuan atau sasaran yang ingin dicapai dengan sistem tersebut, seperti tipe transaksi yang akan didukung, setelah itu, kendala (*constraints*) yang akan dihadapi dari Penggunaan sistem terpilih, seperti peraturan pemerintah, waktu pemrosesan, akan muncul keputusan apa, masukan (*input*) dan keluar (*output*) dari sistem tersebut.



**Gambar 2.1. Sistem Dan Lingkungannya (Punjul, 2016)**

Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis. Ada yang membuat perbedaan yang jelas antara sistem informasi, dan komputer sistem TIK, dan proses bisnis. Sistem informasi yang berbeda dari teknologi informasi dalam sistem informasi biasanya terlihat seperti memiliki komponen TIK. Hal ini terutama berkaitan dengan tujuan pemanfaatan teknologi informasi. Sistem informasi juga berbeda dari proses bisnis. Sistem informasi membantu untuk mengontrol kinerja proses bisnis. Menurut (Putu, 2011) berpendapat untuk sistem informasi sebagai tipe khusus dari sistem kerja. Sistem kerja adalah suatu sistem di mana manusia dan/atau mesin melakukan pekerjaan dengan menggunakan sumber daya untuk memproduksi produk tertentu dan/atau jasa bagi pelanggan. Sistem informasi adalah suatu sistem kerja yang kegiatannya ditujukan untuk pengolahan (menangkap, transmisi, menyimpan, mengambil, memanipulasi dan menampilkan) informasi.

Sistem informasi adalah suatu bentuk komunikasi sistem di mana data yang mewakili dan diproses sebagai bentuk dari memori sosial. Sistem informasi juga dapat dianggap sebagai bahasa semi formal yang mendukung manusia dalam pengambilan keputusan dan tindakan. Sistem informasi merupakan fokus utama dari studi untuk disiplin sistem informasi dan organisasi informatika. Sistem informasi adalah gabungan yang terorganisasi dari manusia, perangkat lunak, perangkat keras, jaringan komunikasi dan sumber data dalam mengumpulkan, mengubah, dan menyebarkan informasi dalam organisasi. Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Sistem informasi adalah kumpulan dari sub-sub sistem baik fisik maupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna (Putu, 2011).

Dari definisi di atas, penulis menyimpulkan bahwaSistem informasi adalah sebuah kumpulan atau kombinasi dari komponen-komponen yang didalamnya terdapat sub-sub sistem yang saling terhubung satu sama lain untuk mencapai tujuan yang sama yaitu mengolah data menjadi sebuah informasi.

1. ***Website***

*Website* adalah salah satu media publikasi elektronik yang terdiri dari halaman–halaman *web* (*web page*) yang terhubung satu dengan yang lain menggunakan link yang diletakan pada suatu teks atau *image*. *Website* dibuat pertama kali oleh Tim Berners Lee pada pada tahun 1990. (Bekti, 2015) mengemukakan bahwa: “*Website* adalah kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman”. *Website* dibagun dengan menggunakan bahasa *HTML*, CSS, *JavaScript*, *PHP* dan memanfaatkan protokol komunikasi *HTTP* yang terletak pada *application layer* pada referensi *layer OSI*. *Website* dibagi menjadi 2 yaitu *website* statis dan *website* dinamis.

1. ***Hypertext Markup Language (HTML)***

*HTML* adalah standar bahasa yang digunakan untuk mengatur penampilan dari halaman *web*. *HTML* tersusun atas pasangan-pasangan tag yang mempunyai fungsi yang berbeda-beda. Ciri dari tag pada *HTML* adalah diawali dengan tanda kurang dari “<” dan diakhiri oleh tanda lebih dari “>” seperti “<head>”. Pada *HTML* ada 2 macam tag yaitu tag pembuka “<…>” dan tag penutup “</…>”. Sebuah halaman *HTML* dibagi menjadi 2 bagian yaitu *header dan content* yang masing-masing diwakili oleh pasangan tag “<head>…</head>” dan “<body>…</body>”. *Header* dalam *HTML* berfungsi menyampaikan judul yang diwakili oleh tag “<title>…</title>”. *Content* pada *HTML* berisi script yang mengatur tampilan suatu halaman *HTML*.

Adapun, hubungan *HTML* dengan *PHP* yaitu *HTML* adalah halaman *web* disusun dari kode-kode *HTML* yang disimpan dalam sebuah file berekstensi *HTML* yang berada di server. File *HTML* ini dikirimkan oleh server ke *browser* pengguna, kemudian *browser* menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan pemrograman *PHP*, pemrograman ini harus diterjemahkan atau diolah oleh *web* server sehingga menghasilkan kode *HTML* yang dikirim ke *browser* agar dapat ditampilkan. Pemrograman *PHP* dapat berdiri sendiri ataupun disisipkan di antara kode *HTML* sehingga dapat ditampilkan bersama dengan kode-kode *HTML* tersebut dengan syarat *web* server harus *support* dengan *PHP*”. *HTML* (*Hypertext Markup Language)* merupakan bahasa yang digunakan untuk mendeskripsikan struktur sebuah halaman *web*. *HTML* berfungsi untuk mempublish dokumen *online.* Statement dasar dari *HTML* disebut tags. Sebuah *tag* dinyatakan dalam sebuah kurung siku (<>). *Tags* yang ditujukan untuk sebuah dokumen atau bagian dari suatu dokumen haruslah dibuat berupa pasangan. Terdiri dari tag pembuka dan tag penutup. Menurut (Sibero, 2013) “*HyperText Markup Language* atau *HTML* adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*”.

1. ***Cascading Style Sheet (CSS)***

Menurut (Abdulloh, 2015) *CSS* singkatan dari *cascading style sheets,* yaitu skrip yang digunakan untuk mengatur desain *website*. Walaupun *HTML* mempunyai kemampuan untuk mengatur tampilan *website*, namun kemampuannya sangat terbatas. Fungsi *CSS* adalah memberikan pengaturan yang lebih lengkap agar struktur *website* yang dibuat dengan *HTML* terlihat lebih rapi dan indah. *CSS* atau *Cascading Style Sheets* merupakan skrip yang digunakan dalam perancangan tampilan *web*. Skrip ini berisi fungsi untuk mengatur elemen-elemen pada tampilan *web*. Seperti *font, color, margin, background, spacing,* dan lainya (Budiyanto, 2013). *CSS* digunakan untuk mengontrol *presentation layer* pada *web*. *Presentation layer* merupakan sebuah *layer* yang mengatur tampilan dari konten web. Salah satu kelebihan *CSS* adalah satu file *CSS* dapat menyimpan *presentation style* untuk mengontrol ribuan halaman *web.*

1. ***JavaScript***

*JavaScript* adalah bahasa pemrograman *web* yang bersifat *client side* Programming *language*. *Client side* *Programming* *language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client.* Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome dan Mozilla Firefox. Bahasa pemrograman *client side* berbeda dengan bahasa pemrograman *server side* seperti *PHP*, dimana untuk *server side* seluruh kode program dijalankan di sisi server. Untuk menjalankan *javaScript*, kita hanya membutuhkan aplikasi *text editor* dan *web browser*. *JavaScript* memiliki fitur: *high-level* programming *language, client-side, loosely typed* dan berorientasi objek (Agung & Leo, 2012).

*Javascript* merupakan bahasa pemrograman berbasis *web* dan berorientasi objek atau sering juga disebut *OOP* (*Object Oriented Programming*). Dimana dianggap sebuah objek memiliki metode, properti dan *event* yang berbeda. Contohnya ketika kita mengklik tombol maka akan muncul sebuah pesan peringatan. Ketika kursor melintasi link muncul pesan. Itulah beberapa contoh *OOP* sederhana. Sebenarnya isi dari *language* tidak hanya *javascript* tetapi anda juga dapat menggunakan *Vbscript*. Yaitu bahasa pemrograman berbasis *visual basic script*. Kita tidak membahas *vbscript* karena tidak kompatibel dengan *browser* selain Internet Explorer. Perlu juga anda ketahui sedikit pengetahuan tentang *javascript* akan sangat membantu anda dalam memahami bahasa pemrograman lainnya seperti *PHP* karena *syntax*nya hampir mirip. *Javascript* biasanya gunakan untuk *event-event* tertentu.

1. ***PHP: Hypertext Preprocessor (PHP)***

Menurut (Effendi, 2017) Jika diartikan *PHP* memiliki beberapa pandangan dalam mengartikannya, akan tetapi kurang lebih *PHP* dapat kita ambil arti sebagai *PHP*. Ini merupakan bahasa yang hanya dapat berjalan pada server dan hasilnya dapat ditampilkan pada *client. PHP* adalah produk *open source* yang dapat digunakan secara gratis tanpa harus membayar untuk menggunakanya. *Interpreter PHP* dalam mengeksekusi kode *PHP* pada sisi server (*server side*), sedangkan tanpa adanya *interpreter PHP*, maka semua skrip dan aplikasi *PHP* yang dibuat tidak dapat dijalankan, (Nugroho, 2008).

Dari definisi di atas, penulis menyimpulkan bahwa *website* merupakan sebuah perangkat lunak yang berfungsi untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang terkoneksi dengan internet sehingga pengguna dapat mengakses *website* dimanapun dan kapanpun.

1. ***Framework***

Menurut (Upton, 2007) *Framework* adalah sekumpulan kode yang disimpan dalam file berbeda dan dapat memudahkan proses pengkodingan operasi yang berulang-ulang. *Framework* adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class dan function* yang dapat membantu *developer*/*programmer* dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke *database*, pemanggilan variabel, dan file. Sehingga *developer* lebih fokus dan lebih cepat membangun aplikasi.

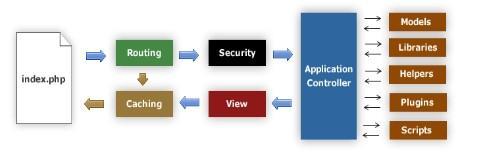
*Framework* adalah sebutan untuk sebuah kerangka kerja yang digunakan para *developer* aplikasi atau *software* untuk mempermudah mereka dalam membuat maupun mengembangkan sebuah *software* atau aplikasi. *Framework* sendiri berisikan fungsi dasar dan perintah yang lazim dipakai untuk membuat dan mengembangkan sebuah *software* atau aplikasi, dengan harapan aplikasi yang dibuat bias dibangun secara lebih terstruktur, lebih cepat serta lebih tersusun dengan cukup rapi.

1. ***Framework CodeIgniter***

Menurut (Hakim, 2010). *CodeIgniter* adalah sebuah *framework* untuk *web* yang dibuat dalam format *PHP*. Format yang dibuat ini selanjutnya dapat digunakan untuk membuat sistem aplikasi *web* yang kompleks. *CodeIgniter* dapat mempercepat proses pembuatan *web* karena semua kelas dan modul yang dibutuhkan sudah ada dan programmer hanya tinggal menggunakannya kembali pada aplikasi *web* yang akan dibuat.

Menurut (L. Aufan, 2010). *CodeIgniter* merupakan sebuah *framework* yang dibuat dengan menggunakan bahasa *PHP* yang dapat digunakan untuk pengembangan *web* secara cepat. *Framework* sendiri dapat diartikan sebagai suatu struktur pustaka-pustaka, kelas-kelas dan infrastruktur *run-time* yang dapat digunakan oleh programmer untuk mengembangkan aplikasi *web* secara cepat. Definisi lain, *codeIgniter* merupakan *framework* *PHP* yang dibuat berdasarkan *model view* *Controller (MVC)*. *CI* memiliki *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh aplikasi berbasis *web* misalnya mengakses *database,* memvalidasi *form* sehingga sistem yang dikembangkan mudah. Menurut (Hakim, 2010) *CodeIgniter* adalah sebuah *framework PHP* yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis *PHP* dibanding jika menulis semua kode program dari awal.

*CodeIgniter* pertama kali dibuat oleh Rick Ellis, CEO Ellislab, *Inc.(*[*http://ellislab.com*](http://ellislab.com)*).* Sebuah perusahaan yang memproduksi *CMS (Content Management System)* yang cukup handal, yaitu *Expression Engine (*[*http://www.expressionengine.com*](http://www.expressionengine.com)). Saat ini, *codeIgniter* dikembangkan dan dimainkan oleh *Expression Engine Development Team.* Adapun beberapa keuntungan menggunakan *codeIgniter,* diantaranya:

1. Gratis *codeIgniter* berlisensi di bawah *Apache/BSD open source*.
2. Ditulis menggunakan *PHP* 4 meskipun *codeIgniter* dapat berjalan di *PHP* 5, namun sampai saat ini kode program *codeIgniter* masih dibuat dengan menggunakan *PHP* 4.
3. Berukuran kecil ukuran *codeIgniter* yang kecil merupakan keunggulan tersendiri. Dibanding dengan *framework* lain yang berukuran besar.
4. Menggunakan konsep *MVC* *codeIgniter* menggunakan konsep *MVC* yang memungkinkan pemisahan layer *application-logic* dan *presentation.*
5. *URL* yang sederhana secara default, *URL* yang dihasilkan *codeIgniter* sangat bersih dan *Search Engine Friendly (SEF).*
6. Memiliki paket *library* yang lengkap *codeIgnite*r mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis web, misalnya mengakses database, mengirim email, memvalidasi *form*, menangani *session* dan sebagainya.
7. *Extensible* sistem dapat dikembangkan dengan mudah menggunakan *plugin* dan *helper,* atau dengan menggunakan *hooks.*
8. Tidak Memerlukan *template engine* meskipun *codeIgniter* dilengkapi dengan *template parser* sederhana yang dapat digunakan, tetapi hal ini tidak mengharuskan kita untuk menggunakannya.
9. Dokumentasi lengkap dan jelas dari sekian banyak *framework, codeIgniter* adalah satu-satunya *framework* dengan dokumentasi yang lengkap dan jelas.
10. Komunitas Komunitas *codeIgniter* saat ini berkembang pesat. Salah satu komunitasnya bisa dilihat di ([*http://CodeIgniter.com/forum*/](http://codeigniter.com/forum/)).

**Gambar 2.2. *Application Flowchart***

**Sumber: (Hakim, 2010)**

Keterangan:

1. *Index. PHP* berfungsi sebagai *front controller*, menginisialisasi *base resource* untuk menjalankan *codeIgniter.*
2. *Router* memeriksa *HTTP* *request* untuk menentukan apa yang harus dilakukan dengannya.
3. Jika *cache* aktif, maka hasilnya akan langsung dikirimkan ke *browser* dengan mengabaikan aliran data normal.
4. *Security.* Sebelum *controller* dimuat, *HTTP* *request* dan data yang dikirimkan user akan di *filter* untuk keamanan.
5. *Controller* memuat model, *core libraries, plugins, helpers* dan semua *resource* yang diperlukan untuk memproses *request.*
6. Akhirnya *view* yang dihasilkan akan dikirimkan ke *browser.* Jika *cache* aktif, maka *view* akan disimpan sebagai *cache* dahulu, sehingga pada *request* berikutnya langsung dapat ditampilkan.

*CodeIgniter* merupakan *framework PHP* yaitu yang bersifat *open source* sehingga dapat digunakan oleh siapapun dan mengadopsi pola desain *MVC* (*Model, View, Controller*) yang memungkinkan pengembangan *website* secara dinamis dan memungkinkan penemuan serta penanganan *error* yang lebih cepat. Penggunaan *framework codeIgniter* tidak hanya membantu memudahkan proses pengkodingan tetapi juga membuat website yang akan dibangun menjadi lebih aman karena memiliki tingkat sekuritas yang lebih tinggi dibanding *framework PHP* lain (Solanki et al, 2017).

1. ***Framework Bootstrap***

*Bootstrap* merupakan sebuah *framework* *CSS* yang memudahkan pengembang untuk membangun *website* yang menarik dan *responsif*. Tidak konsistensinya terhadap aplikasi individual membuat sulitnya untuk mengembangkan dan pemeliharaannya. *Bootstrap* adalah *CSS* tetapi dibentuk dengan *LESS*, sebuah *pre-prosessor* yang memberi fleksibilitas dari *CSS* biasa. *Bootstrap* memberikan solusi rapi dan seragam terhadap solusi yang umum, tugas *interface* yang setiap pengembang hadapi, sehingga dengan bantuan *Bootstrap*, kita bisa membuat *responsive website* dengan cepat dan mudah dan dapat berjalan sempurna pada *browser-browser* populer seperti *Chrome, Firefox, Safari, Opera* dan Internet *Explorer* (Alatas, 2013). Keunggulan dalam menggunakan *bootstrap* adalah semua bagian untuk antarmuka pengguna menggunakan *style* *CSS*, *bootstrap* dapat menggunakan *LESS* *preprocessor* sebuah teknologi yang mengurangi dan mengefisienkan penulisan kode *CSS*. *Bootstrap* dapat diintegrasikan dengan *javaScript* untuk menjadikan lebih menarik dengan efek-efek yang dapat diberikan dengan *javaScript.*

Menurut(Spurlock, 2013) menyatakan bahwa *bootstrap* adalah sebuah *framework* untuk *CSS* dan berupa produk *open source* yang dibuat oleh Mark Otto dan Jacob Thornton. Pada awalnya *bootstrap* ini dibuat untuk membuat standarisasi *front end* untuk semua programmer di perusahaannya. *Bootstrap* telah berubah dari yang sebelumnya adalah *CSS -Driven* proyek ke sebuah *host* dari *javaScript plugins* dan ikon yang dapat dengan mudah digunakan untuk formulir dan tombol. *Bootstrap* adalah paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, *bootstrap* adalah template desain *web* dengan fitur *plus*. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai *HTML* dan *CSS*, anda pun siap menggunakan *bootstrap.*

Berdasarkan definisi di atas, penulis menyimpulkan bahwa *framework* merupakan kumpulan *class* atau object yang masing-masing memiliki fungsi tersendiri dalam membangun sebuah kerangka kerja. Sebuah *framework codeIgniter* terdiri dari Model mencakup semua proses yang terkait dengan pemanggilan struktur data baik berupa pemanggilan fungsi, *input processing* atau mencetak *output ke* dalam *browse*r. *View* mencakup semua proses yang terkait *layout output*. Bisa dibilang untuk menaruh template *interface* website atau aplikasi. *Controller* mencakup semua proses yang terkait dengan pemanggilan database dan kapsulisasi proses proses utama. Jadi semisal dibagian ini ada file bernama member *PHP* maka semua proses yang terkait dengan memberikan di kapsulisasi/dikelompokan dalam file ini.

1. ***My Structured Query Language (MySQL)***

Menurut (Kadir, 2014) *SQL* (*Structure Query Language*) adalah bahasa yang digunakan untuk mengakses basis data yang tergolong relational. Dalam sistem database tak relasional, semua informasi disimpan pada satu bidang luas, yang kadangkala data di dalamnya sangat sulit dan melelahkan untuk diakses. Tetapi *MySQL* merupakan sebuah sistem database relasional, sehingga dapat mengelompokkan informasi ke dalam tabel-tabel atau grup-grup informasi yang berkaitan. Sesungguhnya *SQL* tidak terbatas hanya untuk mengambil data(*query*), tetapi juga dapat dipakai untuk menciptakan tabel(*create*), menghapus(delete), menambah data ke tabel(*insert*), menghapus data di tabel(*drop*), mengganti data di tabel(*update*), dan berbagai operasi lain. Setiap tabel memuat bidang-bidang yang terpisah, yang mempresentasikan setiap bit informasi. *MySQL* menggunakan indeks untuk mempercepat proses pencarian terhadap baris informasi tertentu. *MySQL* memerlukan sedikitnya satu indeks pada tiap tabel. Biasanya akan menggunakan suatu *primary key* atau pengenal unik untuk membantu penjejakan data.

Berdasarkan definisi di atas, penulis menyimpulkan bahwa *MySQL* adalah salah satu aplikasi DBMS (Database Management System) yang sudah sangat banyak digunakan oleh para pemrograman aplikasi web. *MySQL* adalah sebuah sistem database relasional dan merupakan *databases* server sehingga dapat di akses oleh multi user serta mengelompokkan informasi ke dalam tabel-tabel atau grup-grup informasi yang berkaitan.

1. ***Unified Modeling Language (UML)***

*Unified Modeling Language* (*UML*) Merupakan teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem (Mulyani, 2017). Dengan terdapatnya *UML*, model buat fitur perangkat lunak yang bisa di jalankan ke fitur perangkat keras bisa terbuat. *UML* sesuai digunakan buat bahasa pemrograman *C++, C#,* *java serta PHP* sebab memakai *Class* serta *operation.*

*UML* juga sediakan standar untuk notasi dan diagram yang digunakan untuk memodelkan sebuah sistem. *UML* dikembangin oleh 3 pendekar berorientasi objek, yaitu Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. *UML* menjadi Bahasa yang telah dipakai untuk komunikasi dalam perspektif objek antar pengguna dengan *developer*, antara *developer* dengan *developer*, antara *developer* dengan *developer* desain, dan antara *developer* desain dengan *developer* pemrograman ((Bima, 2021).

1. ***Use Case* Diagram**

*Use Case* Diagram merupakan salah satu dari diagram *Unified Modelling Language* yang mendeskripsikan hubungan aktor dan sistem.

1. *Aktor*: Yaitu orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dibuat.
2. *Use Case*: Yaitu fungsionalitas yang disediakan sistem sebagai unit unit yang saling bertukar pesan antar unit atau *aktor*.

Simbol-simbol tersebut dapat digambar pada tabel 2.1 sebagai berikut:

**Tabel 2.1 Simbol *Use Case* Diagram (Hermawan, 2017)**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Gambar** | **Nama** | **Keterangan** |
| 1. |  | Actor | Menspesifikasikan himpunan kepada pemakai untuk memainkan pada saat interaksi dengan use case |
| 2. |  | Dependency | Hubungan disaat perubahan yang terjadi kepada suatu elemen mandiri dan akan menjadi masalah di saat elemen yang bergantung pada elemen yang tidak mandiri (independent) |
| 3. |  | Generalization | Hubungan yang dimana letak antara objek kecil atau disebut anak yang berbagi perilaku dan struktur data dari objek yang ada diatasnya objek induk |
| 4. |  | Include | Mem Spesifikasi bahwa use case diagram sumber secara eksplisit |
| 5. |  | Exclude | Mem Spesifikasi bahwa use case target memperlebar perilaku dari use case sumber kepada sebuah titik yang diberi. |
| 6. |  | Association | Apa yang menghubungkan antara objek satu Dengan objek lainnya. |

1. ***Class* Diagram**

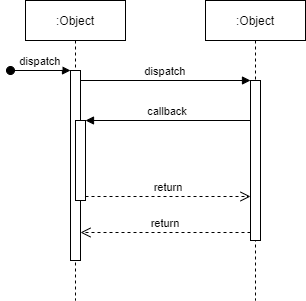
*Class* diagram digunakan untuk menggambarkan struktur dalam objek sistem. Diagram ini menunjukkan *class* object yang menyusun sistem dan juga hubungan antara *class* object (Sugiarti, 2013). Kelas memiliki atribut dan metode atau operasi. Atribut adalah variabel-variabel yang mendeskripsikan properti dengan bentuk sebaris teks dalam kelas tersebut, sedangkan metode adalah fungsi yang dimiliki oleh kelas yang dalam *class* diagram dilambangkan menggunakan simbol-simbol. (Sugiarti, 2013).

**Tabel 2.2 Simbol pada *class* diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **Simbol** | | | **Deskripsi** |
| *Package* | | | *Package* merupakan sebuah bungkusan dari satu atau lebih kelas. |
| Kelas | | | Kelas pada struktur sistem, setiap kelas memiliki nama, *attribute*, dan *operation* atau *method*. |
|  | nama kelas |  |
| +*Attribute1*  +*Attribute2* |
| +*Operation*1()  +*Operation*2() |
| Antarmuka / *interface* | | | Sama seperti konsep *interface* dalam pemrograman berorientasi objek. |
| Asosiasi | | | Relasi antar kelas dengan pengertian umum. |
| Asosiasi berarah | | | Relasi antar kelas dengan pengertian kelas yang satu digunakan oleh kelas yang lain. |
| Generalisasi | | | Relasi antar kelas dengan pengertian generalisasi-spesialisasi (umum-khusus). |
| Kebergantungan | | | Relasi antar kelas dengan pengertian kebergantungan antar kelas. |
| Agregasi | | | Relasi antar kelas dengan makna semua-sebagian  (*whole-part*). |

***3. Sequence* Diagram**

Diagram sekuen merupakan diagram yang digunakan untuk menggambarkan perilaku objek pada *use case* dengan mendeskripsikan proses objek dengan pesan yang dikirimkan. Oleh karena itu, untuk menggambar diagram sekuen harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu (Sugiarti, 2013).



**Gambar 2.5 Simbol *Sequence* Diagram**

1. **Penelitian Relevan**

Tujuan dari penelitian yang relevan yaitu untuk mencari perbandingan berupa kesamaan dan perbedaan dari penelitian sebelumnya dengan penelitian yang akan dilakukan agar kedepannya tidak terjadi kesamaan (plagiat). Adapun penelitian relevannya sebagai berikut:

1. Penelitian sebelumnya dilakukan oleh (Saro dkk, 2021) tentang sistem informasi pengajuan judul skripsi menggunakan bahasa pemrograman *PHP* dan database *MySQL* (Studi kasus teknik universitas ibnu sina) yang bertujuan untuk membangun sistem informasi pengajuan judul tugas akhir yang diharapkan dapat mendukung dan memudahkan proses pengajuan judul tugas akhir dan dengan mudah mendapatkan informasi tentang pelayanan kampus terutama saat pengajuan tugas akhir di Universitas Ibnu Sina. Sistem pengajuan judul skripsi ini dirancang sebagai upaya meningkatkan efisiensi dan pengefektifan dalam pengelolaan data tugas akhir mahasiswa fakultas teknik program studi teknik informatika. Sistem ini juga dapat memberikan banyak kemudahan dalam melakukan pengajuan judul khususnya bagi mahasiswa fakultas teknik program studi pendidikan teknologi informasi.
2. Penelitian ini juga dilakukan oleh (Zulkifli, 2021) tentang sistem informasi pengajuan judul skripsi secara *online* untuk mencegah penyebaran virus Covid-19 di lingkungan Universitas Almuslim. Selama masa pandemi seluruh instansi pendidikan di Indonesia menerapkan kebijakan pembelajaran daring untuk mengantisipasi penularan virus Covid-19. Oleh karena itu dalam penelitian ini dilakukan pengembangan suatu *platform* aplikasi yang mampu mendukung pelaksanaan kegiatan administrasi dalam kasus ini adalah pengajuan judul skripsi mahasiswa yang dilakukan secara daring. Metode pengembangan sistem yang digunakan adalah metode pengembangan *prototype*. Adapun hasil yang didapatkan yaitu membantu menyelesaikan permasalahan yang dihadapi dalam proses pengajuan judul skripsi di Fakultas Ilmu Komputer saat ini dengan menerapkan teknologi informasi.
3. Penelitian lainnya yang membahas mengenai sistem informasi pengajuan judul menggunakan *codeigniter* berbasis *website* yang dilakukan oleh (Jamasnia dkk, 2021) tentang sistem informasi pengajuan judul skripsi berbasis *web service.* Tujuan dari penelitian ini yaitu merancang sistem informasi yang dapat memfasilitasi proses kebutuhan mahasiswa dalam mengajukan judul penelitian yang lebih efektif dan efisien. Sistem ini berfungsi sebagai fasilitas pengajuan judul mahasiswa ke ketua prodi dengan konsep komunikasi dua arah terkait tindak lanjut judul yang diajukan. Penelitian ini mengimplementasikan metode kualitatif dengan mengumpulkan data pada Fakultas Ilmu Komputer Universitas Al Asyariah Mandar yakni wawancara dan studi pustaka. Penelitian ini telah berhasil merancang sistem informasi pengajuan judul skripsi berbasis *web service* yang diharapkan mampu mengefisienkan dan mengefektifkan proses pengajuan judul mahasiswa dengan konsep komunikasi dua arah antara mahasiswa dan ketua prodi.
4. Penelitian lainnya yang membahas mengenai sistem informasi pengajuan judul menggunakan *framework codeIgniter* yang dilakukan oleh (Widaretna dkk, 2017) tentang pengembangan aplikasi *web* pengajuan judul skripsi dengan *framework codeIgniter* dan *HTML5* pada Sekolah Tinggi Teknologi Bandung. Aplikasi *web* pengajuan judul skripsi merupakan sebuah aplikasi *web* yang dibangun dengan tujuan untuk mengganti proses pengajuan judul skripsi yang saat ini digunakan di Sekolah Tinggi Teknologi Bandung. Aplikasi *web* ini memungkinkan mahasiswa untuk melakukan proses pengajuan judul skripsi secara *online*, dari mulai mengajukan judul hingga memperoleh informasi judul yang diajukan tersebut diterima atau ditolak. Selain itu, pihak lain yang terlibat dalam proses pengajuan judul skripsi ini juga dapat melakukan tugasnya masing-masing tanpa perlu bertatap muka secara langsung. Aplikasi *web* pengajuan judul skripsi ini dibangun dengan menggunakan metode *prototype* dan bahasa pemrograman *PHP* dengan menggunakan *framework CodeIgniter* dan *HTML5*. Antarmuka dari aplikasi *web* dirancang agar menjadi *web responsive,* sehingga menjadi aplikasi yang *user friendly* dan dapat memaksimalkan penyampaian informasi kepada pengguna meskipun diakses menggunakan *device* dengan ukuran yang berbeda. Hasil akhir dari penelitian ini berupa aplikasi *web* pengajuan judul skripsi yang dapat diakses secara *online* dengan menggunakan berbagai macam ukuran *device* yang berbeda oleh semua pengguna yang terlibat dalam proses pengajuan judul skripsi di Sekolah Tinggi Teknologi Bandung, karena dari hasil pengujian yang telah dilakukan dapat disimpulkan bahwa hampir seluruh tampilan dari aplikasi *web* ini bersifat *responsive* dan tidak memperlihatkan perbedaan yang terlalu besar.
5. Penelitian lainnya yang membahas mengenai sistem informasi pengajuan judul menggunakan *framework codeIgniter* berbasis *website* yang dilakukan oleh (Humairah, 2020) tentang pengembangan sistem pengajuan judul skripsi di Jurusan Pendidikan Teknik Elektro Universitas Negeri Makassar berbasis *website*. Penelitian ini adalah penelitian dan pengembangan (*R&D*) yang bertujuan untuk mengembangkan sistem informasi pengajuan judul skripsi di Jurusan Pendidikan Teknik Elektro Universitas Negeri Makassar berbasis *websit*e. Pada Jurusan Pendidikan Teknik Elektro FT UNM terdapat mekanisme yang rutin dilaksanakan oleh setiap mahasiswa di Universitas Negeri Makassar, yakni penyusunan skripsi. Oleh karena itu penting adanya untuk membuat sistem informasi skripsi yang dimulai dari sistem pengajuan judul di Jurusan Pendidikan Teknik Elektro karena sistem yang diterapkan masih berjalan secara manual baik dari pengajuan, pengetikan judul yang diajukan untuk dirapatkan maupun dari pengumuman judul yang telah dirapatkan yang tidak dapat dilihat hasilnya oleh mahasiswa secara langsung ketika rapat judul telah selesai dilakukan dan tidak adanya pemberitahuan kepada mahasiswa jika judul yang diajukan masuk dalam kategori penolakan. Berdasarkan kondisi tersebut dibutuhkan sebuah sistem informasi pengajuan judul di Jurusan Pendidikan Teknik Elektro. Adanya sistem yang berbasis digital maka segala pengurusan judul skripsi dapat dilakukan dengan mudah, dilihat dari sisi mahasiswa yang dapat dengan mudah memasukan judul yang diajukan, adanya format berkas untuk tanda tangan ketua prodi dan pembimbing akademik yang dapat didapatkan langsung pada sistem.

Berdasarkan kelima hasil penelitian tersebut, perbedaan dengan penelitian yang akan dilakukan dengan judul “Perancangan Sistem Informasi Pengajuan Judul Tugas Akhir Menggunakan *CodeIgniter* Berbasis *Website* di Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari”. Pada pembuatan sistem informasi ini akan menggunakan *framework codeIgniter* sehingga dapat memudahkan dalam perancangan maupun pengembangan sistem informasi tersebut. Pada perancangan sistem informasi pengajuan judul tugas akhir yang akan dirancang di Program Studi Pendidikan Teknologi Informasi yang berbasis *website* dan memiliki fitur yaitu pengajuan judul oleh mahasiswa serta *fitur* berkas persyaratan sebelum mengajukan judul skripsi seperti KRS, KHS, dan jurnal penelitian yang berkaitan dengan judul yang akan diajukan sehingga pihak prodi dapat melakukan tahap seleksi terkait dengan literasi judul skripsi yang akan diajukan. Pada website ini akan mempermudah user dalam mengakses website tersebut untuk mendapatkan informasi yang berhubungan dengan pengajuan judul di Program Studi Pendidikan Teknologi Informasi, dan pengguna dapat mengakses dimanapun dan kapanpun.

**BAB III**

**METODOLOGI PENELITIAN**

1. **Jenis Penelitian**

Penelitian ini adalah jenis penelitian perancangan dengan model *waterfall* yang terdiri dari lima langkah: 1. Analisis kebutuhan *(Analyzing)* 2. Desain Sistem *(Design System)* 3. Pembuatan kode program *(Coding)* 4. Pengujian sistem *(testing System)* 5. Implementasi *(implementation)*. Adapun tahapan-tahapan tersebut dapat dilihat pada gambar 3.1. Menurut penelitian (Dennis, 2021) Metode air terjun atau sering disebut metode *Waterfall* sering dinamakan siklus hidup klasik (*Classic Life Cycle*), nama model ini sebenarnya adalah “*Linear Sequential Model*” dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak.

1. **Prosedur Penelitian**

Untuk memudahkan peneliti dalam melakukan penelitian ini maka dibuatlah sebuah prosedur penelitian berdasarkan metode yang digunakan dapat digambarkan menu *flowchart* dibawah ini.

Analisis kebutuhan

Desain sistem

Penulisan kode Program

Pengujian

Implementasi

**Gambar 3.1 Metode *Waterfall* (Dennis, 2021)**

* + 1. **Analisis Kebutuhan**
       1. Identifikasi Masalah

Permasalahan yang terjadi di Program Studi Pendidikan Teknologi Informasi yaitu Pengajuan Judul tugas akhir yang diterapkan dalam Program Studi Pendidikan Teknologi Informasi tersebut masih bersifat manual sehingga kaprodi, dosen dan staf merasa kesulitan dalam hal rekap data judul yang diajukan dan kurang mendapatkan informasi secara cepat dan tepat, serta memperlambat proses kerja Program Studi dalam mengolah informasi.

* + - 1. Analisis Sistem

1. Analisis sistem berjalan

Menurut (Bentley & Whitten, 2009) analisa sistem adalah sebuah metode untuk mencari solusi dari permasalahan sistem yang ada dengan cara mengelompokkan komponen yang ada menjadi komponen-komponen yang lebih kecil agar solusi yang ditemukan sesuai dengan kebutuhan sistem.

Pada tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem: 1. *Identify*, yaitu mengidentifikasi masalah 2. *Understand*, yaitu memahami kerja dari sistem yang ada 3. *Analyze*, yaitu menganalisis sistem 4. *Report*, yaitu membuat laporan hasil analisis.

Sistem yang berjalan di Program Studi Pendidikan Teknologi Informasi saat ini masih bersifat manual, sehingga proses dari pengajuan judul, proses rekap judul yang diterima, ditolak dan sampai disetujui masih dilakukan secara manual menggunakan *google form*. Selain itu sistem informasi pengajuan judul tugas akhir ini masih dilakukan secara manual dengan cara menerima *notifikasi* judul yang disetujui dan ditolak melalui *Gmail*.

Analisis sistem yang sedang berjalan dilakukan dengan tujuan untuk mengetahui proses kerja yang sedang berjalan dan juga untuk mengevaluasi serta memberikan gambaran rencana pemecahan masalah yang dihadapi. Prosedur yang sedang berjalan adalah sebagai berikut:

1. Mahasiswa mengajukan judul melalui *google form*
2. Mahasiswa mengajukan judul, kemudian kaprodi akan mengecek judul yang telah dikirim, kemudian kaprodi akan mengadakan rapat bersama dosen Program Studi Pendidikan Teknologi Informasi terkait dengan judul yang disetujui dan ditolak.
3. Jika judul disetujui, maka kaprodi akan memberitahukan judul yang disetujui melalui email mahasiswa yang terdaftar.
4. KaProdi rekap judul yang disetujui secara manual menggunakan *microsoft excel*.

Evaluasi sistem dimaksudkan untuk menemukan kelemahan sistem yang akan diperbaiki. Berdasarkan uraian mengenai analisis sistem yang sedang berjalan di Program Studi Pendidikan Teknologi Informasi, maka dapat ditemukan beberapa kekurangan atau kelemahan yang diantaranya adalah:

**Tabel 3.1 Evaluasi system**

|  |  |  |
| --- | --- | --- |
| No | Masalah | Solusi |
| 1 | Pengajuan judul masih dilakukan secara manual. Yang dimana judul yang diajukan berbeda *platform* dengan pemberitahuan judul yang diterima oleh mahasiswa | menggunakan sistem informasi pengajuan judul berbasis *web* agar informasi pengajuan judul dapat dilakukan secara terpusat yang dimana mahasiswa mengajukan judul sampai pemberitahuan diterima atau ditolak menggunakan satu *platform*. |
| 2 | Pengolahan data pengajuan judul dilakukan secara manual menggunakan *microsoft excel* | Menggunakan sistem informasi pengajuan judul berbasis *web* yang telah menggunakan *database*. |
| 3 | Proses rekap dan penyimpanan data belum terorganisir dengan baik. | Menggunakan sistem informasi ini rekap data pengajuan judul tugas akhir akan tersimpan dengan baik. |
| 4 | Belum adanya sistem informasi yang baik yang digunakan untuk memberikan kemudahan dalam proses pengajuan judul tugas akhir di Program Studi Pendidikan Informasi. | Sistem informasi pengajuan judul tugas akhir berbasis *web* dapat menjadi sebuah sistem informasi yang lebih efektif. |

1. Analisis sistem ajuan

Berdasarkan analisa sistem yang diusulkan maka diketahui sistem yang lama masih bersifat manual sehingga tidak bisa memenuhi kebutuhan dalam pengolahan data secara cepat dan efektif. Setelah kebutuhan sistem diketahui maka langkah selanjutnya adalah merancang sistem informasi pengajuan judul tugas akhir yang bertujuan untuk memenuhi kebutuhan Program Studi Pendidikan Teknologi Informasi dalam melakukan pengajuan judul tugas akhir secara cepat dan efektif.

Dalam hal ini prosedur yang dibuat tidak mengalami banyak perubahan dari sistem yang sedang berjalan, hanya berbeda dalam penggunaan sistem komputerisasi dengan teknologi internet yang diharapkan dapat mengefektifkan pengajuan judul tugas akhir pada Program Studi Pendidikan Teknologi Informasi.

Berikut adalah prosedur sistem informasi pengajuan judul tugas akhir yang diusulkan pada Program Studi Pendidikan Teknologi Informasi.

1. Mahasiswa *login* ke sistem
2. Mahasiswa mengupload berkas persyaratan untuk pengajuan judul seperti upload kartu tanda mahasiswa, bukti pembayaran, dan transkrip nilai.
3. Mahasiswa mengupload seluruh berkas persyaratan dalam pengajuan judul tugas akhir maka prodi akan memberikan feedback apakah berkas tersebut diterima atau ditolak serta memberikan catatan.
4. Jika KaProdi setuju maka selanjutnya mahasiswa melakukan pengajuan judul tugas akhir
5. KaProdi melakukan persetujuan terkait dengan judul yang diajukan oleh mahasiswa serta memberikan catatan.
6. KaProdi melakukan pembagian dosen pembimbing.
7. KaProdi melakukan pembagian dosen pembimbing maka dosen dapat melihat keterangan sebagai dosen pembimbing satu atau dosen pembimbing dua serta dosen juga dapat melihat list mahasiswa bimbingan.
8. Kebutuhan Data

Data yang dibutuhkan dalam merancang sistem informasi pengajuan judul tugas akhir berbasis *website* di Program Studi Pendidikan Teknologi Informasi diantaranya ialah:

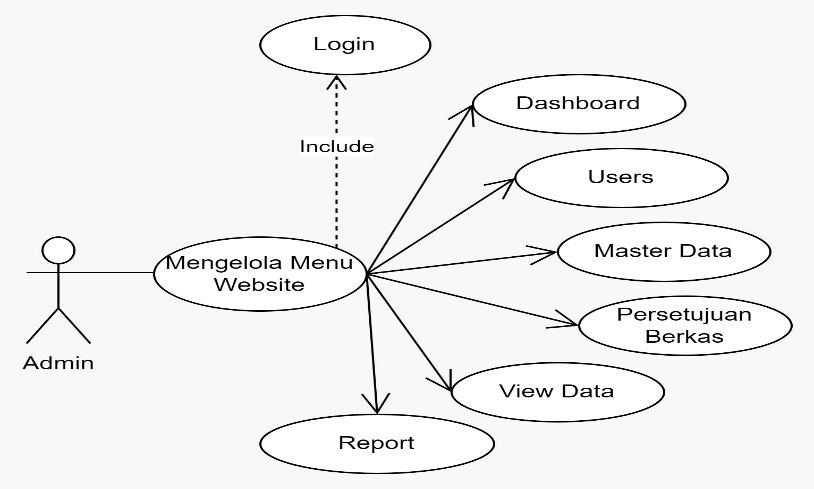
1. Data mahasiswa di Program Studi Pendidikan Teknologi Informasi.
2. Data dosen di Program Studi Pendidikan Teknologi Informasi.

**2. Desain Sistem**

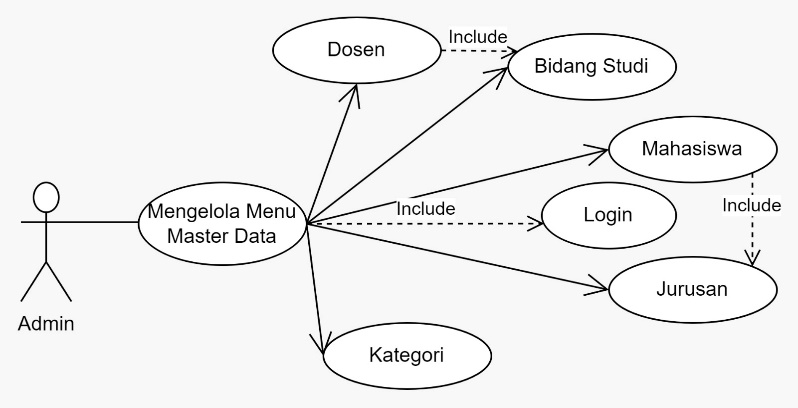
Pada proses desain ini sistem dilakukan perancangan sesuai spesifikasi kebutuhan sebelumnya, desain sistem membantu dalam menentukan perangkat keras (*hardware),* serta membantu dalam mendefinisikan arsitektur sistem secara keseluruhan. Perancangan suatu sistem membutuhkan adanya teknik-teknik penyusunan sistem untuk menganalisis dan mendokumentasikan data yang mengalir di dalam sistem tersebut. Desain sistem yang akan dikembangkan meliputi desain perancangan *Unified Modelling Language* (*UML*) antara lain: *use case diagram*, *activiti diagram*, dan *sequence diagram.*

Dalam perancangan *website*, peneliti membuat penggambaran fungsionalitas menggunakan *Unified Modelling Language* (*UML*), yaitu menggunakan *Use Case Diagram*, *Activity Diagram* dan *Sequence Diagram*.

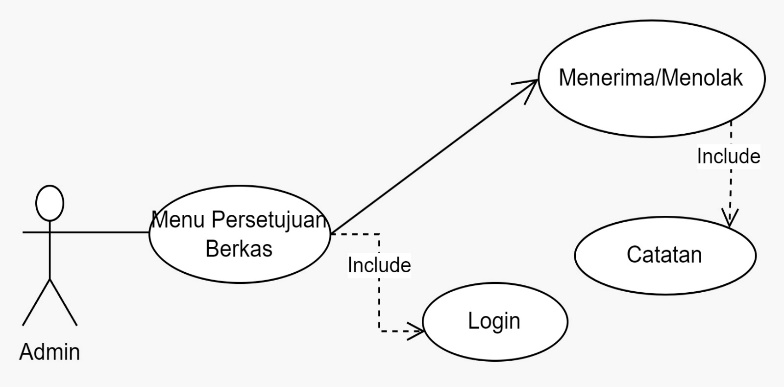
1. ***Use Case* Diagram**
2. *Use case* Admin



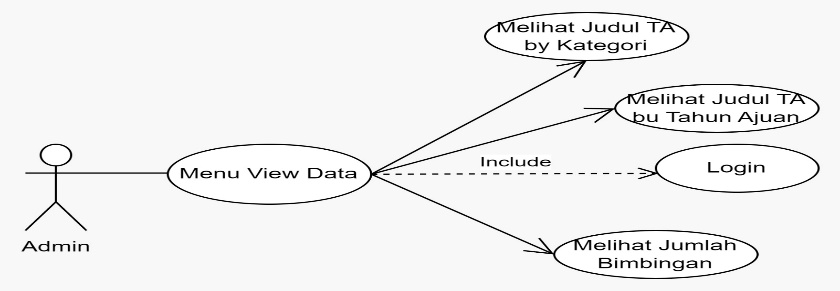
**Gambar 3.2 *Use case* admin mengelola *menu* *website* pengajuan judul skripsi**



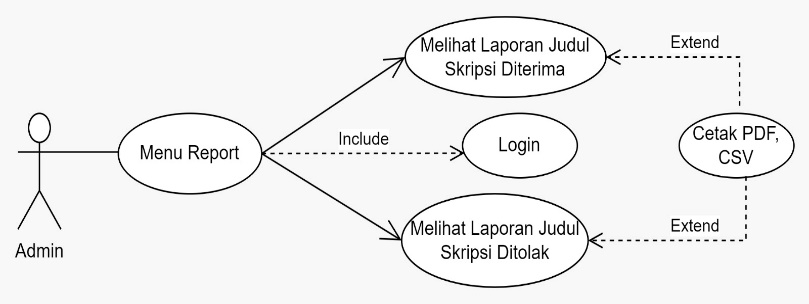
**Gambar 3.3 *Use case* admin mengelola menu *website* pengajuan judul skripsi**



**Gambar 3.4 *Use case* admin menu persetujuan berkas**

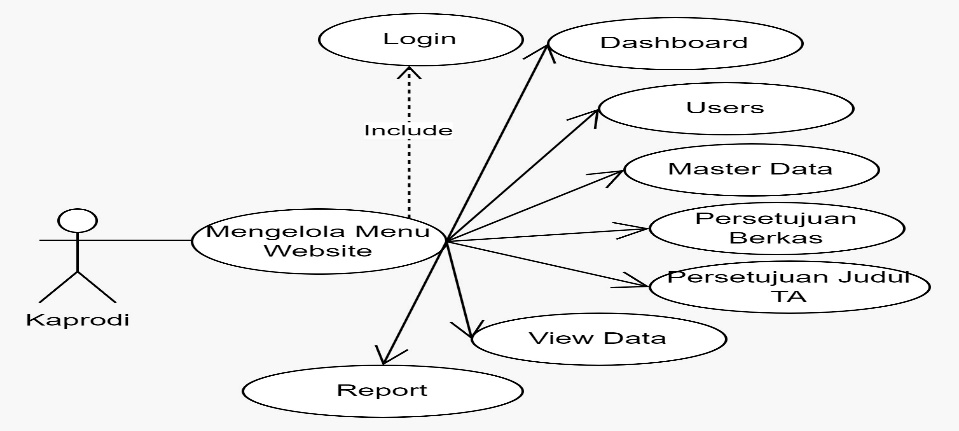


**Gambar 3.5 *Use case* *admin* *menu* *view* data**

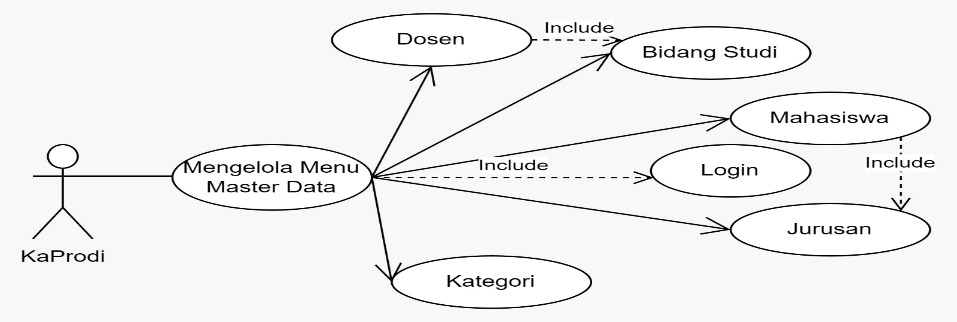


**Gambar 3.6 *Use case* *admin* mengelola *menu* *report***

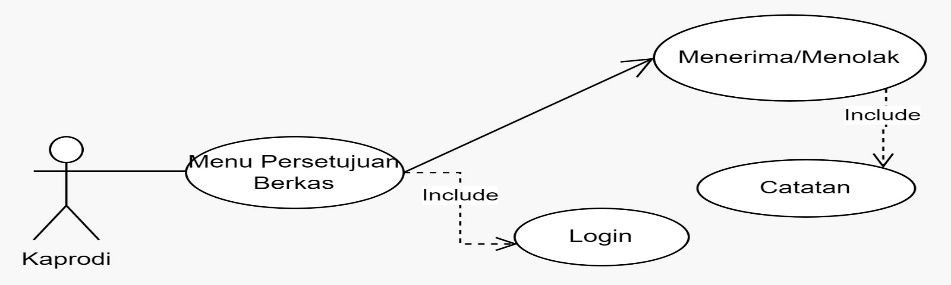
1. *Use Case* Kaprodi



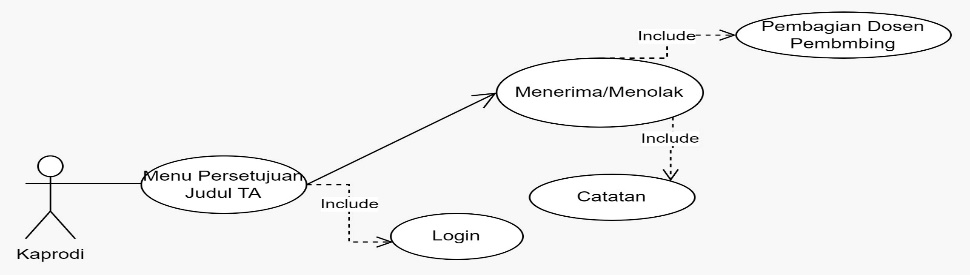
**Gambar 3.7 *Use case* kaprodi mengelola *menu* *website***



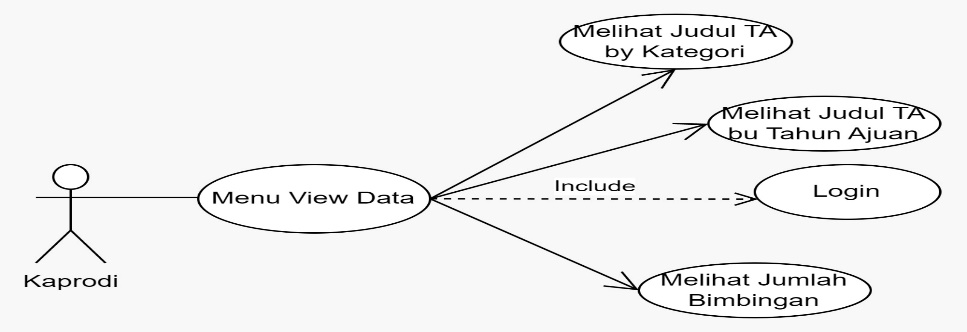
**Gambar 3.8 *Use case* kaprodi mengelola *menu* master data**



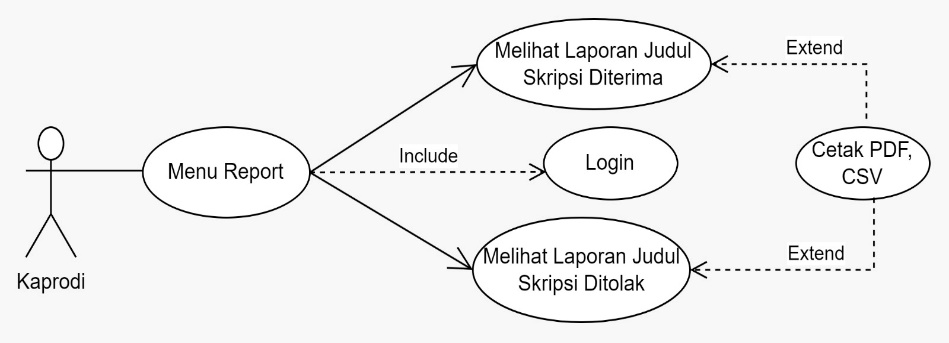
**Gambar 3.9 *Use case* kaprodi mengelola *menu* persetujuan berkas**



**Gambar 3.10 Use case kaprodi *menu* persetujuan judul skripsi**

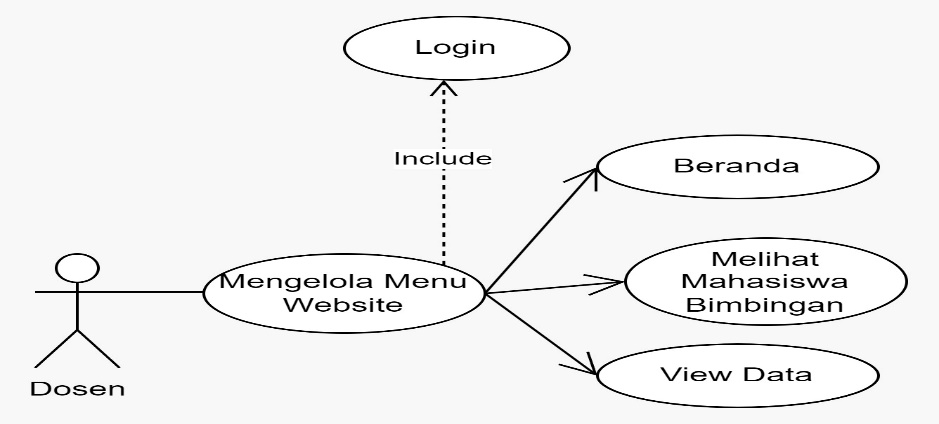


**Gambar 3.11 *Use case* kaprodi *menu* *view* data**

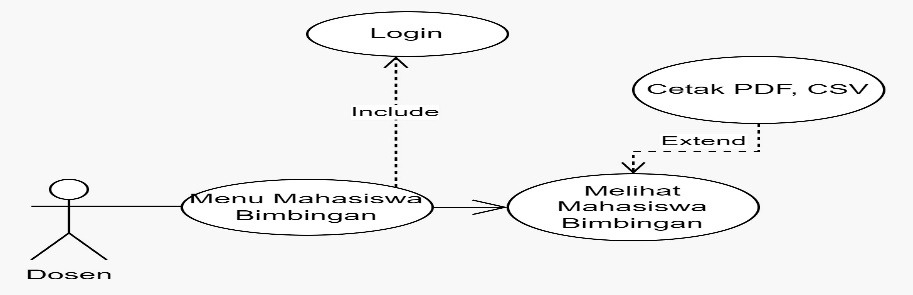


**Gambar 3.12 *Use case* kaprodi mengelola *menu* *report***

1. *Use Case* Dosen

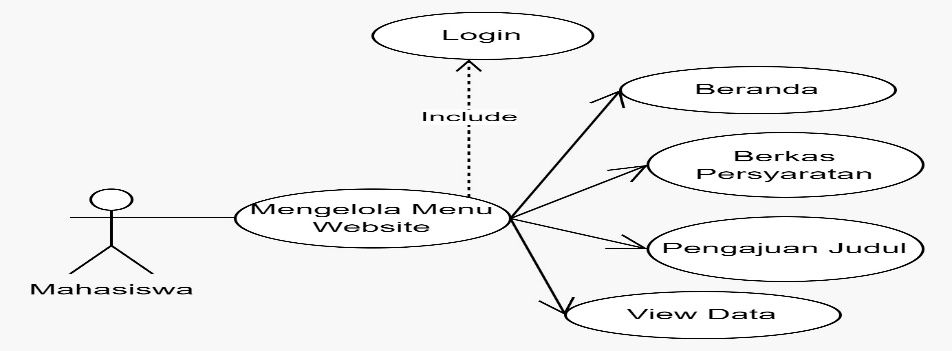


**Gambar 3.13 *Use case* dosen *menu* *website***

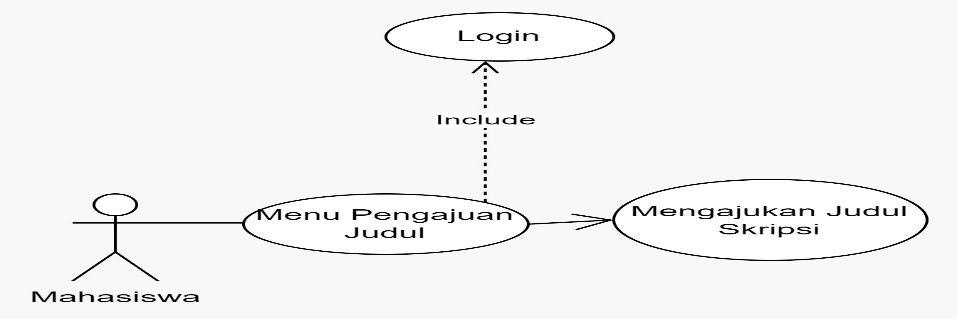


**Gambar 3.14 *Use case* dosen *menu* Mahasiswa Bimbingan data**

1. *Use Case* Mahasiswa

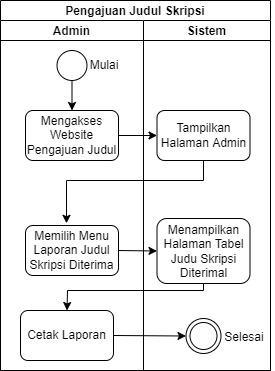


**Gambar 3.15 *Use case* mahasiswa menu *website***



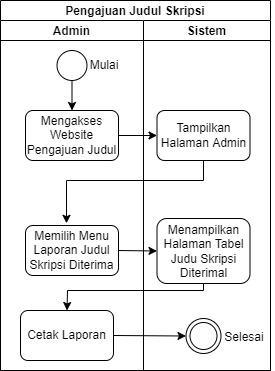
**Gambar 3.16 *Use case* mahasiswa menu Pengajuan Judul data**

1. ***Activity* Diagram**
2. *Activity* Diagram Admin Menu *Login* Pengguna



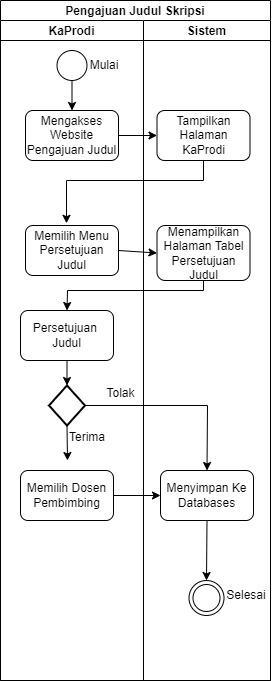
**Gambar 3.17 *Activity* diagram admin menu *login* pengguna**

1. *Activity* Diagram Admin Menu Pengajuan Judul



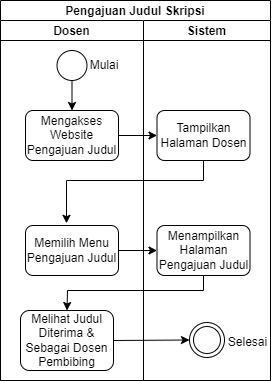
**Gambar 3.18 Activity diagram admin menu pengajuan judul**

1. *Activity* Diagram KaProdi Menu Pengajuan Judul



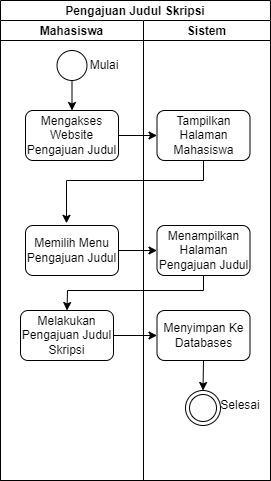
**Gambar 3.19 *Activity diagram* kaprodi menu pengajuan judul**

1. *Activity* Diagram Dosen Menu Pengajuan Judul



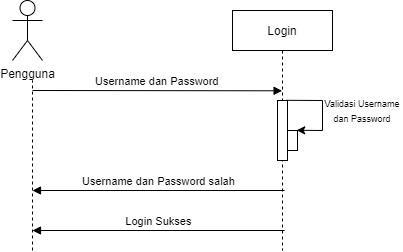
**Gambar 3.20 *Activity diagram* dosen menu pengajuan judul**

1. *Activity* Diagram Mahasiswa Menu Pengajuan Judul



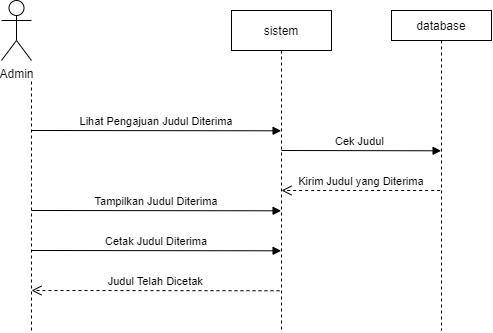
**Gambar 3.21 Activity diagram mahasiswa menu pengajuan judul**

1. ***Sequenc*e Diagram**
2. *Sequence* Diagram *Login* Pengguna



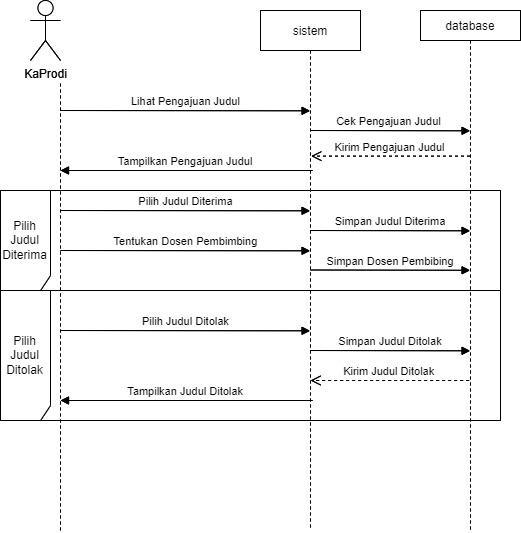
**Gambar 3.22 *Sequence* diagram login pengguna**

1. *Sequence* Diagram Admin Menu Pengajuan Judul



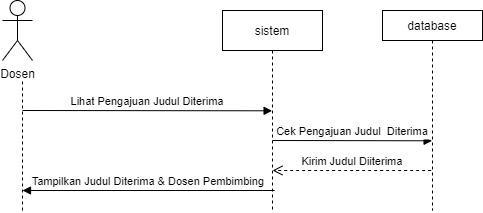
**Gambar 3.23 *Sequence* diagram admin menu pengajuan judul**

1. *Sequence* Diagram KaProdi Menu Pengajuan Judul



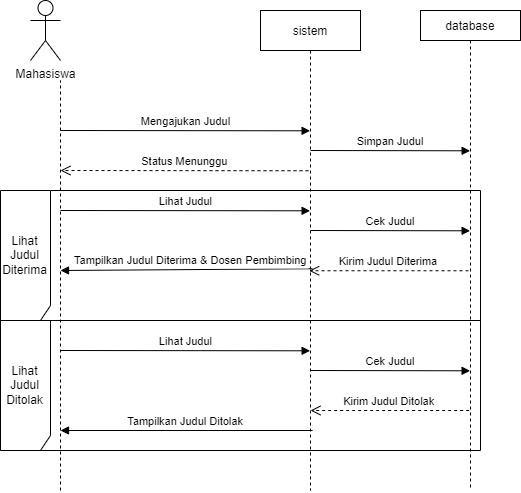
**Gambar 3.24 *Sequence* diagram kaprodi menu pengajuan judul**

1. *Sequence* Diagram Dosen Menu Pengajuan Judul



**Gambar 3.25 *Sequence* diagram dosen menu pengajuan judul**

1. Sequence Diagram Mahasiswa Menu Pengajuan Judul



**Gambar 3.26 *Sequence* diagram mahasiswa menu pengajuan judul**

1. **Desain *Database***

Perancangan *database* merupakan proses untuk menentukan isi data yang dibutuhkan untuk mendukung rancangan sistem. Model rancangan database yang dibangun adalah model *relationship* dimana seluruh tabel data yang digunakan diantaranya sebagai berikut:

**Tabel 3.2 Tabel User**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| id | *int*(11) | *Primary Key* |
| name | *varchar*(255) |  |
| username | *char(50)* |  |
| password | *varchar(255)* |  |
| role\_id | *int*(11) |  |
| Is\_active | *Int(*11) |  |
| created\_at | *datetime* |  |
| updated\_at | *datetime* |  |

**Tabel 3.3 Tabel Roles**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| id | *int*(11) | *Primary Key* |
| name | *varchar*(255) |  |

**Tabel 3.4 Tabel Lecturers**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| id | *int*(11) | *Primary Key* |
| nidn | *char(10)* |  |
| name | *varchar*(128) |  |
| field\_of\_study\_id | *Int(11)* |  |
| create\_at | *datetime* |  |
| update\_at | *datetime* |  |

**Tabel 3.5 Tabel Students**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| id | *int*(11) | *Primary Key* |
| nim | *Char(8)* |  |
| name | *varchar*(128) |  |
| generation | *varchar(4)* |  |
| major\_id | *int(11)* |  |
| create\_at | *datetime* |  |
| update\_at | *datetime* |  |

**Tabel 3.6 Tabel titles**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| *id* | *int*(11) | *Primary Key* |
| *name* | *varchar*(255) |  |
| *description* | *char(50)* |  |
| *notes* | *varchar(255)* |  |
| *category\_id* | *int*(11) |  |
| *Essay\_id* | *Int(*11) |  |
| *approve* | *enum(‘proses’,’terima’,’tolak’* |  |
| *mentor\_one* | *char(10)* |  |
| *mentor\_two* | *char(10)* |  |
| *created\_at* | *datetime* |  |
| *updated\_at* | *datetime* |  |

**Tabel 3.7 Tabel essays**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| *id* | *int*(11) | *Primary Key* |
| *date* | *date* |  |
| *student\_id* | *int(*11*)* |  |
| *id\_approval* | *int*(11) |  |
| *created\_at* | *datetime* |  |
| *updated\_at* | *datetime* |  |

**Tabel 3.8 Tabel field\_of\_studys**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| *id* | *int*(11) | *Primary Key* |
| *name* | *varchar*(128) |  |
| *created\_at* | *datetime* |  |
| *updated\_at* | *datetime* |  |

**Tabel 3.9 Tabel majors**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| *id* | *int*(11) | *Primary Key* |
| *name* | *varchar*(128) |  |
| *created\_at* | *datetime* |  |
| *updated\_at* | *datetime* |  |

**Tabel 3.10 Tabel categories**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| *id* | *int*(11) | *Primary Key* |
| *name* | *varchar*(128) |  |
| *created\_at* | *datetime* |  |
| *updated\_at* | *datetime* |  |

**Tabel 3.11 Tabel files**

|  |  |  |
| --- | --- | --- |
| **Nama Field** | **Tipe Data** | **Keterangan** |
| *id* | *int*(11) | *Primary Key* |
| *approve* | *enum(‘proses’,’terima’,’tolak’* |  |
| *notes* | *text* |  |
| *file\_transkrip\_nilai* | *text* |  |
| *file\_ktm* | *text* |  |
| *file\_bukti\_pembayaran* | *text*) |  |
| *student\_id* | *int(11)* |  |
| *created\_at* | *datetime* |  |
| *updated\_at* | *datetime* |  |

Perancangan *database* ini menggunakan Diagram Hubungan Entitas atau *entity relationship diagram* (*ERD*) dalam perancangannya hal ini melihat dari *database* yang dibuat diatas mengenai data tabel berdasarkan fungsi yang diberikan. *ERD* merupakan sebuah model diagram untuk mengetahui hubungan antara beberapa tabel berdasarkan isi dari tabel tersebut. Berikut merupakan *ERD* yang digunakan dalam perancangan sistem ini.



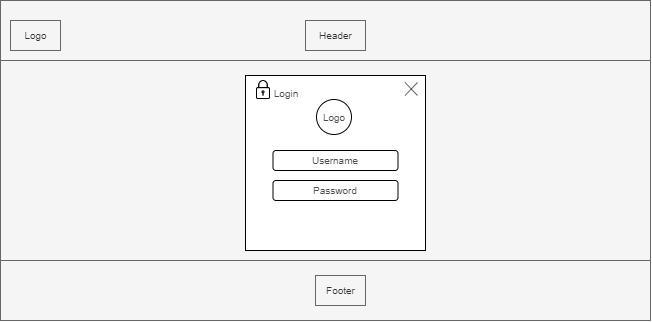
**Gambar 3.27 *Entity Relationship Diagram***

1. ***User Interface***

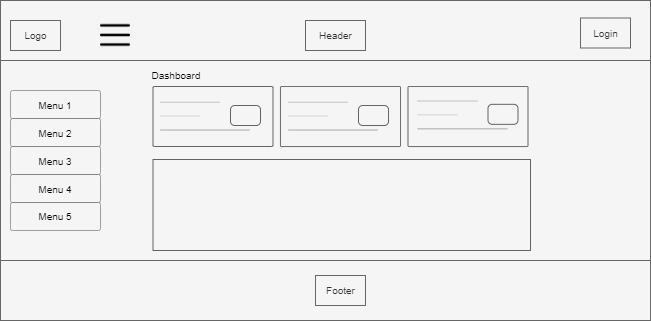
Antarmuka pengguna atau *user interface (UI*) adalah suatu istilah yang digunakan untuk menggambarkan tampilan dari mesin atau komputer yang berinteraksi langsung dengan pengguna. Tujuan sebuah *interface* adalah mengkomunikasikan fitur-fitur system yang tersedia agar *user* mengerti dan dapat menggunakan *website* tersebut. Desain dan penyusunan tampilan antarmuka perlu diperhatikan untuk menghasilkan tampilan yang bagus. Berikut ini adalah tampilan penjelasan sistem informasi pengajuan judul tugas akhir berbasis *website* di Program Studi Pendidikan Teknologi Informasi yang akan dibuat.



**Gambar 3.28 *User interface* tampilan menu utama**



**Gambar 3.29 *User interface* tampilan Menu *Login***



**Gambar 3.30 User interface tampilan menu *dashboard***

**3. Penulisan Kode**

Proses tahap penulisan kode, peneliti melakukan penyesuaian dengan desain produk yang sudah direncanakan, selanjutnya dilakukan ke tahap pemrograman atau sering disebut dengan *development*. *Software* yang digunakan adalah *CodeIgniter* 3 dan *Visual Studio Code* untuk memudahkan dalam melakukan proses pemrograman dan implementasi desain produk.

<?php

defined('BASEPATH') OR exit('No direct script access allowed');

class Dashboard extends CI\_Controller {

$data = "data dari *server* may*SQL* yang akan di tampilkan";

$view = "template *boostrap* yang akan menampilkan data";

viewTemplate($data, $view);

}

?>

**4. Pengujian**

Untuk metode pengujian yang dilakukan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian *Black Box*. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Berdasarkan keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahan-kesalahannya.

Berikut kisi-kisi uji *black box* dalam pengujian sistem berdasarkan perencanaannya yang ditampilkan pada tabel 3.12 Kisi-kisi uji *black box*:

**Tebel 3.12 Kisi-kisi Uji *Black Box* Testing**

|  |  |  |  |
| --- | --- | --- | --- |
| **Aktivitas Pengujian** | **Realisasi yang diharapkan** | **Hasil Pengujian** | **Kesimpulan** |
| Masuk ke halaman *login* | Masukan *username* dan *password* | Menampilkan halaman *dashboard* pada sistem | Berhasil |
| Mahasiswa *klik* pada *menu* pengajuan judul tugas akhir | Menampilkan halaman pengajuan judul tugas akhir | Tombol “Tambah Judul” berfungsi sesuai yang diharapkan | Berhasil |
| KaProdi *klik* pada *menu* pengajuan judul tugas akhir | Menampilkan halaman daftar pengajuan judul tugas akhir | Tombol “persetujuan” berfungsi sesuai yang diharapkan | Berhasil |
| Dosen *klik* pada *menu* Mahasiswa Bimbingan | Menampilkan halaman daftar mahasiswa bimbingan | Berfungsi sesuai yang diharapkan | Berhasil |
| Admin *klik* pada *menu* *report* pengajuan judul yang disetujui atau ditolak | Menampilkan halaman *report* pengajuan judul tugas akhir | Berfungsi sesuai yang diharapkan | Berhasil |

**5. Implementasi**

Proses tahap implementasi, peneliti melakukan implementasi program dan uji coba serta evaluasi perancangan sistem informasi pengajuan judul tugas akhir menggunakan *codeIgniter 3* berbasis *website* pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari, yang bertujuan untuk mengetahui hasil dari kebutuhan fungsional sistem yang telah disusun dan dicocokan dengan hasil implementasi dalam bentuk perancangan sistem informasi.

1. **Tempat dan Waktu Penelitian**

Penelitian ini dilaksanakan di Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari pada bulan agustus 2022 sampai mei 2023.

1. **Alat dan Bahan Penelitian**

Penelitian ini membutuhkan beberapa *hardware* (perangkat keras) dan *software* (perangkat lunak). Peralatan *hardware* maupun *software* dipilih disesuaikan dengan kebutuhan sebagai berikut:

1. **Alat Penelitian**
2. Hardware (Perangkat Keras):

Laptop *HP* : (Sistem model: *14s-dk0126AU, Memory: 8048 RAM, Processor: AMD Ryzen 3 3200U with Radeon Vega Mobile Gfx 2.60 GHz, Operation System: Windows 11 Home Single Language 64-bit*.

Keyboard key: menginput data.

1. Software (Perangkat Lunak)

*Xampp*  : aplikasi pengelola *database*

*Visual Studio Code* : aplikasi *text editor* untuk menulis *script code*

*CI (Codeigniter)* : *framework* untuk membuat *backend*

*Bootstrap* : *framework* untuk membuat *front end*

*Google Crome* :aplikasi untuk menampilkan hasil *user* *interface*

*Draw Io* : aplikasi membuat perancangan *website*

1. **Bahan Penelitian**

Bahan yang digunakan dalam penelitian ini sebagai berikut:

1. Observasi

Penulis melakukan pengamatan secara langsung ke lapangan dengan mengamati perancangan *website* yang ada di Program Studi Pendidikan Teknologi Informasi.

1. Dokumentasi

Dokumentasi merupakan tahapan untuk mengumpulkan data dengan menggunakan sumber-sumber tertulis seperti mengenai data dari dosen dan *reviewer* dari kedua data ini penulis dapat mengetahui proses atau tahapan yang ada di tempat penelitian.

1. Studi Literatur

Studi literatur dalam penelitian ini yaitu mengumpulkan data ke perpustakaan dari jurnal, dan buku, skripsi yang relevan yang menunjang peneliti membangun sebuah sistem informasi lebih baik kedepannya.

**BAB IV**

**HASIL PENELITIAN DAN PEMBAHASAN**

1. **Hasil Penelitian**
2. **Implementasi Simpelta Program Studi pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari**

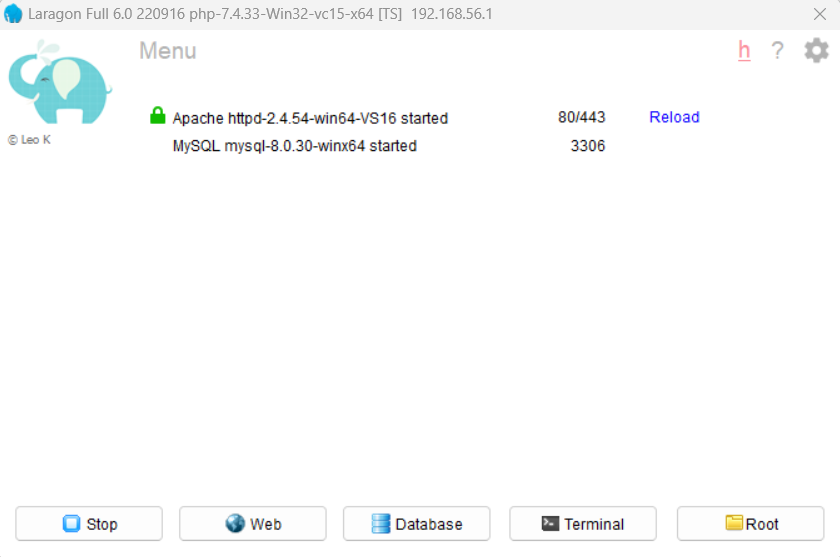
Hasil dari penelitian ini adalah terciptanya SIMPELTA PTI dimana dalam SIMPELTA PTI ini didominasi oleh teks bahasa pemrograman seperti *HTML, CSS, JavaScript* dan *PHP*. Dari semua bahasa pemrograman tersebut dibangun secara terstruktur menggunakan *framework Codeigniter* kemudian dipercantik dengan *framework bootstrap*, lalu dikerjakan melalui aplikasi *Visual Studio Code* dan kemudian disimpan dalam sebuah folder yang telah disediakan oleh *Laragon* yaitu folder *www*, selama proses pengerjaan SIMPELTA PTI ini masih diakses secara *offline* yang memanfaatkan *software Laragon* sebagai *server* mandiri *(localhost),* karena didalam *Laragon* tersebut terdapat fitur yang sangat dibutuhkan saat proses pengerjaan SIMPELTA PTI ini seperti *Apache* yang digunakan sebagai *web server* dan *MySql* sebagai *database server* sehingga hasil kerja dapat dijalankan atau dilihat melalui *web browser* dan *browser* yang saya gunakan adalah *Chrone.*

1. **Database**

*Database* merupakan wadah untuk menampung semua data berupa *text* yang diinputkan oleh *user* kemudian tersusun berdasarkan tabel yang telah disediakan oleh *developer* atau pengembang sistem. *Database* ini menggunakan *MySQL*, adapun tabel yang digunakan sebanyak 10 tabel, untuk menjalankan *server* *MySQL* dan *apache* di permudah menggunakan aplikasi *Laragon*.

Berikut tampilan *Laragon* beserta tabel yang digunakan:

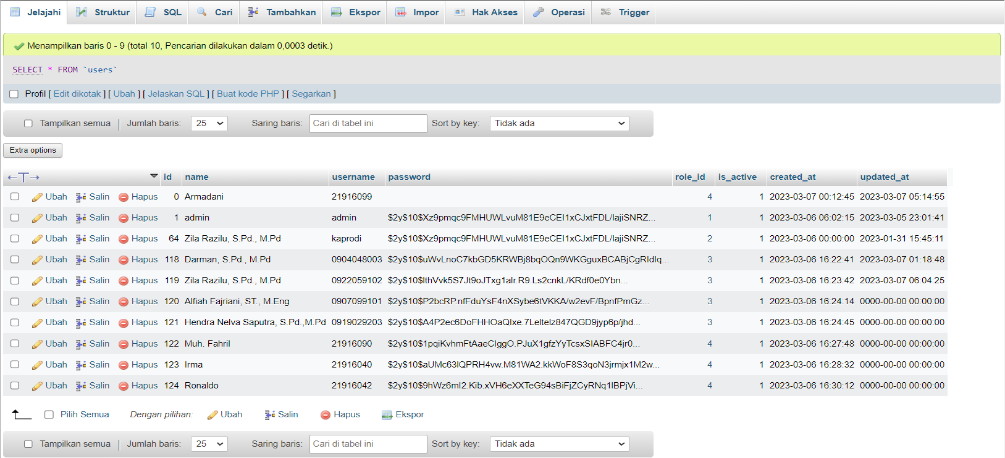
1. Tampilan Laragon



**Gambar 4.1 Tampilan Laragon**

1. Tampilan Tabel *user’s*

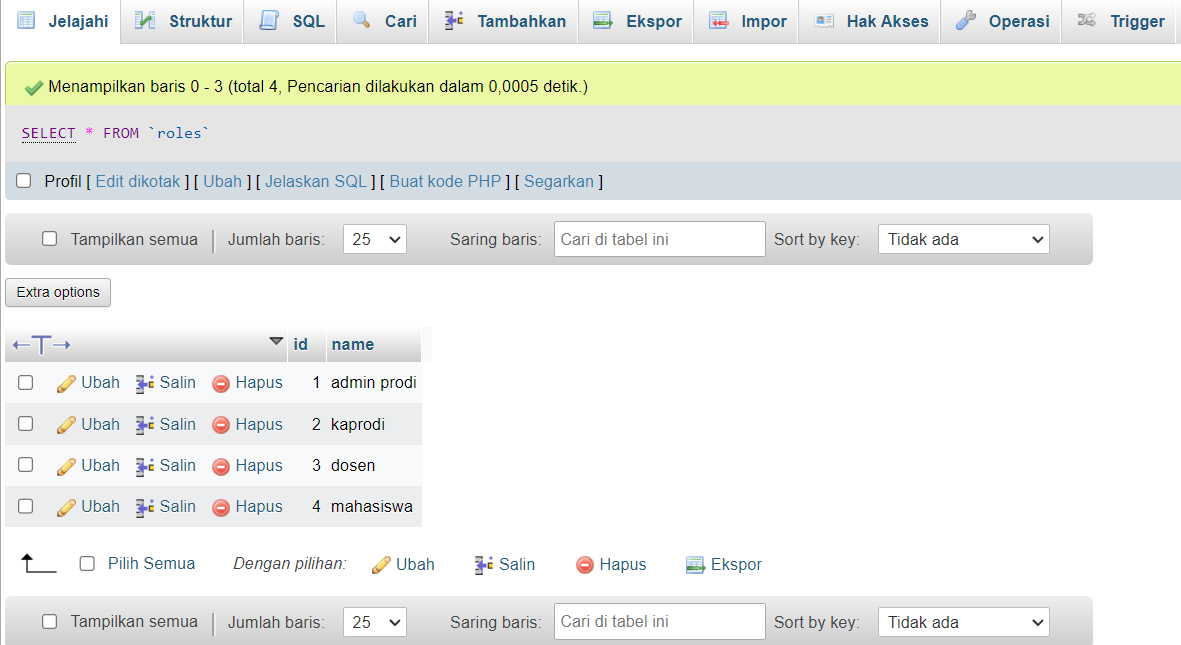
Tabel *users* ini digunakan untuk menampung data *user* baik untuk login maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *user.*



**Gambar 4.2 Tampilan tabel *user***

1. Tampilan Tabel *roles*

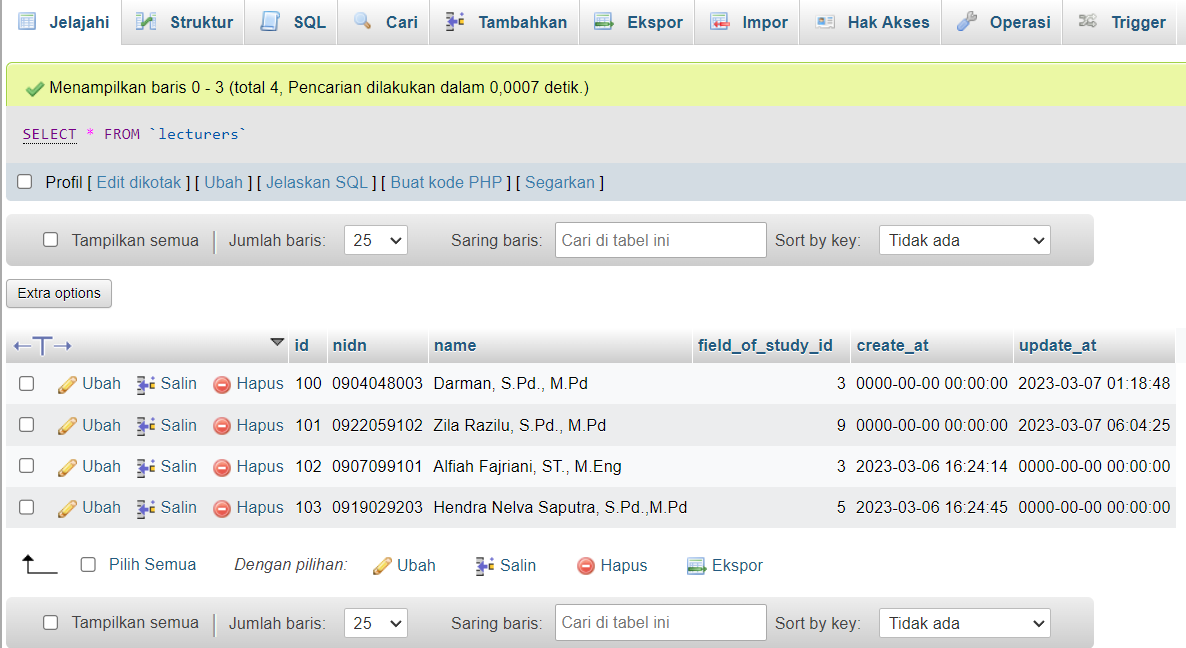
Tabel *roles* digunakan untuk menentukan level user baik level admin, kaprodi, dosen dan mahasiswa.



**Gambar 4.3 Tampilan tabel *roles***

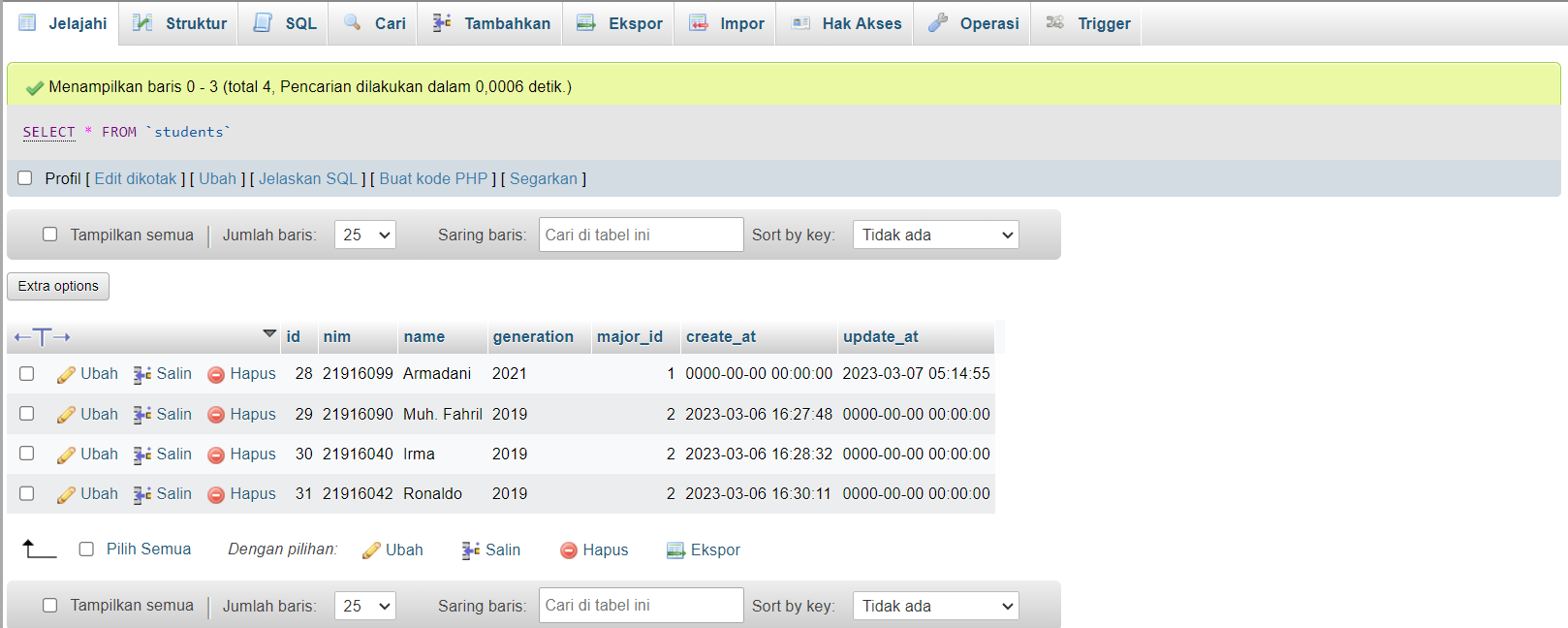
1. Tampilan Tabel *lecturers*

Tabel *lecturers* ini digunakan untuk menampung data *lecturer* baik untuk pemilihan dosen pembimbing maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *lecturer*. Tabel *lecturer* juga dapat digunakan untuk memilih dosen pembimbing untuk mahasiswa. Data yang terdapat pada tabel *lecturer*, seperti bidang keahlian/studi, dapat membantu mahasiswa dalam memilih dosen pembimbing yang tepat.

**Gambar 4.4 Tampilan tabel *lecturers***

1. Tampilan Tabel *students*

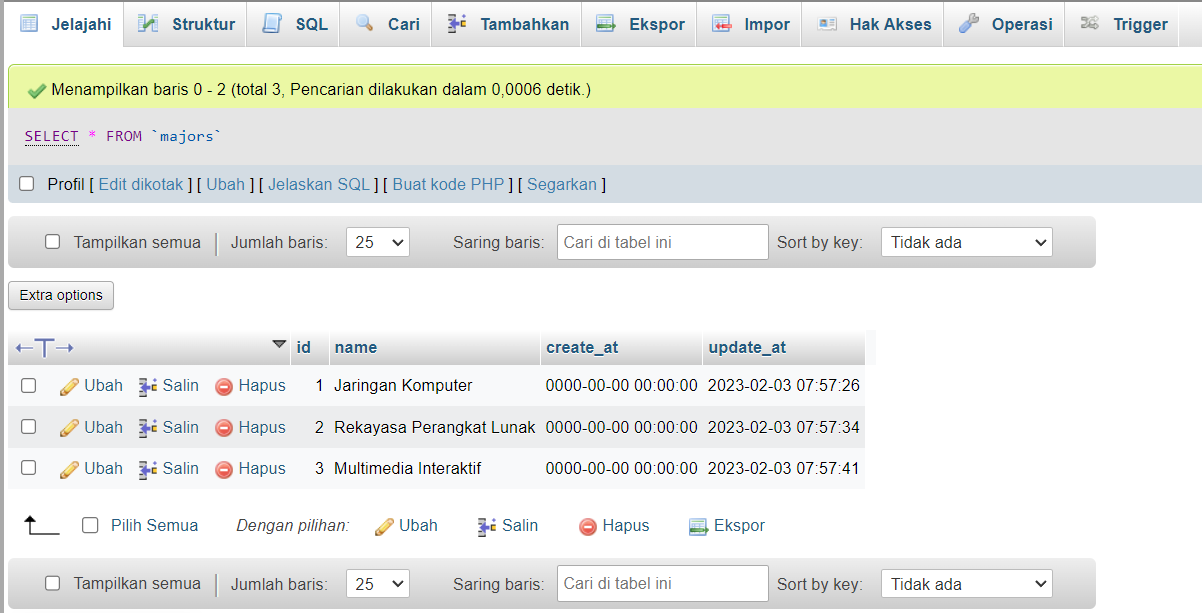
Tabel *students* ini digunakan untuk menampung data *students* baik untuk melihat jumlah mahasiswa di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *students*. Data yang terdapat pada tabel *students*, seperti jurusan, dapat membantu prodi dalam melihat jurusan yang dipilih oleh mahasiswa.



**Gambar 4.5 Tampilan tabel *students***

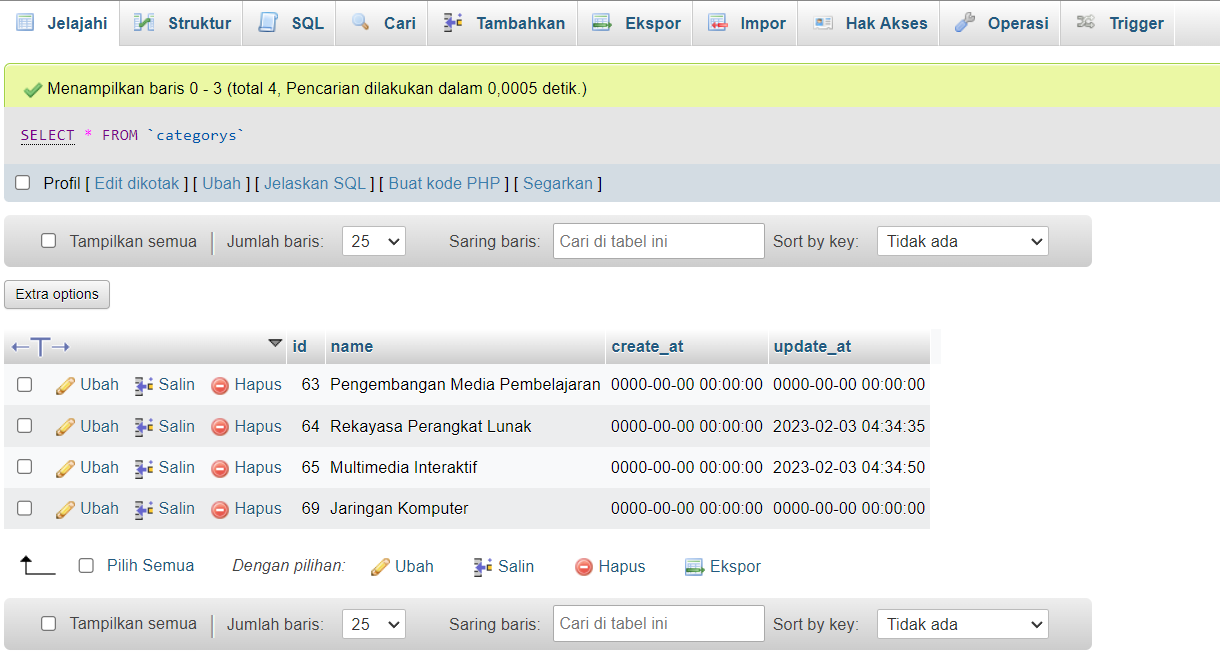
1. Tampilan Tabel *majors*

Tabel *majors* ini digunakan untuk menampung data *majors* baik untuk melihat jumlah jurusan di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *majors*.

**Gambar 4.6 Tampilan tabel *majors***

1. Tampilan Tabel *categories*

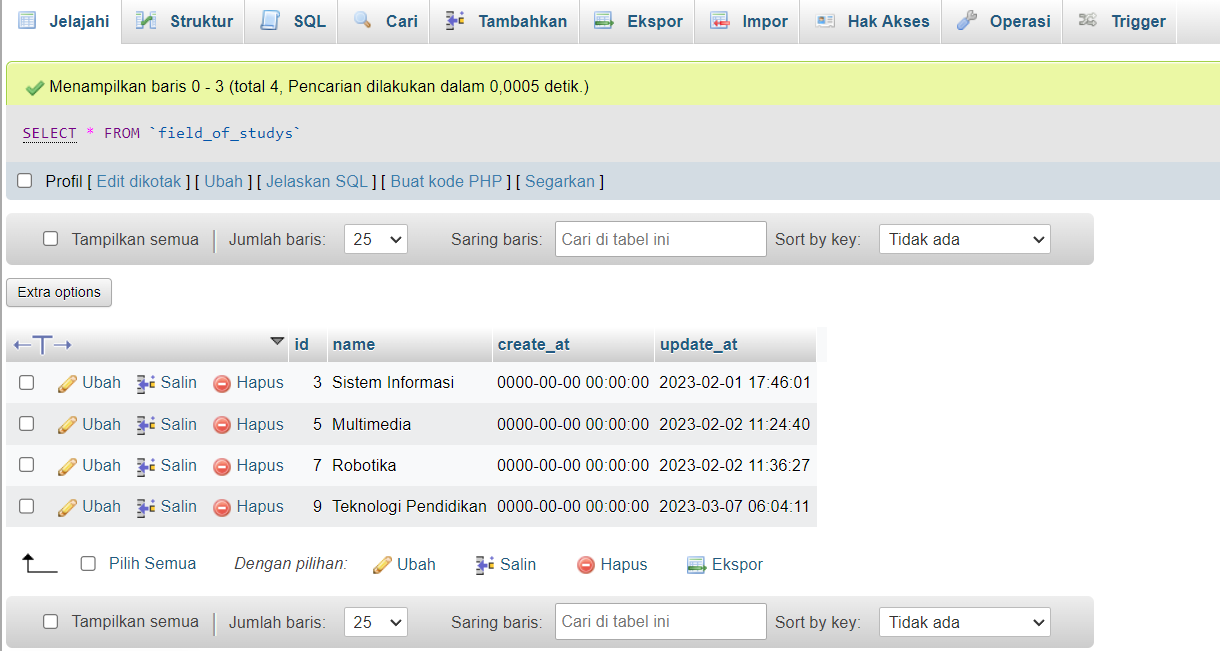
Tabel *categories* ini digunakan untuk menampung data *categories* baik untuk melihat jumlah *categories* di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *categories*.



**Gambar 4.7 Tampilan tabel *category***

1. Tampilan Tabel *field\_of\_studys*

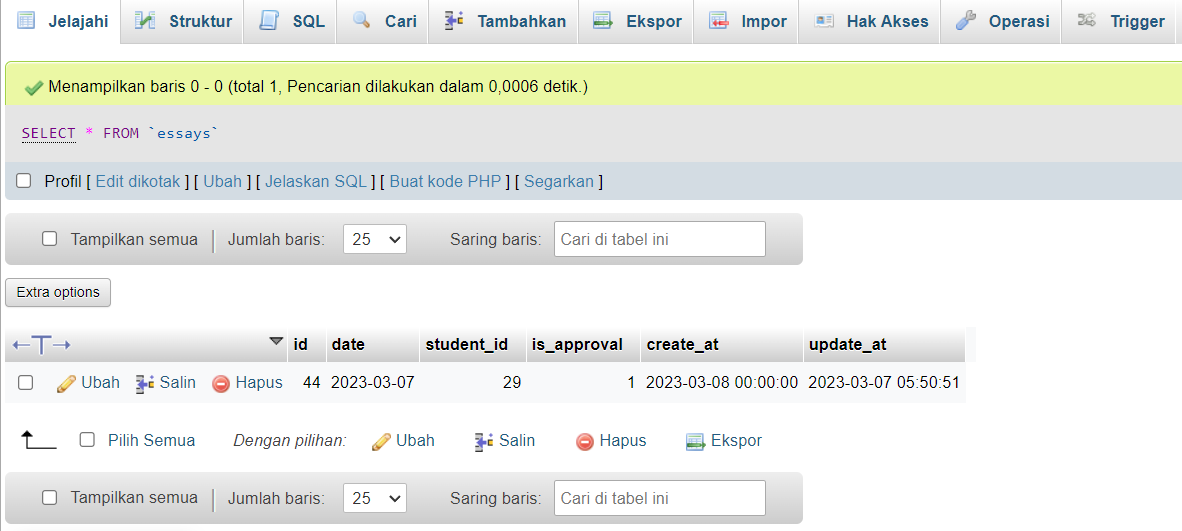
Tabel *field of study* ini digunakan untuk menampung data *field of study* baik untuk melihat jumlah *field of study* di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *field\_of\_studys*.



**Gambar 4.8 Tampilan tabel *field\_of\_studys***

1. Tampilan Tabel *essays*

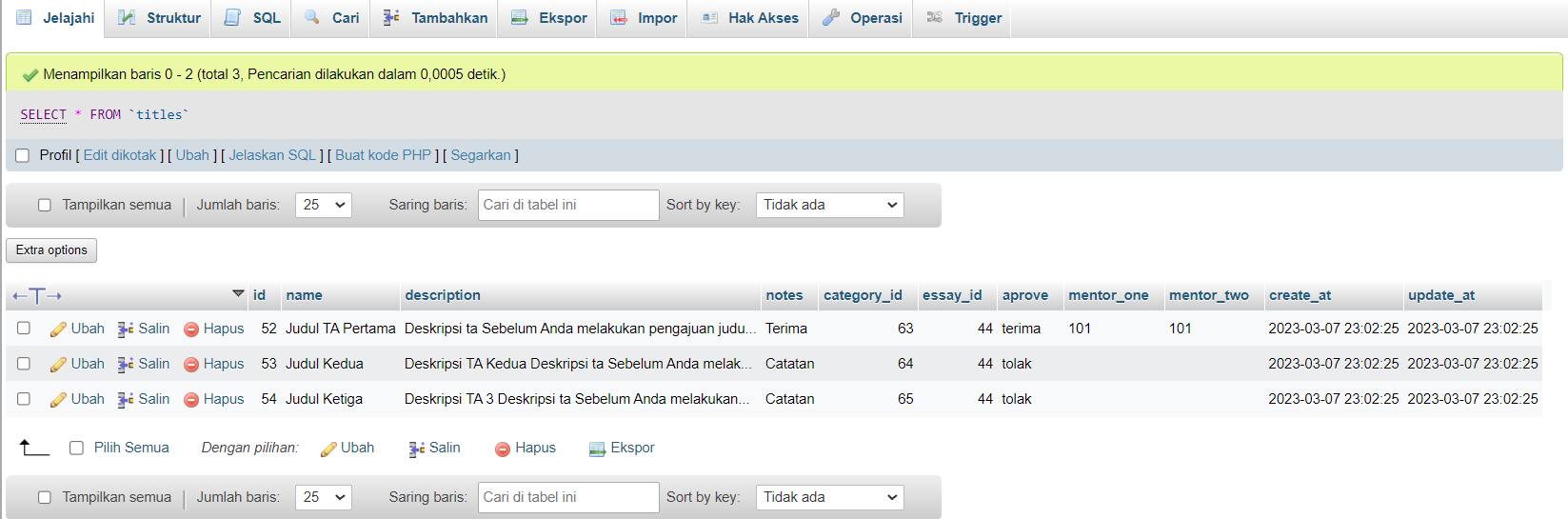
Tabel *essays* ini digunakan untuk menampung data pengajuan baik untuk melihat jumlah pengajuan mahasiswa di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *essays*. Data yang terdapat pada tabel *essays*, seperti date, *student\_id*, dapat membantu prodi dalam melihat waktu pengajuan mahasiswa.



**Gambar 4.9 Tampilan tabel *essays***

1. Tampilan Tabel *titles*

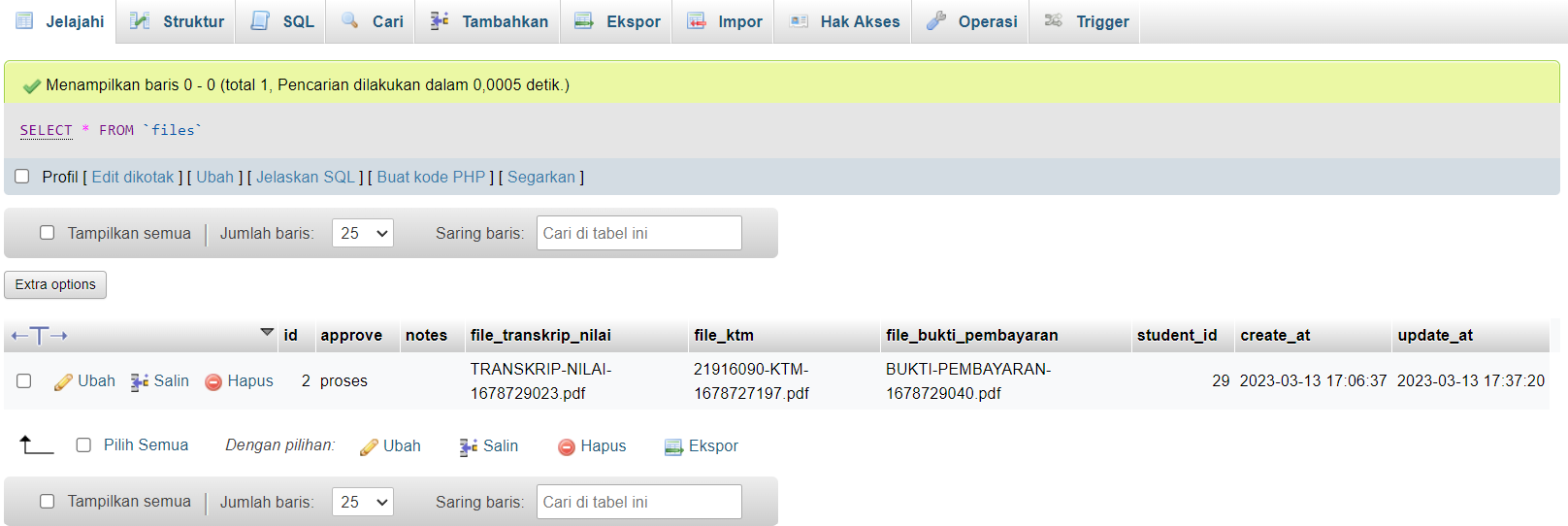
Tabel *titles* ini digunakan untuk menampung data judul pengajuan baik untuk melihat jumlah judul pengajuan mahasiswa di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *titles*. Data yang terdapat pada tabel *titles*, seperti approve dapat membantu prodi dalam menyetujui judul pengajuan mahasiswa.



**Gambar 4.10 Tampilan tabel *titles***

1. Tampilan Tabel *files*

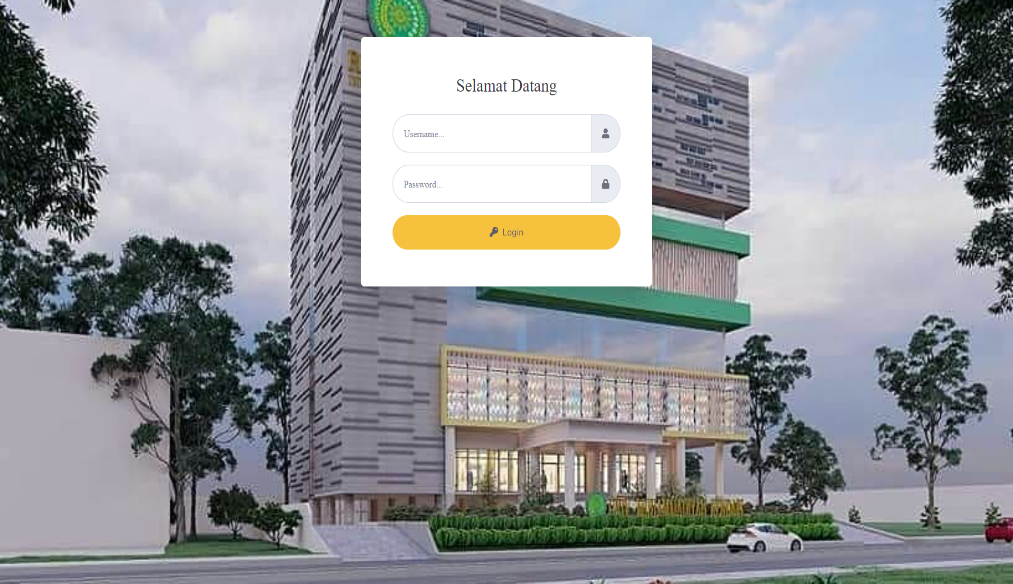
Tabel *files* ini digunakan untuk menampung data berkas pengajuan baik untuk melihat jumlah berkas pengajuan mahasiswa di program studi pendidikan teknologi informasi maupun untuk kebutuhan relasi pada tabel-tabel lainnya yang memerlukan data *files*. Data yang terdapat pada tabel *files*, seperti approve dapat membantu prodi dalam menyetujui berkas pengajuan mahasiswa.



**Gambar 4.11 Tampilan tabel *files***

1. **Tampilan SIMPELTA PTI Universitas Muhammadiyah Kendari**
2. **Halaman Login**

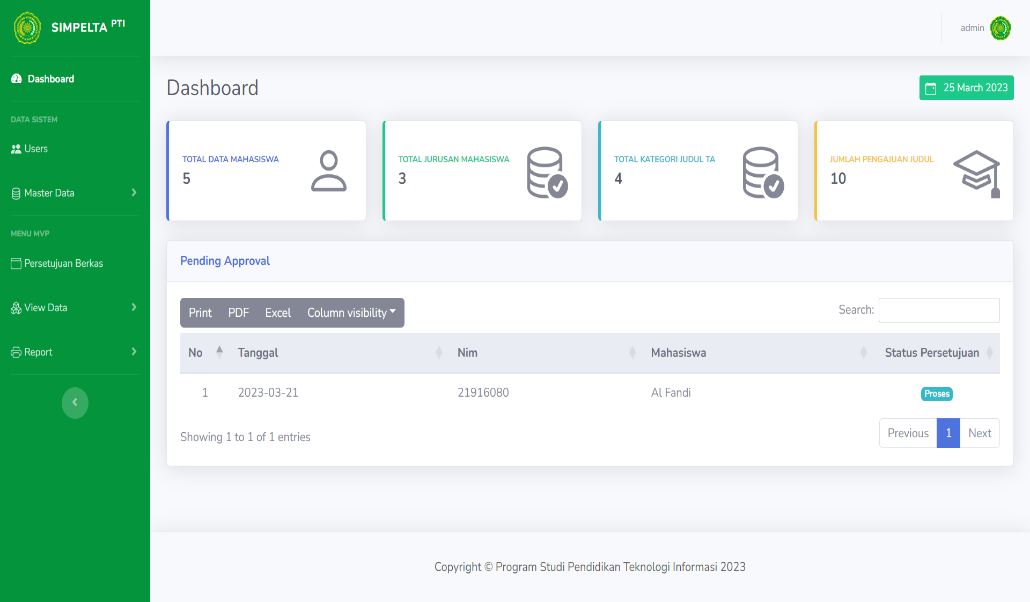
Halaman *login* digunakan sebagai pintu utama untuk masuk ke-*website* *SIMPELTA PTI* ini, pada halaman login ini terdapat inputan username dan password.



**Gambar 4.12 Tampilan *Login***

1. **Halaman Admin Prodi**
2. Dashboard Admin Prodi

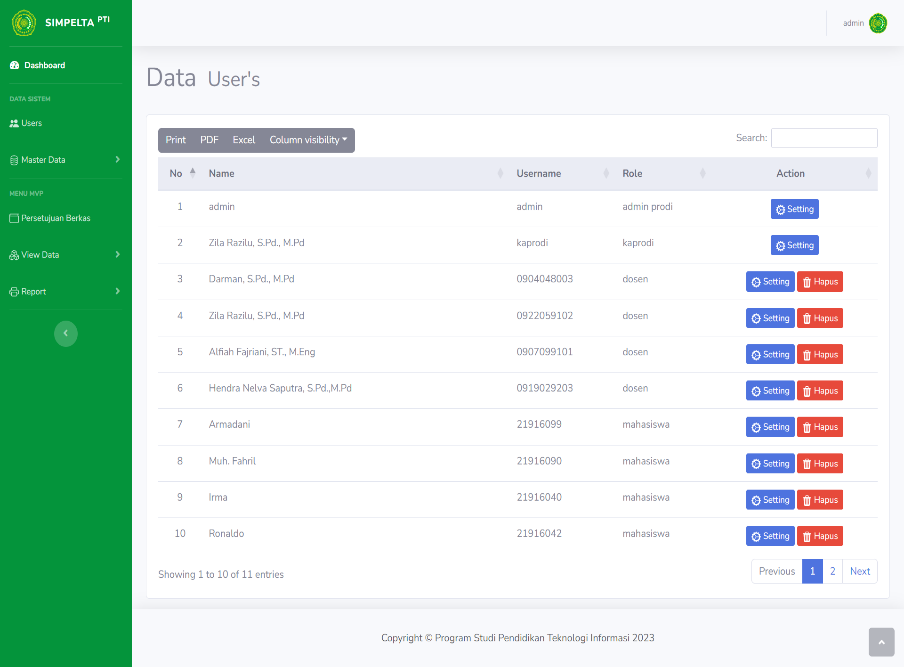
Halaman *dashboard* *admin* prodi ini menampilkan jumlah statistik data mahasiswa, jurusan, kategori dan pengajuan judul tugas akhirserta tabel *pending approval* judul tugas akhir yang diajukan atau sedang diproses.



**Gambar 4.13 Tampilan Menu *Dashboard***

1. User’s

Menu *User’s* Data adalah tempat di mana *admin* prodi dapat mengelola semua informasi terkait pengguna di dalam sistem. Di sini, *admin* prodi dapat *setting password* dan menghapus akun pengguna. Menu ini dibuat untuk membantu *admin* prodi mengelola pengguna dengan mudah dan efisien. Dengan akses mudah ke informasi pengguna, *admin* prodi dapat mengelola pengguna dengan lebih efektif dan meminimalkan risiko keamanan. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



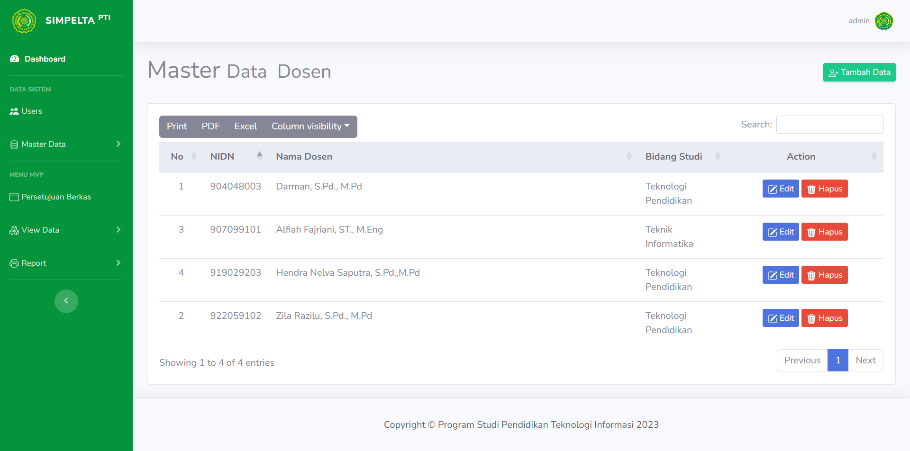
**Gambar 4.14 Tampilan Menu *User’s***

1. Master Data

Halaman Menu Master Data memiliki 5 submenu seperti Data Dosen, Data Mahasiswa, Data Kategori, Data Bidang Studi, dan Data Jurusan digunakan sebagai tempat di mana *admin* prodi dapat mengelola semua informasi penting yang terkait dengan data SIMPELTA PTI. Berikut submenu yang terdapat dalam menu master data:

1. Data Dosen

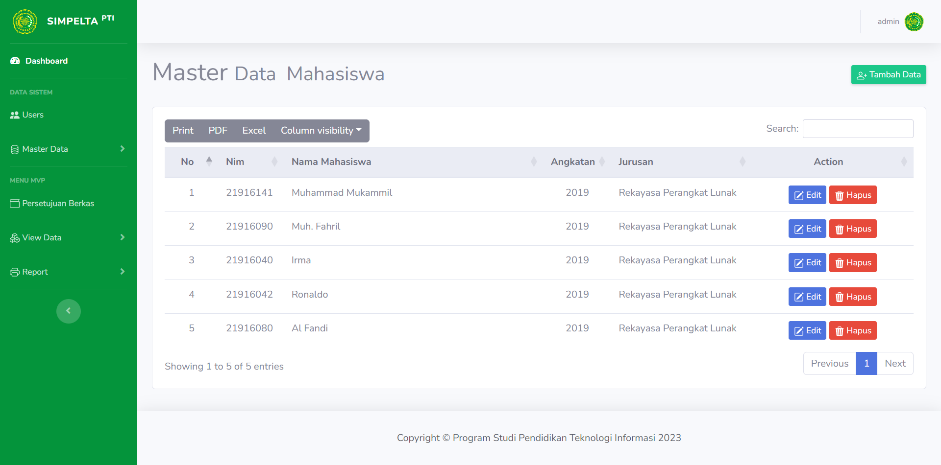
Halaman *sub menu* Data Dosen digunakan sebagai tempat untuk menyimpan data-data dosen yang ada di program studi pendidikan Teknologi informasi. Selain itu data dosen juga dapat digunakan sebagai pemilihan pembimbing tugas akhir yang diberikan oleh kaprodi kepada mahasiswa. Di sini, *admin* prodi dapat menambahkan, mengedit, dan menghapus Data Dosen. Menu ini dibuat untuk membantu *admin* prodi untuk mengelola data dengan mudah dan efisien. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.15 Tampilan *Menu* Data Dosen**

1. Data Mahasiswa

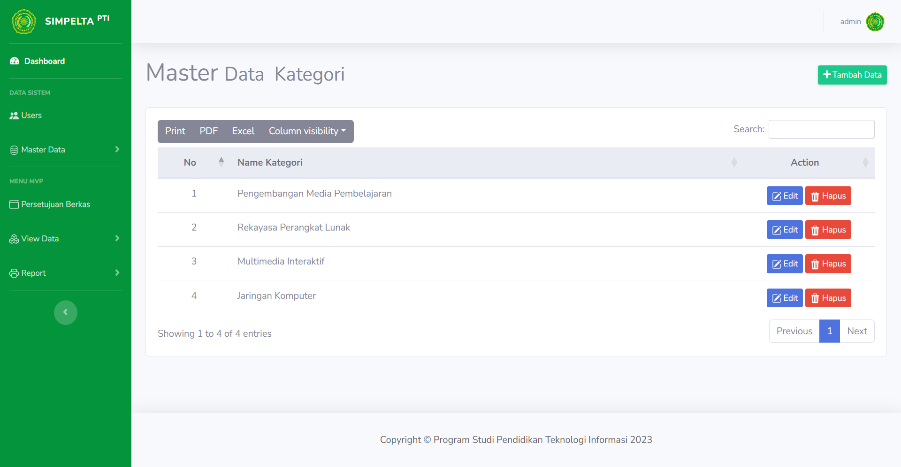
Halaman *sub menu* Data Mahasiswa digunakan sebagai tempat untuk menyimpan data-data mahasiswa yang ada di program studi pendidikan Teknologi informasi. Di sini, *admin* prodi dapat menambahkan, mengedit, dan menghapus Data Mahasiswa. *Menu* ini dibuat untuk membantu *admin* prodi untuk mengelola data dengan mudah dan efisien. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.16 Tampilan *Menu* Data Mahasiswa**

1. Data Kategori

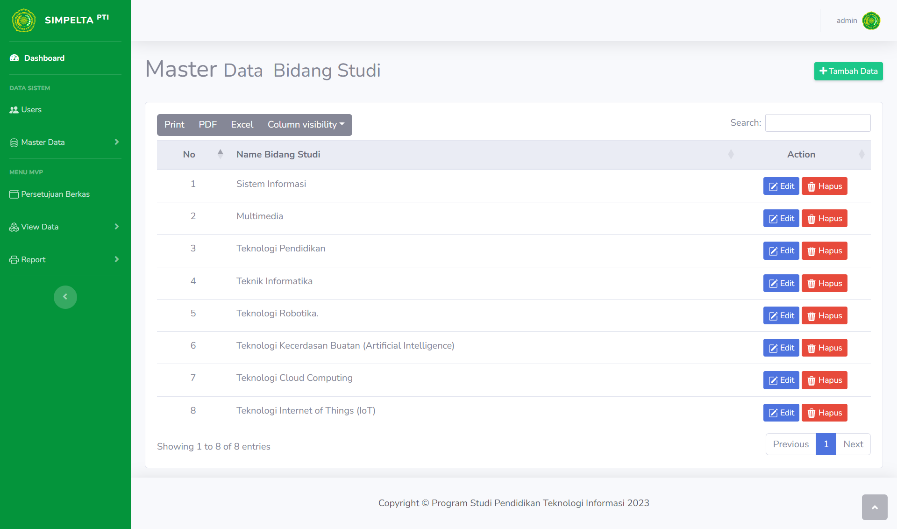
Halaman sub menu Data Kategori digunakan sebagai tempat untuk menyimpan data-data kategori judul tugas akhir yang ada di program studi pendidikan Teknologi informasi. Di sini, *admin* prodi dapat menambahkan, mengedit, dan menghapus Data Kategori. Menu ini dibuat untuk membantu *admin* prodi untuk mengelola data dengan mudah dan efisien. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.17 Tampilan *Menu* Data Kategori**

1. Data Bidang Studi

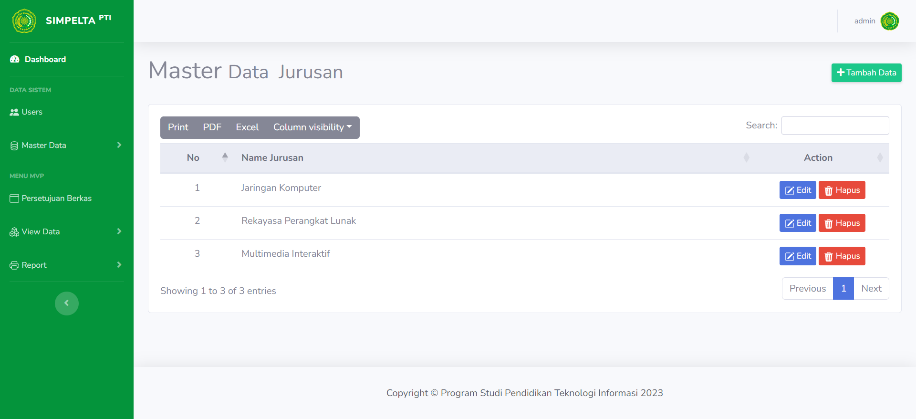
Halaman *sub menu* Data Bidang Studi digunakan sebagai tempat untuk menyimpan data-data bidang studi dosen yang ada di Program Studi Sendidikan Teknologi Informasi. Di sini, *admin* prodi dapat menambahkan, mengedit, dan menghapus Data Bidang Studi. Menu ini dibuat untuk membantu *admin* prodi untuk mengelola data dengan mudah dan efisien. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.18 Tampilan *Menu* Data Bidang Studi**

1. Data Jurusan

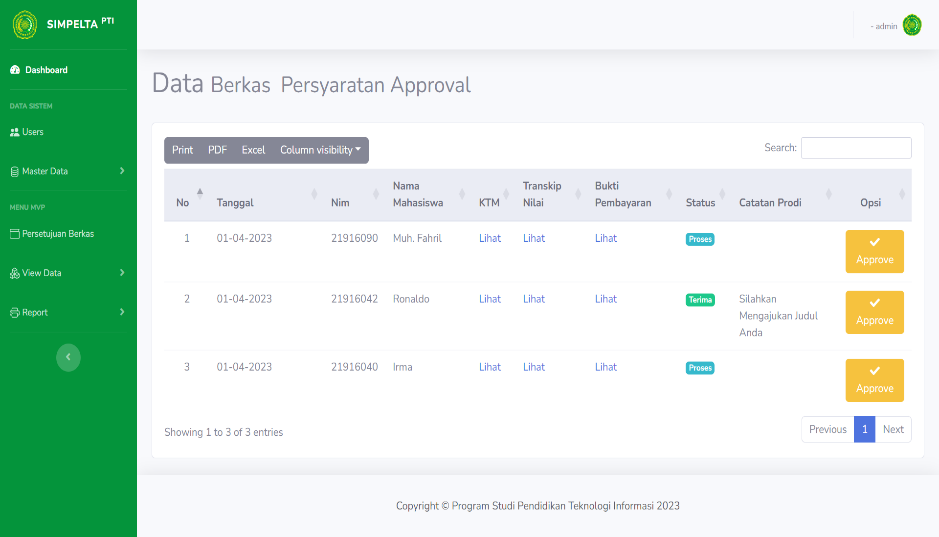
Halaman *sub menu* Data Jurusan digunakan sebagai tempat untuk menyimpan data-data jurusan mahasiswa yang ada di Program Ptudi Pendidikan Teknologi informasi. Di sini, *admin* prodi dapat menambahkan, mengedit, dan menghapus Data Jurusan. *Menu* ini dibuat untuk membantu *admin* prodi untuk mengelola data dengan mudah dan efisien. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan *mengklik* *column visibility*.



**Gambar 4.19 Tampilan *Menu* Data Jurusan**

1. Persetujuan Berkas

Halaman *Menu* Persetujuan Berkas sebagai tempat untuk menyimpan berkas persyaratan yang diajukan oleh mahasiswa. Di sini, *Admin* prodi dapat menyetujui berkas serta memberikan catatan kepada mahasiswa. Dengan akses mudah ke menu Persetujuan Berkas, *admin* prodi dapat membuat keputusan yang lebih baik dan lebih cepat, serta mengoptimalkan dalam manajemen berkas persyaratan sebelum menyetujui judul tugas akhir mahasiswa. Selain itu, *admin* prodi dapat menjaga berkas persyaratan pada SIMPELTA PTI tetap teratur dan terorganisir dengan baik. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



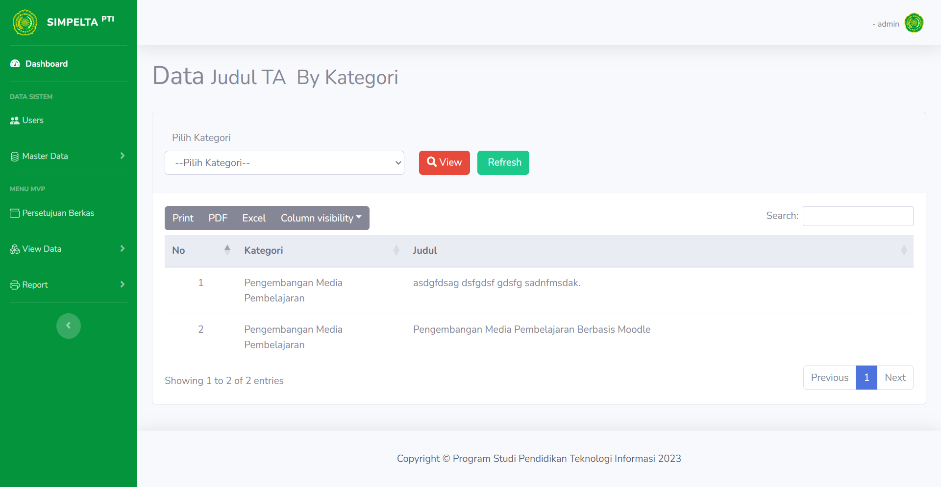
**Gambar 4.20 Tampilan *Menu* Persetujuan Berkas**

1. View Data

Halaman *Menu View* Data memiliki 3 submenu seperti Judul TA By Kategori, Judul TA By Tahun Ajuan, dan Pembimbing digunakan sebagai tempat di mana *admin* prodi dapat melihat semua informasi judul tugas akhir mahasiswa serta jumlah bimbingan pada setiap pembimbing. Berikut submenu yang terdapat dalam *menu view* data:

1. Judul TA by Kategori

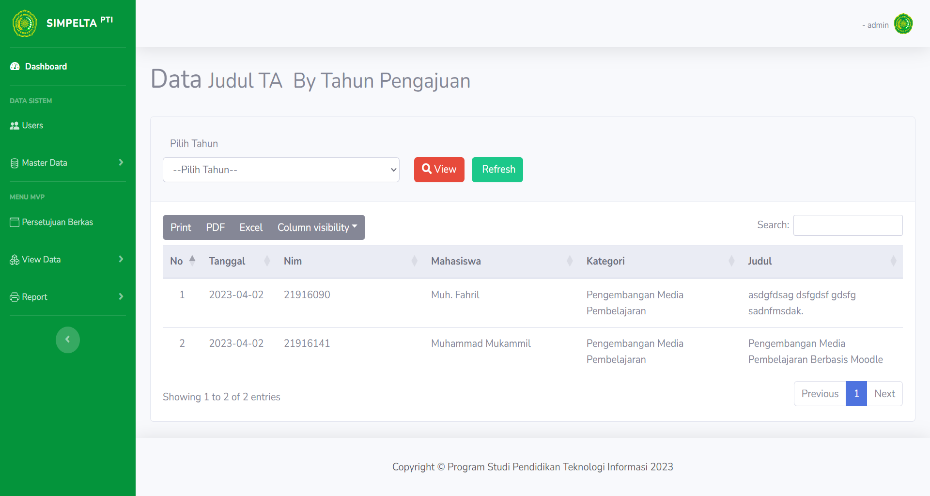
Halaman Judul TA by Kategori adalah tempat di mana *admin* prodi dapat melihat semua data judul yang telah disetujui oleh kaprodi. Di sini, *admin* prodi dapat dengan mudah mencari data judul tugas akhir yang dibutuhkan, filter data berdasarkan kategori dan mengurutkan data sesuai dengan *preferensi*. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga admin prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.21 Tampilan *Menu* Judul TA *By* Kategori**

1. Judul TA by Tahun Ajuan

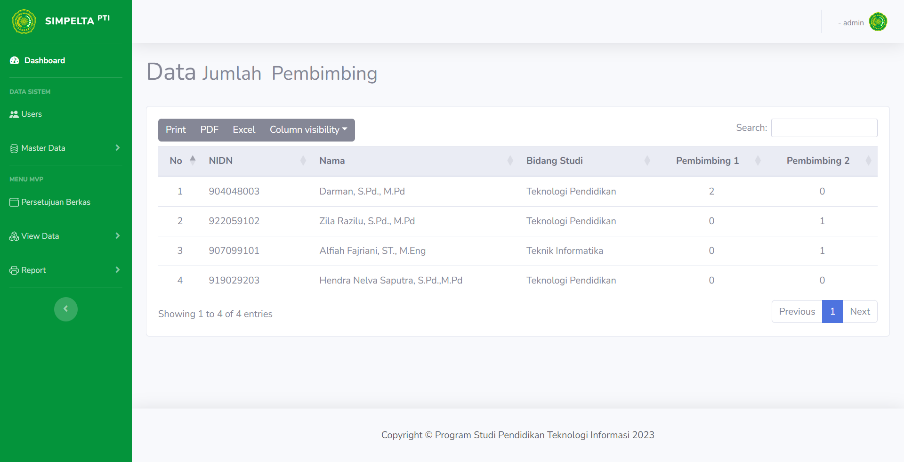
Halaman Judul TA by Tahun Ajuan adalah tempat di mana *admin* prodi dapat melihat semua data judul yang telah disetujui oleh kaprodi. Di sini, *admin* prodi dapat dengan mudah mencari data judul tugas akhir yang dibutuhkan, *filter* data berdasarkan tahun ajuan dan mengurutkan data sesuai dengan *preferensi*. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan preferensi, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.

****

**Gambar 4.22 Tampilan *Menu* Judul TA *By* Tahun Ajuan**

1. Pembimbing

Halaman Pembimbing adalah tempat di mana *admin* prodi dapat melihat semua pembimbing yang telah dipilih oleh kaprodi. Di sini, *admin* prodi dapat dengan mudah melihat jumlah bimbingan pada setiap pembimbing dan mengurutkan data sesuai dengan preferensi. Dalam menu ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan preferensi, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.23 Tampilan *Menu* Pembimbing**

1. Report

Halaman *Menu Report* memiliki 2 *sub menu* seperti Laporan Judul Diterima dan Laporan Judul Ditolak digunakan sebagai tempat dimana *admin* prodi dapat melihat semua informasi laporan data yang terkait dengan judul tugas akhir. Berikut *sub menu* yang terdapat dalam *menu report*:

1. Laporan Judul Diterima

Halaman *Menu Report* Data Judul Diterima adalah tempat di mana *admin* prodi dapat melihat laporan data yang terkait dengan judul tugas akhir. Di sini, *admin* prodi dapat melihat tabel data judul yang diterima yang akan membantu *admin* prodi mengetahui data judul tugas akhir mahasiswa. *Menu* ini dibuat untuk membantu *admin* prodi membuat keputusan yang lebih baik dan memperbaiki permasalahan sebelumnya dan mengurutkan data sesuai dengan preferensi. Dalam menu ini, *admin* juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.24 Tampilan *Menu* Report Judul Diterima**

1. Laporan Judul Ditolak

Halaman *Menu Report* Data Judul Ditolak adalah tempat di mana *admin* prodi dapat melihat laporan data yang terkait dengan judul tugas akhir. Di sini, *admin* prodi dapat melihat tabel data judul yang ditolak yang akan membantu *admin* prodi mengetahui data judul tugas akhir mahasiswa. *Menu* ini dibuat untuk membantu *admin* prodi membuat keputusan yang lebih baik dan memperbaiki permasalahan sebelumnya dan mengurutkan data sesuai dengan *preferensi*. Dalam *menu* ini, *admin* prodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga *admin* prodi dapat menggunakan data tersebut di luar sistem. Selain itu, *admin* prodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



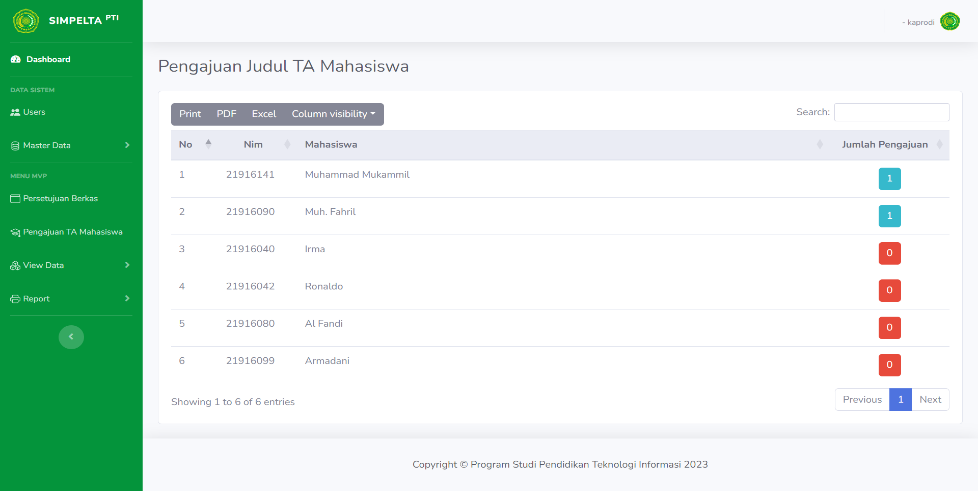
**Gambar 4.25 Tampilan *Menu* Report Judul Ditolak**

1. **Halaman Kaprodi**

Halaman kaprodi sendiri mempunyai *menu* yang sama dengan *menu* yang dimiliki oleh admin prodi, perbedaannya terdapat pada *menu* tambahan untuk kaprodi yaitu menu pengajuan judul TA. Berikut tampilan menu pengajuan judul TA:

1. Pengajuan Judul TA

Halaman Menu Pengajuan Judul TA sebagai tempat untuk menyimpan judul yang diajukan oleh mahasiswa. Di sini, kaprodi dapat melihat jumlah pengajuan judul pada setiap mahasiswa. Selain itu, Kaprodi dapat menyetujui/menolak judul dan memilih pembimbing satu dan pembimbing dua yang akan diberikan oleh mahasiswa serta memberikan catatan pada Setiap judul yang diajukan oleh mahasiswa. Dengan akses mudah ke menu pengajuan judul TA, kaprodi dapat membuat keputusan yang lebih baik dan lebih cepat, serta mengoptimalkan dalam manajemen pengajuan judul TA sebelumnya. Selain itu, kaprodi dapat menjaga pengajuan judul TA mahasiswa tetap teratur dan terorganisir dengan baik. Dalam menu ini, kaprodi juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga kaprodi dapat menggunakan data tersebut di luar sistem. Selain itu, kaprodi juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.26 Tampilan *Menu* Pengajuan Judul TA**

1. **Halaman Dosen**
2. Halaman Beranda

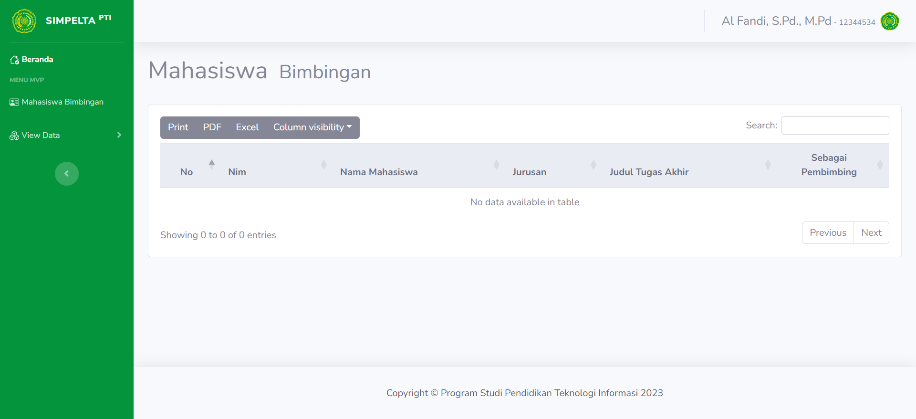
Halaman beranda *dosen* ini menampilkan informasi nama dan nomor induk dosen nasional.



**Gambar 4.27 Tampilan *Menu* Beranda**

1. Mahasiswa Bimbingan

*Menu* mahasiswa bimbingan adalah tempat di mana dosen dapat melihat semua data mahasiswa bimbingan yang telah diberikan oleh kaprodi. Di sini, dosen dapat dengan mudah mengetahui mahasiswa bimbingannya. Dalam menu ini, dosen juga dapat mengekspor data dalam format yang berbeda seperti PDF, Excel, serta print sehingga dosen dapat menggunakan data tersebut di luar sistem. Selain itu, dosen juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



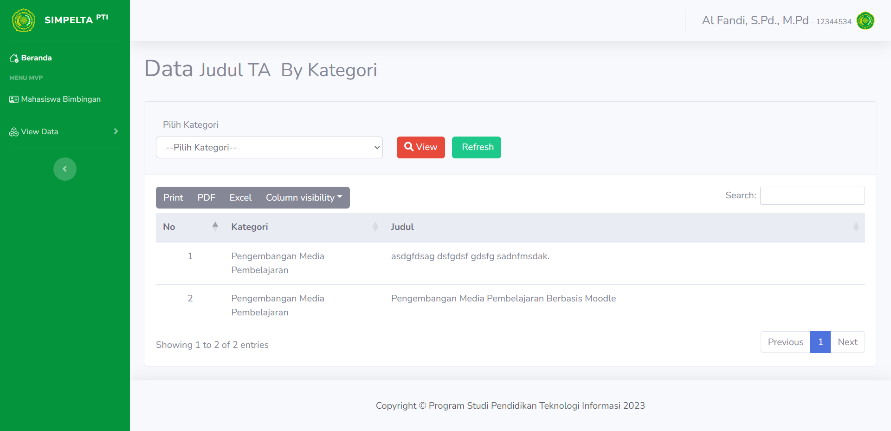
**Gambar 4.28 Tampilan *Menu* Mahasiswa Bimbingan**

1. View Data

Halaman *Menu View* Data memiliki 2 submenu seperti Judul TA *By* Kategori, Judul TA *By* Tahun Ajuan, dan Pembimbing digunakan sebagai tempat di mana dosen dapat melihat semua informasi judul tugas akhir mahasiswa. Berikut submenu yang terdapat dalam menu view data:

1. Judul TA by Kategori

Halaman Judul TA by Kategori adalah tempat di mana dosen dapat melihat semua data judul yang telah disetujui oleh kaprodi. Di sini, dosen dapat dengan mudah mencari data judul tugas akhir yang dibutuhkan, filter data berdasarkan kategori dan mengurutkan data sesuai dengan *preferensi*. Dalam menu ini, dosen juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga dosen dapat menggunakan data tersebut di luar sistem. Selain itu, dosen juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.29 Tampilan *Menu* Judul TA *By* Kategori**

1. Judul TA by Tahun Ajuan

Halaman Judul TA by Tahun Ajuan adalah tempat di mana dosen dapat melihat semua data judul yang telah disetujui oleh kaprodi. Di sini, dosen dapat dengan mudah mencari data judul tugas akhir yang dibutuhkan, filter data berdasarkan tahun ajuan dan mengurutkan data sesuai dengan preferensi. Dalam menu ini, dosen juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga dosen dapat menggunakan data tersebut di luar sistem. Selain itu, dosen juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.30 Tampilan *Menu* Judul TA *By* Tahun Ajuan**

1. **Halaman Mahasiswa**
2. Halaman Beranda

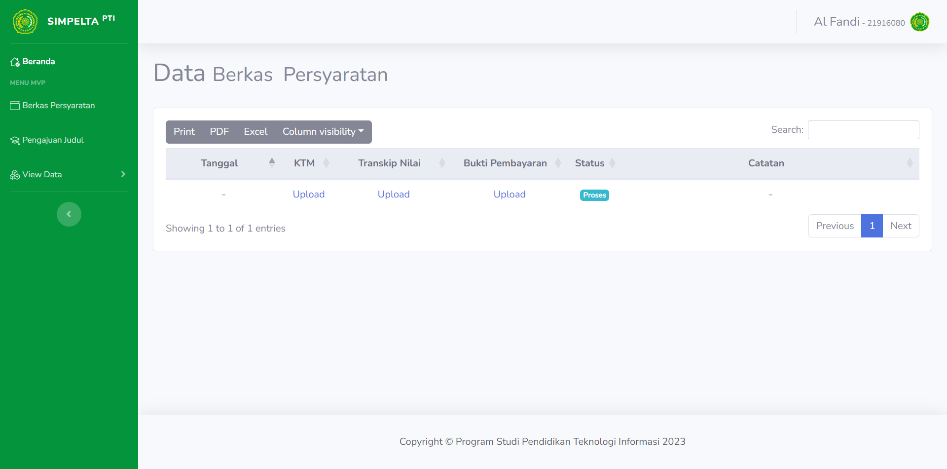
Halaman *beranda* mahasiswa ini menampilkan informasi nama mahasiswa dan nomor induk mahasiswa.



**Gambar 4.31 Tampilan *Menu* Beranda Mahasiswa**

1. Persyaratan Berkas

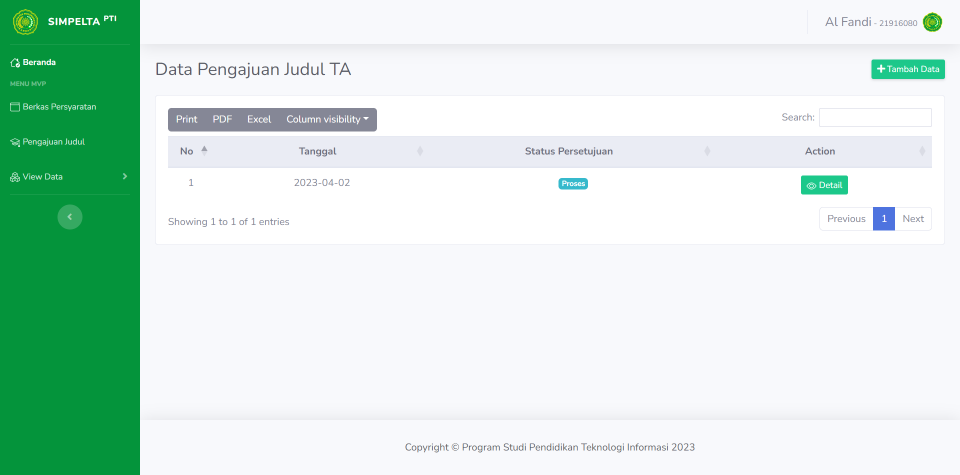
*Menu* Persyaratan Berkas adalah tempat di mana mahasiswa dapat mengajukan berkas persyaratan seperti kartu tanda mahasiswa, transkrip nilai dan bukti pembayaran. Di sini, mahasiswa dapat dengan mudah melihat apakah kaprodi sudah menyetujui berkas persyaratannya serta melihat catatan. Dalam menu ini, mahasiswa juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga mahasiswa dapat menggunakan data tersebut di luar sistem. Selain itu, mahasiswa juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.32 Tampilan *Menu* Persyaratan Berkas**

1. Pengajuan Judul TA

*Menu* Pengajuan Judul TA adalah tempat di mana mahasiswa dapat mengajukan judul TA dengan jumlah 3 judul untuk satu kali pengajuan. Di sini, mahasiswa dapat dengan mudah menambah data pengajuan judul TA dan mengurutkan data sesuai dengan preferensi. Menu ini dibuat untuk membantu mahasiswa mengajukan judul TA dengan jelas dan cepat. Dalam *menu* ini, mahasiswa juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga mahasiswa dapat menggunakan data tersebut di luar sistem. Selain itu, mahasiswa juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



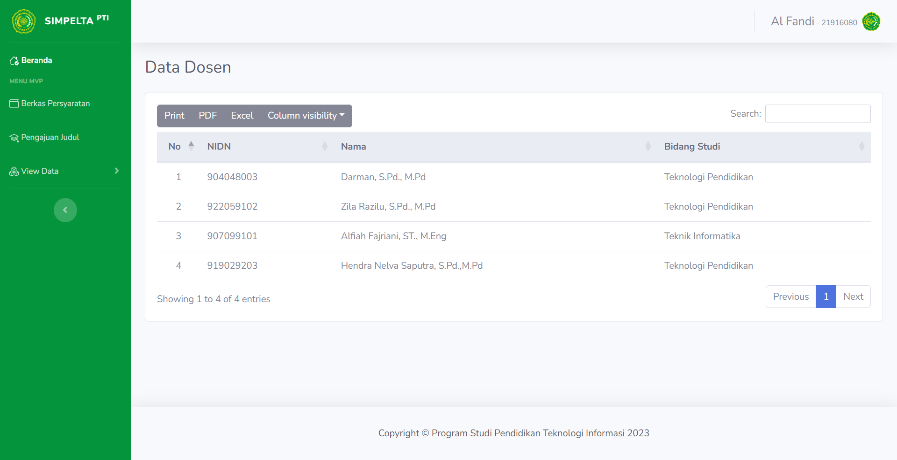
**Gambar 4.33 Tampilan *Menu* Pengajuan Judul TA**

1. View Data

Halaman Menu View Data memiliki 3 submenu seperti dosen, Judul TA By Kategori dan Judul TA *By* Tahun Ajuan. digunakan sebagai tempat di mana *mahasiswa* dapat melihat semua informasi judul tugas akhir mahasiswa serta daftar dosen. Berikut submenu yang terdapat dalam *menu view* data:

1. Dosen

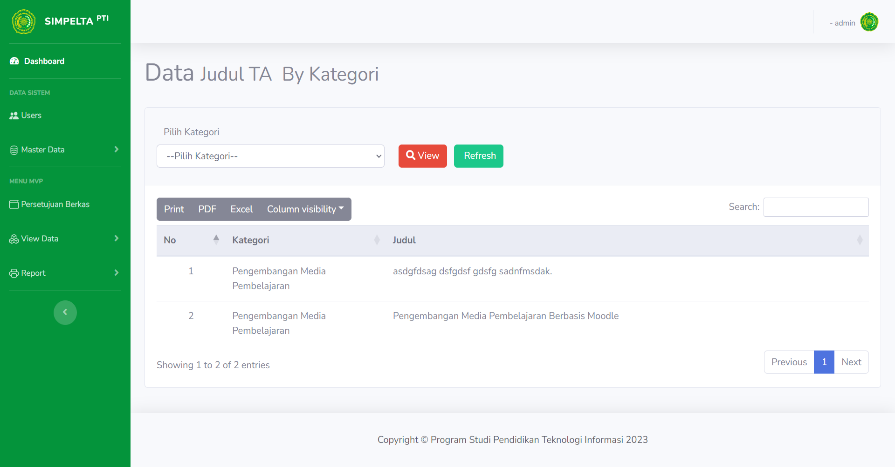
Halaman Dosen adalah tempat di mana mahasiswa dapat melihat semua data dosen yang telah ditambahkan oleh admin. Di sini, mahasiswa dapat dengan mudah melihat dosen yang ada dijadikan sebagai pembimbing sesuai dengan bidang studi dosen tersebut dan m­engurutkan data sesuai dengan preferensi. Dalam menu ini, mahasiswa juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga mahasiswa dapat menggunakan data tersebut di luar sistem. Selain itu, mahasiswa juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.34 Tampilan *Menu View* Dosen**

1. Judul TA by Kategori

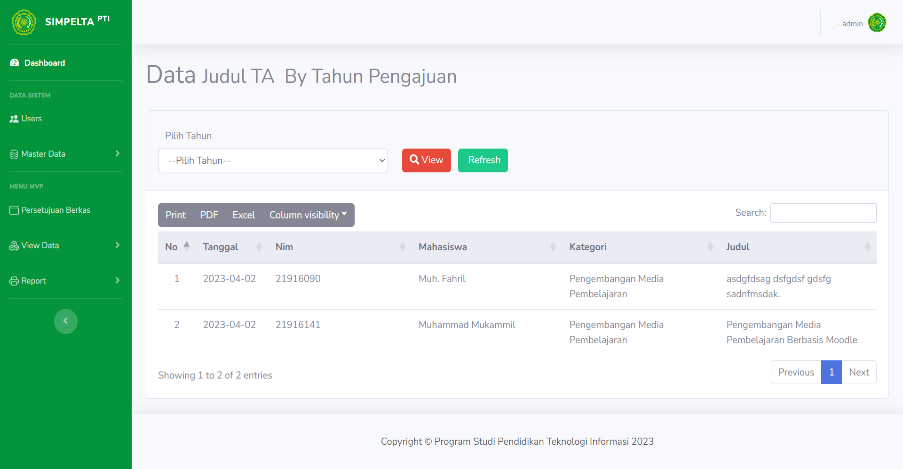
Halaman Judul TA by Kategori adalah tempat di mana mahasiswa dapat melihat semua data judul yang telah disetujui oleh kaprodi. Di sini, mahasiswa dapat dengan mudah mencari data judul tugas akhir yang dibutuhkan, *filter* data berdasarkan kategori dan mengurutkan data sesuai dengan preferensi. Dalam menu ini, mahasiswa juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga mahasiswa dapat menggunakan data tersebut di luar sistem. Selain itu, mahasiswa juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.35 Tampilan *Menu* Judul TA By Kategori**

1. Judul TA by Tahun Ajuan

Halaman Judul TA by Tahun Ajuan adalah tempat di mana mahasiswa dapat melihat semua data judul yang telah disetujui oleh kaprodi. Di sini, mahasiswa dapat dengan mudah mencari data judul tugas akhir yang dibutuhkan, *filter* data berdasarkan tahun ajuan dan mengurutkan data sesuai dengan preferensi. Dalam menu ini, mahasiswa juga dapat mengekspor data dalam format yang berbeda seperti *PDF*, *Excel*, serta *print* sehingga mahasiswa dapat menggunakan data tersebut di luar sistem. Selain itu, mahasiswa juga dapat menyesuaikan tampilan data sesuai dengan *preferensi*, seperti menampilkan atau menyembunyikan kolom tertentu dengan mengklik *column visibility*.



**Gambar 4.36 Tampilan *Menu* Judul TA By Tahun Ajuan**

1. **Pengujian Sistem**

Pengujian sistem informasi pengajuan judul tugas akhir di program studi pendidikan teknologi informasi(SIMPELTA PTI) ini menggunakan pengujian *black box.* Pengujian SIMPELTA PTI dilakukan pada Setiap fitur, terdapat 18 fitur yang ada didalam SIMPELTA PTI seperti *Fitur Login*, *Dashboard*/*Beranda*, *User's*, Data Dosen, Data Mahasiswa, Data Kategori, Data Bidang Studi, Data Jurusan, *View* Data Judul By Kategori, *View* Data Judul *By* Kategori, *View* Data Pembimbing, *Report* Judul yang diterima, *Report* Judul yang ditolak, Persetujuan Berkas, Persetujuan Judul TA, Mahasiswa Bimbingan, Pengajuan Judul TA, Pengajuan Berkas Persyaratan.

Berikut tampilan *table* testing yang digunakan:

1. Fitur Login

**Tebel 4.1 Uji *Black Box* Testing Fitur Login**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| *Login* | Menguji *Form Login* | Tidak mengisi *Username* dan *Password* | *Klik* Tombol *Login* | Sistem akan menolak dan menampilkan pesan "*The Username field is required*." | sesuai |  |
| Mengisikan *username*, dan tidak mengisi *password* | *Klik* Tombol *Login* | Sistem akan menolak dan menampilkan pesan "*The Username field is required*." | sesuai |  |
| Mengisikan *password*, dan tidak mengisi *username* | *Klik* Tombol *Login* | Sistem akan menolak dan menampilkan pesan "Nama *User* atau *Password* Tidak Sesuai !" | sesuai |  |
| Mengisikan *username* yang tidak sesuai (salah), dan mengisi *password* | *Klik* Tombol *Login* | Sistem akan menolak dan menampilkan pesan "*This username or password is wrong!*" | sesuai |  |
| Mengisikan password yang tidak sesuai (salah), dan mengisi username | *Klik* Tombol *Login* | Sistem akan menolak dan menampilkan pesan "*This username or password is wrong*!" | sesuai |  |
| Mengisikan username dan password tidak sesuai (salah) | *Klik* Tombol *Login* | Sistem akan menolak dan menampilkan pesan "*This username or password is wrong!*" | sesuai |  |
| mengisi username, pasword dengan benar | *Klik* Tombol *Login* | Sistem akan menerima dan menampilkan halaman *dashboard* | sesuai |  |

1. Fitur Dashboard/Beranda

**Tebel 4.2 Uji *Black Box* Testing Fitur Dashboard/Beranda**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| *Dashboard* | Menguji *menu Dashboard* | Melihat isi *menu Dashboard* | *Klik menu Dashboard* | Sistem Menampilkan Halaman informasi Total Data Mahasiswa, Total data Jurusan, Total Kategori Judul TA, Jumlah Pengajuan Judul dan tabel pengajuan judul yang sedang di proses | sesuai |  |
| *Beranda* | Menguji *menu Beranda* | Melihat isi *menu Beranda* | *Klik menu Beranda* | Sistem Menampilkan Halaman informasi Akun nama dan NIDN Dosen, NIM Mahasiswa | sesuai |  |

1. Fitur User

**Tebel 4.3 Uji *Black Box* Testing Fitur User**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data *User* | Menguji *form* pada *menu setting* Data *User* | Melihat isi menu *User* | *Klik menu* master data dosen | Sistem Menampilkan Halaman data *user* | sesuai |  |
| *Setting* data *User* | *Klik menu user*, *Klik* tombol *setting* | Sistem akan menampilkan halaman *setting* data *user* | sesuai |  |
| Tidak Mengisi **Password** pada form pengisian data User, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Password field is required*." | sesuai |  |
| Tidak Mengisi ***Konfirmasi Password***pada *form* pengisian data *User*, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "The Konfirmasi Password field is required." | sesuai |  |
| Tidak Mengisi semua data pada *form* pengisian data *User*, | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan"Bidang isian wajib di isi" pada kolom yang berbintang merah | sesuai |  |
| Mengisikan semua data pada *form* pengisian data *User* | *Klik* Tombol *Save* | Sistem akan menampilkan halaman data User dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman data *User* | sesuai |  |
| Menguji tombol *export Print*, *PDF* dan *Excel* serta *Column visibility* pada Data *User* | *Mengexport Print* Data *User* | *Klik menu* data dosen, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| Mengexport *PDF* Data *User* | *Klik* menu data dosen, Klik tombol PDF | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| Mengexport *PDF* Data *User* | *Klik menu* data dosen, *Klik* tombol *Excel* | Sistem akan mendownload dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom yang dipilih pada tampilan | sesuai |  |
|  | Menghapus data *User* | *Klik* tombol hapus | Sistem akan menampilkan notif "anda yakin? untuk menghapus data" | sesuai |  |
| *Klik* OK | Data akan terhapus | sesuai |  |
| *Klik Cancel* | Data batal terhapus dan kembali menampilkan halaman data *User* | sesuai |  |

1. Fitur Data Dosen

**Tebel 4.3 Uji *Black Box* Testing Fitur Dosen**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Dosen | Menguji *form* pada *menu* Tambah Data Dosen | Melihat isi *menu* master data dosen | *Klik menu* master data dosen | Sistem Menampilkan Halaman master data dosen | sesuai |  |
| Menambah data dosen | *Klik menu* master data dosen, *Klik* tombol tambah | Sistem akan menampilkan halaman tambah data dosen | sesuai |  |
| Tidak Mengisi **Nidn** pada form pengisian data dosen, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The NIDN field is required*." | sesuai |  |
| Tidak Mengisi **Nama Dosen** pada form pengisian data dosen, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The* NAMA DOSEN *field is required*." | sesuai |  |
| Tidak Memasukkan Pilihan **Bidang Studi** pada *form* pengisian data pribadi, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Bidang Studi field is required.*" | sesuai |  |
| Tidak Mengisi ***Username***pada *form* pengisian data dosen, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Username field is required*." | sesuai |  |
| Tidak Mengisi ***Password***pada *form* pengisian data dosen, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Username field is required*." | sesuai |  |
| Tidak Mengisi semua data pada *form* pengisian data dosen, dan mengisi data lainnya | *Klik* Tombol *Simpan* | Sistem akan menolak dan menampilkan pesan"Bidang isian wajib di isi" pada kolom yang berbintang merah | sesuai |  |
| Mengisikan semua data pada *form* pengisian data dosen | *Klik* Tombol Simpan | Sistem akan menampilkan halaman dosen dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Mengosongkan *form* pengisian | *Klik* tombol reset | Maka data pada kolom isian *form* akan kosong | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman data Dosen | sesuai |  |
| Menguji tombol export Print, PDF dan Excel serta Column visibility pada Master Data Dosen | *Mengexport Print* Data Dosen | *Klik* menu data dosen, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Data Dosen | *Klik* menu data dosen, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| Mengexport PDF Data Dosen | Klik menu data dosen, Klik tombol Excel | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom yang dipilih pada tampilan | sesuai |  |
|  | Menghapus data dosen | *Klik* tombol hapus | Sistem akan menampilkan *notif* "anda yakin? untuk menghapus data" | sesuai |  |
| *Klik* OK | Data akan terhapus | sesuai |  |
| *Klik Cancel* | Data batal terhapus dan kembali menampilkan halaman data dosen | sesuai |  |
| Mengedit data Dosen | *Klik* tombol edit | Sistem akan menampilkan halaman Edit data dosen | sesuai |  |
| Mengganti salah satu data Dosen | *Klik* tombol simpan | Sistem akan menampilkan halaman data dosen dan menampilkan pesan " Berhasil! Data berhasil diubah" | sesuai |  |

1. Fitur Data Mahasiswa

**Tebel 4.3 Uji *Black Box* Testing Fitur Mahasiswa**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Mahasiswa | Menguji form pada menu Tambah Data Mahasiswa | Melihat isi *menu* master data mahasiswa | *Klik* menu master data mahasiswa | Sistem Menampilkan Halaman master data mahasiswa | sesuai |  |
| Menambah data mahasiswa | *Klik menu* master data mahasiswa, *Klik* tombol tambah | Sistem akan menampilkan halaman tambah data mahasiswa | sesuai |  |
| Tidak Mengisi **Nim** pada form pengisian data mahasiswa, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Nim field is required*." | sesuai |  |
| Tidak Mengisi **Nama Mahasiswa** pada *form* pengisian data mahasiswa, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The NAMA MAHASISWA field is required.*" | sesuai |  |
| Tidak Memasukkan Pilihan **Jurusan** pada *form* pengisian data mahasiswa, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The JURUSAN field is required*." | sesuai |  |
| Tidak Mengisi ***Username***pada *form* pengisian data mahasiswa, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Username field is required*." | sesuai |  |
| Tidak Mengisi ***Password***pada *form* pengisian data mahasiswa, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Password field is required*." | sesuai |  |
| Tidak Mengisi semua data pada *form* pengisian data mahasiswa, dan mengisi data lainnya | *Klik* Tombol Simpan | Sistem akan menolak dan menampilkan pesan"Bidang isian wajib di isi" pada kolom yang berbintang merah | sesuai |  |
| Mengisikan semua data pada form pengisian data mahasiswa | *Klik* Tombol Simpan | Sistem akan menampilkan halaman Mahasiswa dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Mengosongkan form pengisian | *Klik* tombol *reset* | Maka data pada kolom isian form akan kosong | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman Master Data Mahasiswa | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada Master Data Mahasiswa | *Mengexport Print* Data Mahasiswa | *Klik menu* data mahasiswa, *Klik* tombol *Print* | Sistem akan menampilkan halaman print out | sesuai |  |
| *Mengexport PDF* Data Mahasiswa | *Klik* menu data mahasiswa, *Klik* tombol *PDF* | Sistem akan mendownload dengan format file.pdf | sesuai |  |
| *Mengexport PDF* Data Mahasiswa | Klik menu data mahasiswa, Klik tombol Excel | Sistem akan mendownload dengan format file.xlsx | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom yang dipilih pada tampilan | sesuai |  |
|  | Menghapus data Mahasiswa | *Klik* tombol hapus | Sistem akan menampilkan notif "anda yakin? untuk menghapus data" | sesuai |  |
| *Klik* OK | Data akan terhapus | sesuai |  |
| *Klik Cancel* | Data batal terhapus dan kembali menampilkan halaman data Mahasiswa | sesuai |  |
| Mengedit data Mahasiswa | *Klik* tombol edit | Sistem akan menampilkan halaman Edit data mahasiswa | sesuai |  |
| Mengganti salah satu data Mahasiswa | *Klik* tombol simpan | Sistem akan menampilkan halaman data Mahasiswa dan menampilkan pesan " Berhasil! Data berhasil diubah" | sesuai |  |

1. Fitur Data Kategori

**Tebel 4.3 Uji *Black Box* Testing Fitur Kategori**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Kategori | Menguji form pada menu Tambah Data Kategori | Melihat isi menu master data Kategori | *Klik menu* master data kategori | Sistem Menampilkan Halaman master data kategori | sesuai |  |
| Menambah data Kategori | *Klik men*u master data kategori, *Klik* tombol tambah | Sistem akan menampilkan halaman tambah data kategori | sesuai |  |
| Tidak Mengisi **Nama Kategori** pada *form* pengisian data kategori | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Nama Kategori field is required*." | sesuai |  |
| Mengisikan semua data pada *form* pengisian data kategori | *Klik* Tombol Simpan | Sistem akan menampilkan halaman Kategori dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Mengosongkan *form* pengisian | *Klik* tombol *reset* | Maka data pada kolom isian *form* akan kosong | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman Master Data Kategori | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada Master Data Kategori | *Mengexport Print* Data Kategori | *Klik men*u data kategori, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Data Kategori | *Klik menu* data kategori, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| Mengexport PDF Data Kategori | *Klik menu* data kategori, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
|  | Menghapus data kategori | *Klik* tombol hapus | Sistem akan menampilkan *notif* "anda yakin? untuk menghapus data" | sesuai |  |
| *Klik* OK | Data akan terhapus | sesuai |  |
| *Klik Cancel* | Data batal terhapus dan kembali menampilkan halaman data kategori | sesuai |  |
| Mengedit data kategori | *Klik* tombol *edit* | Sistem akan menampilkan halaman *Edit* data kategori | sesuai |  |
| Mengganti salah satu data kategori | *Klik* tombol simpan | Sistem akan menampilkan halaman data kategori dan menampilkan pesan " Berhasil! Data berhasil diubah" | sesuai |  |

1. Fitur Data Bidang Studi

**Tebel 4.3 Uji *Black Box* Testing Fitur Bidang Studi**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Bidang Studi | Menguji *form* pada *menu* Tambah Data BIdang Studi | Melihat isi menu master data Bidang Studi | *Klik menu* master data Bidang Studi | Sistem Menampilkan Halaman master data bidang studi | sesuai |  |
| Menambah data Bidang Studi | *Klik menu* master data Bidang Studi, *Klik* tombol tambah | Sistem akan menampilkan halaman tambah data bidang studi | sesuai |  |
| Tidak Mengisi **Nama Bidang Studi** pada *form* pengisian data Bidang Studi | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Nama Bidang Studi field is required*." | sesuai |  |
| Mengisikan semua data pada *form* pengisian data Bidang Studi | *Klik* Tombol Simpan | Sistem akan menampilkan halaman Bidang Studi dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Mengosongkan *form* pengisian | *Klik* tombol *reset* | Maka data pada kolom isian form akan kosong | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman Master Data Bidang Studi | sesuai |  |
| Menguji tombol export Print, PDF dan Excel serta Column visibility pada Master Data Bidang Studi | *Mengexport Print* Data Bidang Studi | *Klik menu* data kategori, *Klik* tombol *Print* | Sistem akan menampilkan halaman print out | sesuai |  |
| *Mengexport PDF* Data Bidang Studi | *Klik men*u data kategori, *Klik* tombol *PDF* | Sistem akan mendownload dengan format file.pdf | sesuai |  |
| *Mengexport PDF* Data Bidang Studi | *Klik menu* data kategori, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility,* pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
|  | Menghapus data Bidang Studi | *Klik* tombol hapus | Sistem akan menampilkan *notif* "anda yakin? untuk menghapus data" | sesuai |  |
| *Klik* OK | Data akan terhapus | sesuai |  |
| *Klik Cancel* | Data batal terhapus dan kembali menampilkan halaman data Bidang Studi | sesuai |  |
| Mengedit data Bidang Studi | *Klik* tombol *edit* | Sistem akan menampilkan halaman Edit data Bidang Studi | sesuai |  |
| Mengganti salah satu data Bidang Studi | *Klik* tombol simpan | Sistem akan menampilkan halaman data Bidang Studi dan menampilkan pesan " Berhasil! Data berhasil diubah" | sesuai |  |

1. Fitur Data Jurusan

**Tebel 4.3 Uji *Black Box* Testing Fitur Jurusan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Jurusan | Menguji *form* pada *menu* Tambah Data Jurusan | Melihat isi menu master data Jurusan | *Klik men*u master data Jurusan | Sistem Menampilkan Halaman master data Jurusan | sesuai |  |
| Menambah data Jurusan | *Klik men*u master data Jurusan, *Klik* tombol tambah | Sistem akan menampilkan halaman tambah data Jurusan | sesuai |  |
| Tidak Mengisi **Nama Jurusan** pada form pengisian data Jurusan | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Nama Jurusan field is required*." | sesuai |  |
| Mengisikan semua data pada form pengisian data Jurusan | *Klik* Tombol Simpan | Sistem akan menampilkan halaman Jurusan dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman Master Data Jurusan | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada Master Data Jurusan | *Mengexport Prin*t Data Jurusan | *Klik* menu data Jurusan, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Data Jurusan | *Klik menu* data Jurusan, *Klik* tombol *PDF* | Sistem akan mendownload dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Data Jurusan | *Klik menu* data Jurusan, *Klik* tombol *Excel* | Sistem akan mendownload dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
|  | Menghapus data Jurusan | *Klik* tombol hapus | Sistem akan menampilkan notif "anda yakin? untuk menghapus data" | sesuai |  |
| Klik OK | Data akan terhapus | sesuai |  |
| *Klik Cancel* | Data batal terhapus dan kembali menampilkan halaman data Jurusan | sesuai |  |
| Mengedit data Jurusan | *Klik* tombol *edit* | Sistem akan menampilkan halaman Edit data Jurusan | sesuai |  |
| Mengganti salah satu data Jurusan | *Klik* tombol simpan | Sistem akan menampilkan halaman data Jurusan dan menampilkan pesan " Berhasil! Data berhasil diubah" | sesuai |  |

1. Fitur View Data Judul By Kategori

**Tebel 4.3 Uji *Black Box* Testing Fitur View Data Judul *By* Kategori**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Judul TA *By Category* | Menguji menu Judul TA by Kategori | Melihat isi menu judul TA *by* Kategori | *Klik menu* judul TA *by* kategori | Sistem Menampilkan Halaman data judul TA *by* kategori | sesuai |  |
| Tidak pilih kategori pada *form* pengisian data judul TA *by* kategori | *Klik* Tombol *View* | Sistem akan menampilkan halaman data judul TA *by* kategori dan tidak menampilkan list judul *by* kategori | sesuai |  |
| Pilih salah satu kategori pada form pengisian data judul TA *by* kategori | *Klik* Tombol *View* | Sistem akan menampilkan halaman data judul TA *by* kategori dan menampilkan list judul *by* kategori | sesuai |  |
| Mengosongkan isi tabel judul yang ada di halaman data judul TA *by* kategori | *klik menu* judul TA *by* kategori, *Klik* tombol *refresh* | Maka data pada kolom isian form akan kosong | sesuai |  |
| Menguji tombol export Print, PDF dan Excel serta Column visibility pada data judul TA by kategori | Mengexport *Print* Data Judul TA *By* Kategori | *Klik menu* data Judul TA *By* Kategori, Klik tombol Print | Sistem akan menampilkan halaman *print out* | sesuai |  |
| Mengexport PDF Data Judul TA By Kategori | *Klik menu* data Judul TA *By* Kategori, *Klik* tombol *PDF* | Sistem akan mendownload dengan format *file.pdf* | sesuai |  |
| Mengexport PDF Data Judul TA By Kategori | *Klik menu* data Judul TA *By* Kategori, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |

1. Fitur View Data Judul By Tahun Ajuan

**Tebel 4.3 Uji *Black Box* Testing Fitur View Data Judul *By* Tahun Ajuan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Data Judul TA *By* Tahun Ajuan | Menguji menu Judul TA *by* Tahun Ajuan | Melihat isi menu judul TA *by* Tahun Ajuan | *Klik menu* judul TA *by* Tahun Ajuan | Sistem Menampilkan Halaman data judul TA *by* Tahun Ajuan | sesuai |  |
| Tidak pilih tahun ajuan pada form pengisian data judul TA *by* Tahun Ajuan | *Klik* Tombol *View* | Sistem akan menampilkan halaman data judul TA *by* tahun ajuan dan tidak menampilkan *list* judul *by* tahun ajuan | sesuai |  |
| Pilih salah satu tahun ajuan pada form pengisian data judul TA *by* Tahun Ajuan | *Klik* Tombol *View* | Sistem akan menampilkan halaman data judul TA *by* tahun ajuan dan menampilkan *list* judul *by* tahun ajuan | sesuai |  |
| Mengosongkan isi tabel judul yang ada di halaman data judul TA *by* tahun ajuan | *klik menu* judul TA *by* Tahun Ajuan, *Klik* tombol *refresh* | Maka data pada kolom isian *form* akan kosong | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada data judul TA *by* tahun ajuan | *Mengexport Print* Data Judul TA *By* Tahun Ajuan | *Klik* menu data Judul TA *By* Tahun Ajuan, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Data Judul TA *By* Tahun Ajuan | *Klik menu* data Judul TA *By* Tahun Ajuan, *Klik* tombol *PDF* | Sistem akan mendownload dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Data Judul TA *By* Tahun Ajuan | *Klik menu* data Judul TA *By* Tahun Ajuan, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |

1. Fitur View Data Pembimbing

**Tebel 4.3 Uji *Black Box* Testing Fitur View Data Pembimbing**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Pembimbing | Menguji menu Pembimbing | Melihat isi *menu* Pembimbing | *Klik menu* Pembimbing | Sistem Menampilkan Halaman data jumlah pembimbing | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada data Jumlah Pembimbing | *Mengexport Print* Data Jumlah Pembimbing | *Klik menu* data Jumlah Pembimbing, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Data Jumlah Pembimbing | *Klik menu* data Jumlah Pembimbing, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Data Jumlah Pembimbing | *Klik menu* data Jumlah Pembimbing, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility,* pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |

1. Fitur Report Judul TA Diterima

**Tebel 4.3 Uji *Black Box* Testing Fitur Report Judul TA Diterima**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| *Report* Judul TA Diterima | Menguji *menu Report* Judul TA Diterima | Melihat isi *menu Report* Judul TA Diterima | *Klik menu Report* Judul TA Diterima | Sistem Menampilkan Halaman Daftar Judul TA diterima | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada Daftar Judul TA Diterima | *Mengexport Print* Daftar Judul TA diterima | *Klik menu* Report Judul TA Diterima, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Daftar Judul TA diterima | *Klik menu* *Report* Judul TA Diterima, *Klik* tombol *PDF* | Sistem akan mendownload dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Daftar Judul TA diterima | *Klik menu Report* Judul TA Diterima, *Klik* tombol *Excel* | Sistem akan mendownload dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |

1. Fitur Report Judul TA Ditolak

**Tebel 4.3 Uji *Black Box* Testing Fitur Report Judul TA Ditolak**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| *Report* Judul TA Ditolak | Menguji *menu* Report Judul TA Ditolak | Melihat isi *menu Report* Judul TA Ditolak | *Klik menu Report* Judul TA Ditolak | Sistem Menampilkan Halaman Daftar Judul TA Ditolak | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada Daftar Judul TA Ditolak | *Mengexport Print* Daftar Judul TA Ditolak | Klik menu Report Judul TA Ditolak, Klik tombol Print | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Daftar Judul TA Ditolak | *Klik menu Report* Judul TA Ditolak, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Daftar Judul TA Ditolak | *Klik menu Report* Judul TA Ditolak, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan export | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |

1. Fitur Persetujuan Berkas

**Tebel 4.3 Uji *Black Box* Testing Fitur Persetujuan Berkas**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Persetujuan Berkas | Menguji *preview* pada menu Persetujuan Berkas | Melihat isi *menu* persetujuan Berkas | *Klik menu* persetujuan berkas | Sistem Menampilkan Halaman persetujuan berkas | sesuai |  |
| Lihat **KTM** pada halaman persetujuan berkas | *Klik menu* persetujuan berkas, *Klik* tombol Lihat | Sistem akan menampilkan *preview* berkas **KTM** | sesuai |  |
| Lihat **Transkrip Nilai** pada halaman persetujuan berkas | *Klik menu* persetujuan berkas, *Klik* tombol Lihat | Sistem akan menampilkan *preview* berkas **Transkrip Nilai** | sesuai |  |
| Lihat **Bukti Pembayaran** pada halaman persetujuan berkas | *Klik menu* persetujuan berkas, Klik tombol Lihat | Sistem akan menampilkan *preview* berkas **Bukti Pembayaran** | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada *menu* Persetujuan Berkas | *Mengexport Print* Data Berkas Persyaratan | *Klik menu* data Jurusan, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| Mengexport PDF Data Jurusan | *Klik menu* data Jurusan, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Data Berkas Persyaratan | *Klik menu* data Jurusan, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
| *Menguji tombol Approve pada menu Persetujuan Berkas* | *Melihat kolom approval berkas persyaratan* | *Klik* tombol *approve* | Sistem akan menampilkan kolom *Approval* berkas persyaratan | sesuai |  |
| *Tidak Mengisi* ***Catatan Prodi*** *pada form pengisian approval berkas persyaratan* | *Klik menu* persetujuan berkas, *Klik* tombol *approval*, *Klik Submit* | Sistem akan menolak dan menampilkan "Data Approval Wajib Diisi." | sesuai |  |
| *Mengisikan semua data pada form approval berkas persyaratan* | *Klik* Tombol *Submit* | Sistem akan menampilkan halaman Data Berkas Persyaratan dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| *Klik Cancel* | | Data batal disetujui dan kembali menampilkan halaman data Berkas Persyaratan | sesuai |  |

1. Fitur Persetujuan Judul TA

**Tebel 4.3 Uji *Black Box* Testing Fitur Persetujuan Judul TA**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Pengajuan Judul TA Mahasiswa | Menguji menu pengajuan TA Mahasiswa | Melihat isi menu pengajuan TA mahasiswa | *Klik menu* pengajuan TA mahasiswa | Sistem Menampilkan Halaman pengajuan judul TA mahasiswa | sesuai |  |
| Menguji tombol jumlah pengajuan Mahasiswa | Melihat isi data persetujuan judul TA | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol jumlah pengajuan | Sistem akan menampilkan data persetujuan judul TA | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman Pengajuan Judul TA | sesuai |  |
| Menguji tombol detail pengajuan judul TA Mahasiswa | Melihat isi detail persetujuan judul TA | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol jumlah pengajuan, *Klik* tombol detail | Sistem akan menampilkan data detail persetujuan judul TA | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman Data Persetujuan Judul TA |  |  |
| Menguji tombol *approve* judul TA Mahasiswa | Melihat isi *approval* judul TA | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol jumlah pengajuan, *Klik* tombol *detail*, *Klik* tombol *approve* | Sistem akan menampilkan kolom approval judul TA | sesuai |  |
| Tidak Mengisi **Catatan Prodi** pada *form Approval* judul TA | *Klik* Tombol *Submit* | Sistem akan menolak dan menampilkan "Data Approve, Wajib Diisi." | sesuai |  |
| Tidak memilih/Mengisi **Dosen Pembimbing 1** pada *form Approval* judul TA | *Klik* Tombol *Submit* | Sistem akan menolak dan menampilkan "Data *Approve*, Wajib Diisi." | sesuai |  |
| Tidak memilih/Mengisi **Dosen Pembimbing 2** pada *form Approval* judul TA | *Klik* Tombol *Submit* | Sistem akan menolak dan menampilkan "Data *Approve*, Wajib Diisi." | sesuai |  |
| Tidak mengisikan semua data pada *form* pengisian *Approval* Judul TA | *Klik* Tombol *Submit* | Sistem akan menolak dan menampilkan "Data *Approve*, Wajib Diisi." | sesuai |  |
| Mengisikan semua data pada *form* pengisian *Approval* Judul TA | *Klik* Tombol *Submit* | Sistem akan menampilkan halaman *Detail* Persetujuan Judul TA dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| *Klik Cancel* | | Data batal disetujui dan kembali menampilkan halaman data Berkas Persyaratan | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada *menu* Persetujuan Berkas | *Mengexport Print* | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
| Menguji tombol Tolak Semua judul TA Mahasiswa | Melihat kolom pengajuan judul TA | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol jumlah pengajuan, *Klik* tombol detail, *Klik* tombol tolak semua | Sistem akan menampilkan kolom pengajuan judul TA | sesuai |  |
| Tidak Mengisi **Catatan Prodi** pada *form* pengisian pengajuan judul TA | *Klik menu* pengajuan TA mahasiswa, *Klik* tombol jumlah pengajuan, *Klik* tombol detail, *Klik* tombol tolak semua | Sistem akan menolak dan menampilkan "Catatan,l Wajib Diisi." | sesuai |  |
| Mengisikan semua data pada *form* pengajuan judul TA | *Klik* Tombol *Submit* | Sistem akan menampilkan halaman *Detail* Persetujuan Judul TA dan menampilkan pesan " Berhasil Ditolak" | sesuai |  |
| *Klik Cancel* | | Data batal ditolak dan kembali menampilkan halaman *Detail* Persetujuan Judul TA | sesuai |  |

1. Fitur Mahasiswa Bimbingan

**Tebel 4.3 Uji *Black Box* Testing Fitur Mahasiswa Bimbingan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Mahasiswa Bimbingan | Menguji *menu* Mahasiswa Bimbingan | Melihat isi *menu* Mahasiswa Bimbingan | *Klik menu* Mahasiswa Bimbingan | Sistem Menampilkan Halaman Mahasiswa Bimbingan | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada Mahasiswa Bimbingan | *Mengexport Print* Mahasiswa Bimbingan | *Klik menu* Mahasiswa Bimbingan, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Mahasiswa Bimbingan | *Klik menu* Mahasiswa Bimbingan, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Mahasiswa Bimbingan | *Klik men*u Mahasiswa Bimbingan, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |

1. Fitur Pengajuan Judul TA

**Tebel 4.3 Uji *Black Box* Testing Fitur Pengajuan Judul TA**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Pengajuan Judul TA | Menguji *form* pada menu Tambah Data Pengajuan Judul TA | Melihat isi menu pengajuan judul TA | *Klik menu* pengajuan judul TA | Sistem Menampilkan Halaman Data Pengajuan Judul TA | sesuai |  |
| Menambah data pengajuan judul TA | *Klik menu* pengajuan judul TA, *Klik* tombol tambah data | Sistem akan menampilkan halaman tambah data pengajuan judul TA | sesuai |  |
| Tidak memasukan pilihan **Kategori** pada *form* pengisian data pengajuan judul TA 1, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Kategori field is required*." | sesuai |  |
| Tidak Mengisi **Judul TA** pada *form* pengisian data pengajuan judul TA 1, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Judul TA field is required*." | sesuai |  |
| Tidak mengisi **Deskripsi** pada form pengisian data pengajuan judul TA 1, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Deskripsi field is required*." | sesuai |  |
| Tidak Mengisi semua data pada *form* pengisian data pengajuan judul TA 1, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan"Bidang isian wajib di isi" pada kolom yang berbintang merah | sesuai |  |
| Tidak memasukan pilihan **Kategori** pada *form* pengisian data pengajuan judul TA 2, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Kategori field is required*." | sesuai |  |
| Tidak Mengisi **Judul TA** pada *form* pengisian data pengajuan judul TA 2, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Judul TA field is required*." | sesuai |  |
| Tidak mengisi **Deskripsi** pada *form* pengisian data pengajuan judul TA 2, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Deskripsi field is required*." | sesuai |  |
| Tidak Mengisi semua data pada *form* pengisian data pengajuan judul TA 2, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan"Bidang isian wajib di isi" pada kolom yang berbintang merah | sesuai |  |
| Tidak memasukan pilihan **Kategori** pada *form* pengisian data pengajuan judul TA 3, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "*The Kategori field is required*." | sesuai |  |
| Tidak Mengisi **Judul TA** pada *form* pengisian data pengajuan judul TA 3, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Judul TA field is required.*" | sesuai |  |
| Tidak mengisi **Deskripsi** pada *form* pengisian data pengajuan judul TA 3, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "*The Deskripsi field is required*." | sesuai |  |
| Tidak Mengisi semua data pada *form* pengisian data pengajuan judul TA 3, dan mengisi data lainnya | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan"Bidang isian wajib di isi" pada kolom yang berbintang merah | sesuai |  |
| Mengisikan semua data pada *form* pengisian data pengajuan judul TA | *Klik* Tombol *Save* | Sistem akan menampilkan halaman dosen dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| Kembali ke halaman sebelumnya | *Klik* tombol *Back* | Sistem akan kembali ke halaman data pengajuan judul TA | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibility* pada *menu* Pengajuan Judul TA | *Mengexport Print* | *Klik menu* pengajuan Judul TA, *Klik* tombol *Print* | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* | *Klik menu* pengajuan Judul TA, *Klik* tombol *PDF* | Sistem akan mendownload dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* | *Klik menu* pengajuan Judul TA, *Klik* tombol *Excel* | Sistem akan mendownload dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
| Menguji aturan pada *form* tambah data pengajuan judul TA | Mengisikan 3 kata pada form pengisian judul TA 1 | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "Minimal 4 suku kata." | sesuai |  |
| Mengisikan **Judul TA** yang sudah ada didalam databasepada *form* pengisian Judul TA 1 | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "Judul sudah ada didalam *database*" | sesuai |  |
| Mengisikan **Judul TA** mirip dengan judul lain pada *form* pengajuan judul TA 1 | *Klik* Tombol *Save* | Sistem akan menampilkan halaman Tambah Data Pengajuan Judul TA dan menampilkan pesan " Judul yang anda ajukan mirip dengan judul yang diajukan mahasiswa lain, tetapi judul tetap terkirim." | sesuai |  |
| Mengisikan 3 kata pada *form* pengisian judul TA 2 | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "Minimal 4 suku kata." | sesuai |  |
| Mengisikan **Judul TA** yang sudah ada didalam databasepada *form* pengisian Judul TA 2 | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "Judul sudah ada didalam database" | sesuai |  |
| Mengisikan **Judul TA** mirip dengan judul lain pada *form* pengajuan judul TA 2 | *Klik* Tombol *Save* | Sistem akan menampilkan halaman Tambah Data Pengajuan Judul TA dan menampilkan pesan " Judul yang anda ajukan mirip dengan judul yang diajukan mahasiswa lain, tetapi judul tetap terkirim." | sesuai |  |
| Mengisikan 3 kata pada *form* pengisian judul TA 3 | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan pesan "Minimal 4 suku kata." | sesuai |  |
| Mengisikan **Judul TA** yang sudah ada didalam *database*pada *form* pengisian Judul TA 3 | *Klik* Tombol *Save* | Sistem akan menolak dan menampilkan "Judul sudah ada didalam *database*" | sesuai |  |
| Mengisikan **Judul TA** mirip dengan judul lain pada *form* pengajuan judul TA 3 | *Klik* Tombol *Save* | Sistem akan menampilkan halaman Tambah Data Pengajuan Judul TA dan menampilkan pesan " Judul yang anda ajukan mirip dengan judul yang diajukan mahasiswa lain, tetapi judul tetap terkirim." | sesuai |  |
| Menguji tombol *detail* pada menu pengajuan judul TA | Melihat isi *menu* pengajuan judul TA | *Klik menu* pengajuan judul TA | Sistem Menampilkan Halaman *Detail* Judul TA | sesuai |  |

1. Fitur Pengajuan Berkas Persyaratan

**Tebel 4.3 Uji *Black Box* *Testing* Fitur Pengajuan Berkas Persyaratan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **FITUR** | **STEP DESCRIPTION** | **TEST STEP** | **ACTION** | **EXPECTED RESULT** | **STATUS** | **NOTE** |
| Pengajuan Berkas Persyaratan | Menguji *upload* pada *menu* Pengajuan Berkas Persyaratan | Melihat isi menu Berkas Persyaratan | *Klik men*u Berkas Persyaratan | Sistem Menampilkan Halaman Berkas Persyaratan | sesuai |  |
| *Upload* **KTM** pada halaman berkas persyaratan | *Klik menu* berkas persyaratan, *Klik* tombol *upload* | Sistem akan menampilkan preview berkas **KTM** | sesuai |  |
| *Upload* **Transkrip Nilai** pada halaman berkas persyaratan | *Klik menu* berkas persyaratan, *Klik* tombol *upload* | Sistem akan menampilkan *preview* berkas **Transkrip Nilai** | sesuai |  |
| *Upload* **Bukti Pembayaran** pada halaman berkas persyaratan | *Klik menu* berkas persyaratan, *Klik* tombol *upload* | Sistem akan menampilkan *preview* berkas **Bukti Pembayaran** | sesuai |  |
| Mengisikkan *file* lebih dari 2 *mb*, pada Data berkas persyaratan | *Klik* Tombol *Upload* | Sistem akan menolak dan menampilkan pesan" Isian seharusnya tidak lebih dari 2 *mb*." | sesuai |  |
| Mengisikan yang bukan jenis file *pdf* | *Klik* Tombol *Upload* | Sistem akan menolak dan menampilkan pesan"Isian harus dokumen berjenis : *pdf*." | sesuai |  |
| *Klik Cancel* | | Data batal di *upload* dan kembali menampilkan halaman data Berkas Persyaratan | sesuai |  |
| Menguji tombol *export Print, PDF* dan *Excel* serta *Column visibilit*y pada menu Persetujuan Berkas | Mengexport Print Data Berkas Persyaratan | Klik menu data Jurusan, Klik tombol Print | Sistem akan menampilkan halaman *print out* | sesuai |  |
| *Mengexport PDF* Data Jurusan | *Klik menu* data Jurusan, *Klik* tombol *PDF* | Sistem akan *mendownload* dengan format *file.pdf* | sesuai |  |
| *Mengexport PDF* Data Berkas Persyaratan | *Klik menu* data Jurusan, *Klik* tombol *Excel* | Sistem akan *mendownload* dengan format *file.xlsx* | sesuai |  |
| Menyembunyikan salah satu kolom sebelum melakukan *export* | *Klik* Tombol *Column visibility*, pilih satu kolom | Sistem akan Menyembunyikan kolom pada tampilan | sesuai |  |
| Menguji tombol *Approve* pada *menu* Persetujuan Berkas | Melihat kolom *approval* berkas persyaratan | *Klik* tombol *approve* | Sistem akan menampilkan kolom *Approval* berkas persyaratan | sesuai |  |
| Tidak Mengisi **Catatan Prodi** pada *form* pengisian approval berkas persyaratan | *Klik menu* persetujuan berkas, *Klik* tombol *approval*, *Klik Submit* | Sistem akan menolak dan menampilkan "Data *Approval* Wajib Diisi." | sesuai |  |
| Menuliskan semua data pada form approval berkas persyaratan | *Klik* Tombol *Submit* | Sistem akan menampilkan halaman Data Berkas Persyaratan dan menampilkan pesan " Berhasil!Data disimpan" | sesuai |  |
| *Klik Cancel* | | Data batal disetujui dan kembali menampilkan halaman data Berkas Persyaratan | sesuai |  |

Dari hasil pengujian *black box* di atas, hasil pengujian *black box* dapat ditarik kesimpulan bahwa SIMPELTA PTI yang telah dirancang dan di bangun secara fungsional, fitur-fitur yang ada sudah sesuai dengan kebutuhan yang diharapkan dengan melihat *expected result* dengan jumlah keseluruhan 220 skenario yang telah di uji

**BAB V**

**PENUTUP**

1. **Kesimpulan**

Berdasarkan Implementasi dan Pengujian di atas dapat ditarik kesimpulan sebagai bahwa Sistem Informasi Pengajuan Judul Tugas Akhir Program Studi Pendidikan Teknologi Informasi (SIMPELTA PTI) Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari ini menggunakan MySQL sebagai database penyimpanan data. Database terdiri dari 10 tabel yang diantaranya adalah *tbl\_user, tbl\_roles, tbl\_lecturers, tbl\_students, tbl\_majors, tbl\_field\_of\_studys, tbl\_categorys, tbl\_essays, tbl\_titles, tbl\_files*. Sistem Informasi Pengajuan Judul Tugas Akhir Program Studi Pendidikan Teknologi Informasi (SIMPELTA PTI) Fakultas Keguruan dan Ilmu pendidikan Universitas Muhammadiyah Kendari ini terdiri dari halaman halaman admin prodi, halaman Kepala Program Studi (kaprodi), halaman dosen serta halaman mahasiswa. Halaman admin, halaman kaprodi, halaman dosen serta halaman mahasiswa memerlukan login untuk mengaksesnya. Pengujian dilakukan dengan menguji setiap proses dan kemungkinan kesalahan yang terjadi pada setiap proses. Hasil pengujian dengan menggunakkan metode *black box* testing, yaitu pengujian dilakukan dengan hanya memperhatikan masukan ke sistem dan keluaran ke sistem. Dari pengujian yang dilakukan telah berhasil dengan baik, bebas dari kesalahan sintak sehingga bisa dikatakan secara fungsional Sistem Informasi Pengajuan Judul Tugas Akhir Program Studi Pendidikan Teknologi Informasi (SIMPELTA PTI) ini telah berjalan dengan cukup baik dan sesuai dengan kebutuhan SIMPELTA PTI dan menghasilkan *output* sesuai dengan apa yang diharapkan.

1. **Saran**

Adapun saran yang dapat diberikan agar selanjutnya SIMPELTA PTI ini dapat dikembangkan lagi adalah sebagai berikut:

1. Perlunya penambahan fitur import file data dosen dan data mahasiswa agar pempermudah dalam prose tambah data dosen dan tambah data mahasiswa (tidak satu-satu).
2. Perlunya penambahan *ajax* pada diagram *dashoard* admin agar lebih terlihat *realtime* perubahan data yang terjadi.
3. Menggunakan algoritma NLP(Natural Language Processing) untuk mengecek judul yang sama dalam sistem pengajuan judul tugas akhir. Dengan menggunakan algoritma NLP, admin dapat melakukan pengecekan otomatis terhadap judul-judul yang diajukan dalam sistem pengajuan judul tugas akhir. Ini akan membantu dalam mengidentifikasi adanya judul-judul yang mirip atau kemungkinan duplikat, sehingga memudahkan dalam mengelola pengajuan tugas akhir dan mencegah kemungkinan plagiarisme atau duplikasi judul.

**DAFTAR PUSTAKA**

Abdullah, R. (2015). *Web Programming is Easy*. Jakarta: Elex Media Komputindo.

Agung, & Leo. (2012). *Aplikasi Pemrograman Javascript untuk Halaman Web*. Yogyakarta: Andi Offset.

Badiyanto. (2013). *Buku Pintar Framework Yii*. Yogyakarta: Penerbit Mediakom.

Badiyanto, & Murya, Y. (2018). *Project Php Membangun Sistem Informasi Akademik Dengan Framework Codeigniter*. Yogyakarta: Cv. Langit Inspirasi.

Bekti, B. H. (2015). *Mahir membuat website dengan adobe dreamweaver CS6, CSS dan jquery*. Yogyakarta: Andi.

Bentley, Lonnie D and Whitten Jeffrey L. (2007). *Systems Analysis and Design for the Global Enterprise Seventh Edition International Edition*. New York: McGrawHill.

Bima, H. (2021). Rancang Bangun Sistem Informasi Kelayakan Pemberian Kredit Mobil Menggunakan Metode Weighted Product Pada Rafi Motor. Universitas Darma Persada.

Bunafit Nugroho. (2008). *Aplikasi Pemrograman Web Dinamis Dengan PHP dan*

*MySQL*. Yogyakarta: Gava Media.

Dennis. (2021). *Perancangan Aplikasi Absensi Karyawan Dengan Menggunakan Kode Qr Berbasis Android, Skripsi Program Studi Teknik Informatika Fakultas Teknik Dan Komputer Universitas Putera*. Batam: Universitas Putera.

Efendi, T.F. (2017). *Pengembangan Website Smk Negeri 3 Sukoharjo*. In Seminar Nasional Sistem Informasi (SENASIF) (Vol. 1, No. 1, pp. 957- 964).

H. Alatas. (2013). *Responsive Web Design dengan PHP & Bootstrap*. Yogyakarta: Lokomedia.

Hakim, Lukmanul. (2010). *Bikin Website Super Keren Dengan PHP dan jQuery*.

Yogyakarta: Lokomedia.

Hermawan, J. (2017). *Analisis Dan Perancangan Sistem Informasi Pengolahan Data Rawat Jalan Berdasarkan Konsep Fast (Framework For The Application Of System Thinking) Berbasis Desktop Pada Poliklinik Gigi Puskesmas Lubuk Besar*. Kab. Bangka Tengah: Stmik Atma Luhur.

Humairah, H. (2020). Pengembangan Sistem Pengajuan Judul Skripsi Di Jurusan Pendidikan Teknik Elektro Universitas Negeri Makassar Berbasis Website. *Jurnal MediaTIK,* 3(3). <https://doi.org/10.26858/jmtik.v3i3.15180>

Jalal, M. (2012). Problematika Kesalahan Bahasa Pada Penulisan Skripsi Mahasiswa Universitas Airlangga. *Jurnal Ilmu Humaniora*, *12*(2), 92–104.

Jamasnia, E., Assidiq, M., & Khairat, U. (2021). Sistem Informasi Pengajuan Judul Skripsi Berbasis Web Service. J*ournal Perguruan: Conference Series,* 3(2). <https://doi.org/10.35329/jp.v3i2.2107>

L. Afuan. (2010). Pemanfaatan *Framework CodeIgniter* Dalam Pengembangan

Sistem Informasi Pendataan Laporan Kerja Praktek Mahasiswa Program Studi Teknik Informatika Unsoed (Codeigniter Framework Used in Information System Development for Student’s Report Data Collection Prac. *Juita*, I, 39–44.

Mulyani, S. (2017). Analisis dan Perancangan Sistem Informasi Manajemen Keuangan Daerah: Notasi Pemodelan Unified Modeling Language (UML). Abdi Sistematika. Studi Teknik Informatika Unsoed. *Juita*. I (2), 39–44.

Murtadho, M., & Sanjaya, C. B. (2019). Sistem Aplikasi Pendaftaran Skripsi Berbasis Mobile Di Universitas Yudharta Pasuruan Menggunakan Metodologi Unified Process. *JASIEK (Jurnal Aplikasi Sains, Informasi, Elektronika Dan Komputer)*, *1*(2), 156–160.

Punjul, T. J. S. (2016). *Sistem Informasi Manajemen*. Yogyakarta: Penerbit Deepublish.

Putu, A. (2011). *Sistem Informasi dan Implementasinya*. Bandung: Informatika.

Saro, D., Teknik, F., & Sina, U. I. (2021). Sistem Informasi Pengajuan Judul Skripsi Mahasiswa Program Studi Teknik Informatika (Studi Kasus Fakultas Teknik Universitas Ibnu Sina). *Engineering Technology International*, 3(2).

Satyahadewi, N. (2019). Sistem Informasi Monitoring Tugas Akhir (SIMTA) Berbasis Web Fakultas MIPA Universitas Tanjungpura. *CESS (Journal of Computer Engineering, System and Science)*, *4*(1), 83–87.

Sibero, A. F. K. (2013). *Web programming power pack*. Yogyakarta: Mediakom.

Sidik, B. (2012). *Framework CodeIgniter*. Bandung: Penerbit Informatika.

Solanki, N., Shah, D., & Shah, A. (2017). A Survey on different Frameworks of PHP. *International Journal of Latest Technology in Engineering, Management & Applied Science (IJLTEMAS*), 155-158.

Spurlock, Jake. (2013). *Bootstrap*. Sebastopol: O’Reilly. Beighley

Sugiarti, Yuni, S.T., M Kom. (2013). Analisis dan Perancangan UML (Unified Modelling Language) Generated VB 6. Yogyakarta: Graha Ilmu.

Upton, D. (2007). *CodeIgniter for Rapid PHP Application Development*. Birmingham: Packt Publishing Ltd.

Wardani, S. K. (2013). Sistem Informasi Pengolahan Data Nilai Siswa Berbasis Web Pada Sekolah Menengah Atas (SMA) Muhammadiyah Pacitan. *IJNS Indonesian Journal on Networking and Security*, *2*(2).

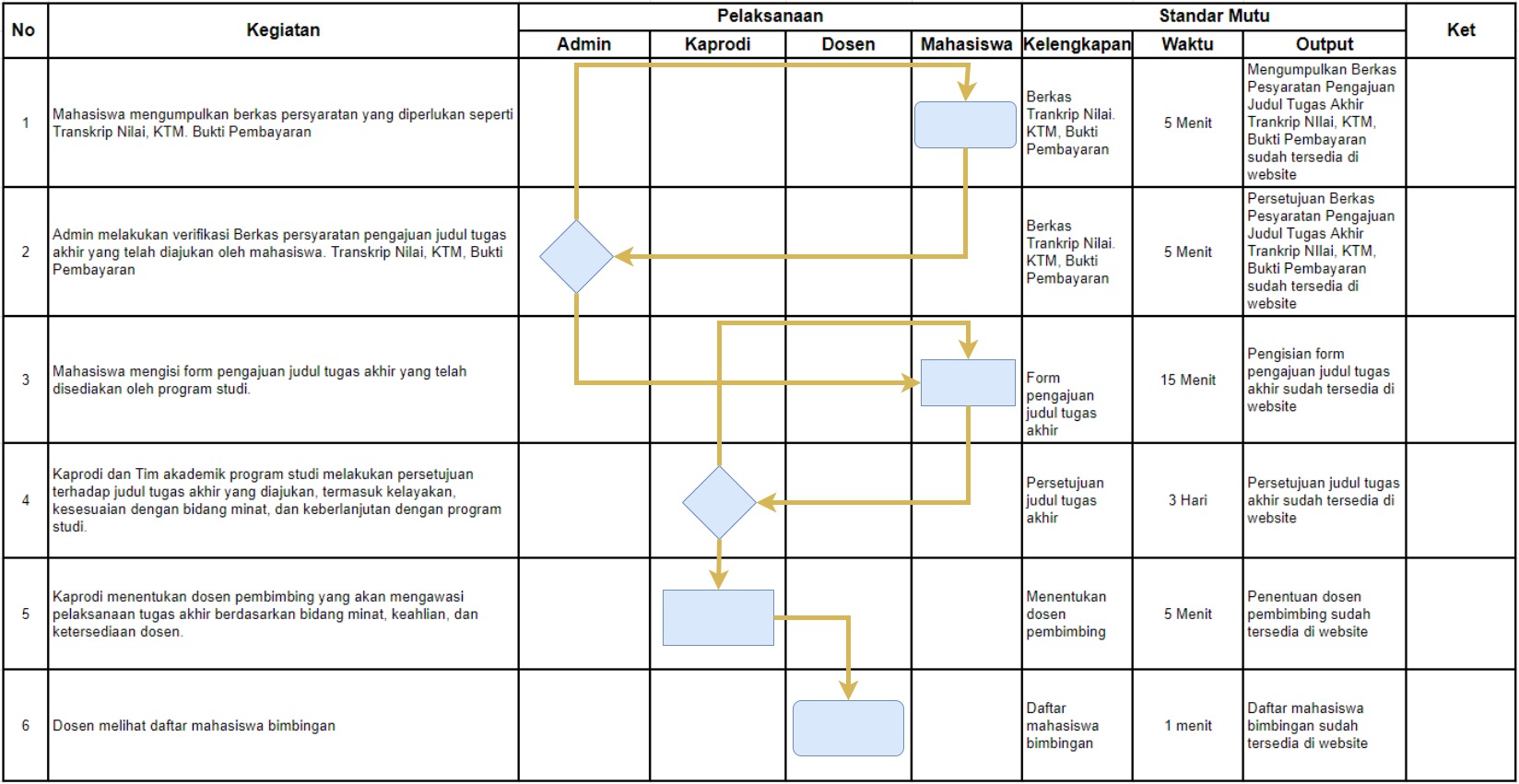
Widyaratna, T., Naseer, M., & Supriadi, I. (2017). Pengembangan Aplikasi Web Pengajuan Judul Skripsi Dengan Framework Codeigniter Dan Html5 Pada Sekolah Tinggi. Seminar Nasional Informatika Dan Sistem Informasi, 5, 200–221.

Zulkifli, Z. (2021). Sistem Informasi Pengajuan Judul Skripsi Secara Online Untuk Mencegah Penyebaran Virus Covid-19. *Jurnal Tika*, 6(02). https://doi.org/ 10.51179/tika. v6i02.478

**LAMPIRAN**

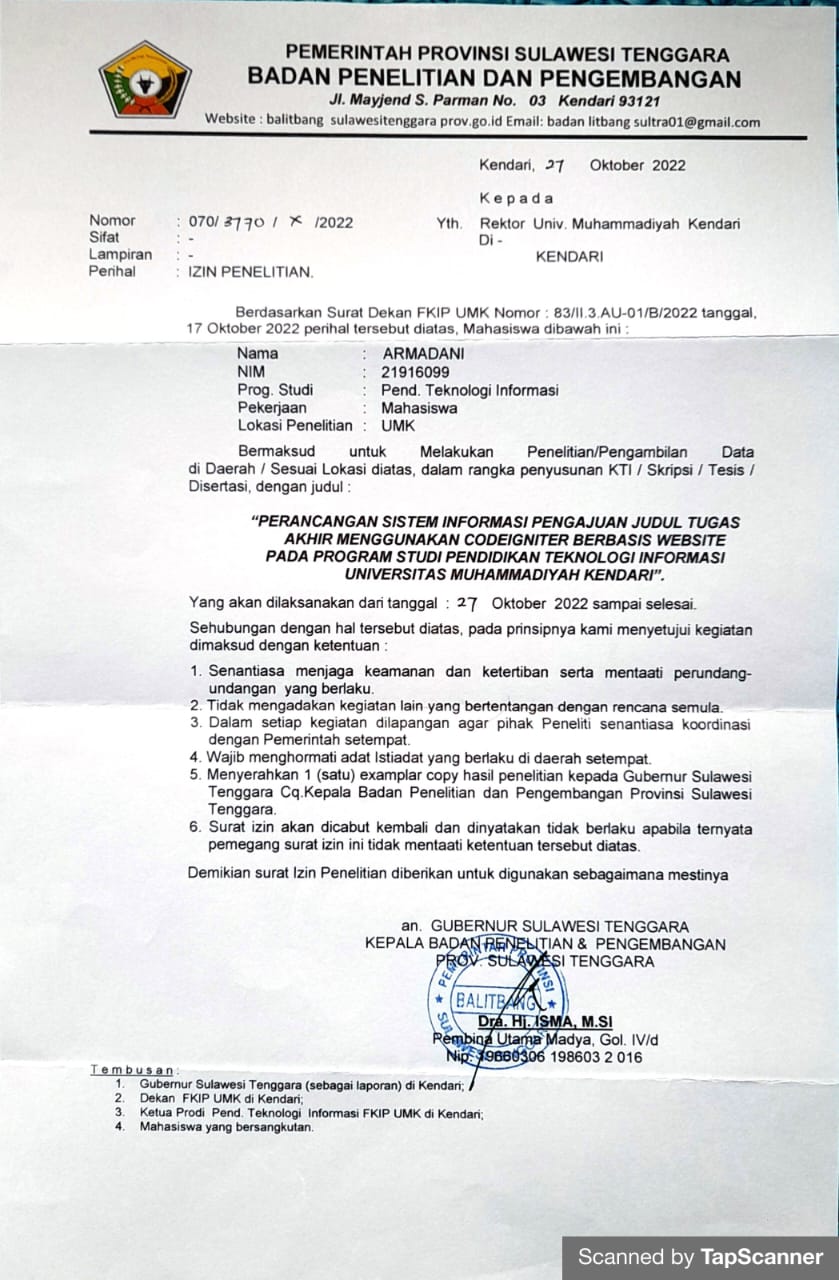
**Lampiran 1**

***STANDARD OPERATING PROCEDURE* (SOP) PERANCANGAN SISTEM INFORMASI PENGAJUAN JUDUL TUGAS AKHIR MENGGUNAKAN *CODEIGNITER* BERBASIS *WEBSITE* PADA PROGRAM STUDI PENDIDIKAN TEKNOLOGI INFORMASI UNIVERSITAS MUHAMMADIYAH KENDARI**

****

**Lampiran 2**

**SURAT IZIN PENELITIAN**

****

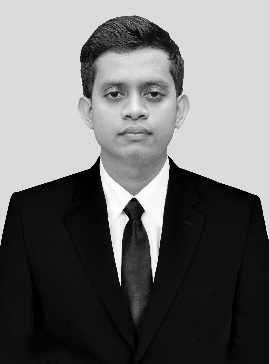
**Lampiran 3**

**DOKUMENTASI DENGAN STAF PRODI PTI**

****

****

**DAFTAR RIWAYAT HIDUP**



Nama lengkap, Armadani lahir di Buton Tengah pada tanggal 18 Oktober 2000. Merupakan anak ke-dua dari empat bersudara, dari pasangan Bapak Arzuman dan Ibu Maldia. Berkebangsaan indonesia dan beragama islam. Kini beralamat di JL. Sisingamangaraja, No. 41, Rahandouna, Kecamatan Poasia, Kota Kendari, Provinsi Sulawesi Tenggara.

Adapun Riwayat pendidikan, yaitu memasuki pendidikan dasar (SD) pada tahun 2006 di SDN 1 Mawasangka Tengah dan tamat pada tahun 2012, melanjutkan sekolah pendidikan menengah pertama (SMP) pada tahun 2012 di SMPN 06 Buton Tengah, kemudian melanjutkan jenjang pendidikan sekolah menengah kejuruan pada tahun 2015 di SMKN 1 Mawasangka Tengah dan tamat pada tahun 2018. Setelah itu melanjutkan pendidikan di Universitas Muhammadiya Kendari pada tahun 2019 dengan Program Studi S1 Pendidikan Teknologi Informasi (PTI) Fakultas Keguruan Dan Ilmu Pendidikan Universitas Muhammadiyah Kendari.