**SKRIPSI**

**SISTEM KEARSIPAN DIGITAL DOKUMEN AKADEMIK MAHASISWA BERBASIS WEB PADA FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS ISLAM KALIMANTAN BANJARMASIN**

**Oleh :**

**RIFQI MULYAWAN**

**NPM : 16.63.1090**



**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ISLAM KALIMANTAN**

**MUHAMMAD ARSYAD AL BANJARI**

**BANJARMASIN**

**2018**

# SKRIPSI

# LEMBAR PENGESAHAN PEMBIMBING

**SKRIPSI**

**SISTEM KEARSIPAN DIGITAL DOKUMEN AKADEMIK MAHASISWA BERBASIS WEB PADA FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS ISLAM KALIMANTAN BANJARMASIN**

**Oleh :**

**RIFQI MULYAWAN**

**NPM : 16.63.1090**

|  |  |
| --- | --- |
| **Pembimbing I,**  **Ikhsanul Fikri, M.Kom**  **NIK. 061510835** | **Pembimbing II,**  **Andie, M.Kom**  **NIK. 061510820** |
| Mengetahui,  **Dekan Fakultas Teknologi Informasi**  **Universitas Islam Kalimantan**  **Dr. Silvia Ratna, M.Kom**  **NIP. 19750913 200501 2 001** | |

# LEMBAR PENGESAHAN PENGUJI

**SISTEM KEARSIPAN DIGITAL DOKUMEN AKADEMIK MAHASISWA BERBASIS WEB PADA FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS ISLAM KALIMANTAN BANJARMASIN**

**Oleh :**

**RIFQI MULYAWAN**

**NPM : 16.63.1090**

Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana (S-1) Program Studi Teknik Informatika

Fakultas Teknologi Informasi Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin

**Banjarmasin, Juli 2018**

|  |  |
| --- | --- |
| **Tim Penguji** | **Tanda Tangan** |
| **Ikhsanul Fikri, M.Kom**  **Penguji I** |  |
| **Andie, M.Kom**  **Penguji II** |  |
| **Ir. H. M. Muflih, M.Kom**  **Penguji III** |  |
| Mengetahui,  **Dekan Fakultas Teknologi Informasi**  **Universitas Islam Kalimantan,**  **Dr. Silvia Ratna, M.Kom**  **NIP. 19750913 200501 2 001** | |

# PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa Skripsi dengan Judul :

**“SISTEM KEARSIPAN DIGITAL DOKUMEN AKADEMIK MAHASISWA BERBASIS WEB PADA FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS ISLAM KALIMANTAN BANJARMASIN”**

Yang dibuat untuk melengkapi peryaratan menjadi Sarjana pada program studi Teknik Informatika Fakultas Teknologi Informasi Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari Skripsi yang sudah dipublikasikan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

|  |
| --- |
| Banjarmasin, 13 Juli 2018 |
| **RIFQI MULYAWAN**  **NPM. 16.63.1090** |

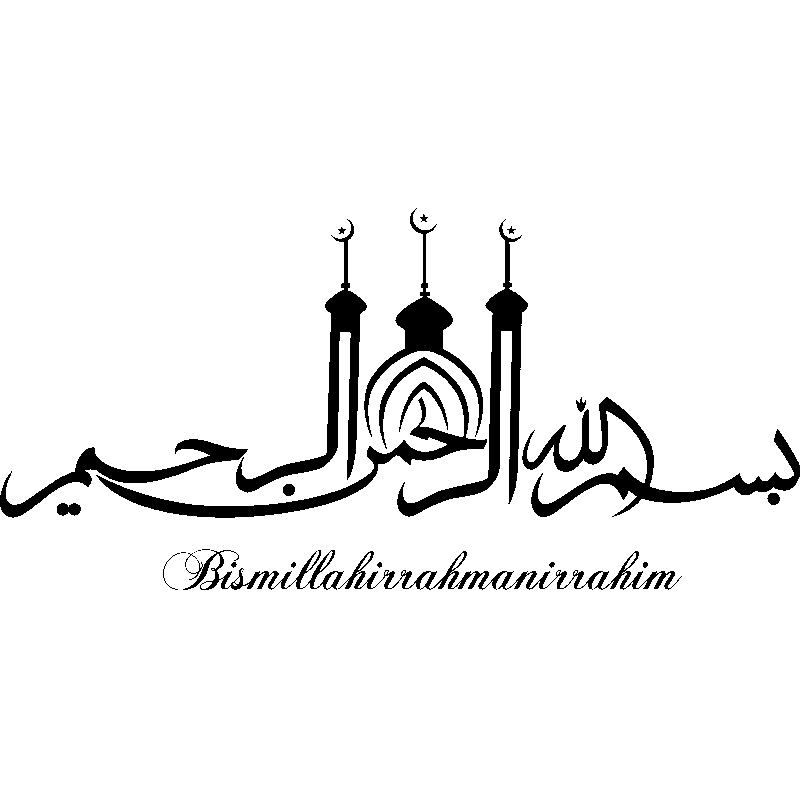
# Halaman Abstrak

**Rifqi Mulyawan, Sistem Kearsipan Digital Dokumen Akademik Mahasiswa Berbasis Web Pada Fakultas Teknologi Informasi Universitas Islam Kalimantan** dibawah bimbingan **Ikhsanul Fikri, M.Kom** dan **Andie, M.Kom**

Arsip sebagai salah satu sumber data dan merupakan hal yang penting bagi sebuah lembaga pendidikan terutama bagi mahasiswa, diperlukan adanya pengelolaan yang baik terhadap arsip. Dalam upaya penekanan kendala-kendala yang dihadapi dalam pengelolaan arsip yang mengakibatkan sulitnya pencarian arsip yang disebabkan arsip yang tidak dikelola dengan sistem penyimpanan yang baik, penyebaran informasi dari arsip yang bersifat tertutup dan rahasia kepada orang-orang yang tidak berkepentingan, dan hilangnya arsip, serta sebagai upaya pemanfaatan teknologi maka pengelolaan arsip memerlukan suatu sistem yang digunakan untuk mempermudah pengelolaan arsip. Dengan memanfaatkan teknologi computer, penulis mencoba merancang sistem kearsipan digital dokumen akademik mahasiswa untuk mengatasi permasalahan – permasalahan tersebut. Dunia pemrograman saat ini, baik itu *desktop* maupun *web based*, semakin marak pengerjaannya denggan menggunakan *framework* dan salah satu *framework* berbasis PHP yang banyak digunakan yaitu Codeigniter (CI). *Framework* CI dikembangkan untuk memudahkan dalam *developing* aplikasi dengan struktur *file source code* – nya menggunakan pendekatan *Models – Views – Controller* (MVC) dan pemrograman berorientasi objek. Oleh sebab itu penulis menggunakan CI dan *CSS Framework* Bootstraps v.4.1.1 dalam pengembangan aplikasi ini dengan metode *Object – Oriented Analyst and Design* sebagai metode pengembangan sistem. Aplikasi ini dirancang untuk dapat diakses dalam internal kampus, sehingga memudahkan mahasiswa untuk mengaksesnya. Aplikasi ini juga memberikan kemudahan dalam proses manajemen arsip dokumen akademik, dengan menyediakan kategori pengarsipan dokumen, selain itu aplikasi juga memiliki performa yang handal serta mudah untuk di – *maintenance* dan dikembangkan lebih lanjut seiring perkembangan kebutuhan para penggunanya.

**Kata Kunci : *Framework,* Codeigniter, Object – Oriented, Arsip Digital Dokumen, Akademik.**

# KATA PENGANTAR



***Assalamualaikum Warahmatullahi Wabarakatuh***

Alhamdulillah hirabbil a’lamin, kata yang dapat penulis ucapkan kepada Allah SWT, atas segala rahmat dan hidayah yang telah dilimpahkan – Nya, sehingga penulis bisa menyelesaikan laporan skripsi ini. Shalawat serta salam senantiasa tercurah kepada Rasulullah SAW, insan yang dapat membawa dunia ini kepada kehidupan yang lebih baik.

Dengan rasa syukur yang mendalam penulis dapat menyelesaikan laporan skripsi dengan judul : “Sistem Kearsipan Digital Dokumen Akademik Mahasiswa Pada Fakultas Teknologi Informasi Universitas Islam Kalimantan Banjarmasin”.

Sehubungan dengan selesainya laporan ini penulis juga mengucapkan terima kasih kepada semua pihak yang telah membantu penulis, juga atas dorongan dan bimbingannya sehingga laporan ini bisa selesai tanpa ada hambatan yang berarti. Semoga Allah SWT membalas semua amal baik mereka. Terima kasih penulis ucapkan kepada :

1. Bapak Abdul Malik, SPt., M.Si., Ph.D, sebagai Rektor Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin,
2. Ibu Dr. Silvia Ratna, M.Kom, selaku Dekan Fakultas Teknologi Informasi,
3. Bapak Yusri Ikhwani, M.Kom, selaku Kepala Program Studi Teknik Informatika,
4. Bapak Ikhsanul, M.Kom, selaku dosen pembimbing pertama,
5. Bapak Andie, M.Kom, selaku dosen pembimbing kedua,
6. Dosen – dosen jurusan Teknik Informatika dan Sistem informasi yang telah memberikan ilmu, pengetahuan, kemampuan dan *skill* manajemen beserta pemupukan moral dan mental yang baik,
7. Kedua orang tuaku yang aku cintai (abah Akhyadi dan mama Ani Noor Asifah, S.Sos), yang selalu memberikan doanya dengan ikhlas, *Allahummaghfirli waliwaalidayya warhamhuma kamaa rabbayanaa shagira*, memberikan motivasi, semangat, moril maupun materil, dan kasih sayang, serta mengajarkan arti hidup dan senantiasa mencurahkan dukungannya demi kesuksesan penulis dalam proses kuliah dan kehidupan ini,
8. *My Lovely sister*, Nadia Aisyah, *My Upa*’s, Fadliansyah dan Rahmadiansyah, S.Sos beserta istri yang selalu mendoakan, memberikan motivasi dan dukungan moril maupun materil hingga skripsi ini selesai,
9. Teman Hatiku, Noor Hasanah Ananda, S.Kom yang selalu mendoakan, memberikan dukungan penuh, semangat, motivasi, menjadi teman cerita, serta hiburan selama pembuatan skripsi ini. Terima kasih
10. Semua teman – teman seperjuangan, mahasiswa TI, sukses untuk kita semua
11. Seluruh pihak yang telah membantu dan namanya tidak dapat disebutkan satu persatu. Terima kasih atas dukungan dan motivasinya.

Semoga Allah SWT memberikan balasan yang setimpal atas segala bantuan dan kebaikan yang telah mereka berikan kepada penulis, Amiiin. Demikianlah, semoga apa yang tertulis dalam laporan skripsi ini dapat bermanfaat bagi semua pihak. Dalam hal ini, penulis mengharapkan kritik dan saran yang membangun demi menambah kesempurnaan laporan skripsi ini.

***Waalaikumsalam Warahmatullahi Wabarakatuh***

|  |
| --- |
| Banjarmasin, Juni 2018 |
|  |
| (Rifqi Mulyawan) |

# DAFTAR ISI

[**SKRIPSI i**](#_Toc516342755)

[**LEMBAR PENGESAHAN PEMBIMBING ii**](#_Toc516342756)

[**LEMBAR PENGESAHAN PENGUJI iii**](#_Toc516342757)

[**PERNYATAAN KEASLIAN SKRIPSI iv**](#_Toc516342758)

[**Halaman Abstrak v**](#_Toc516342759)

[**KATA PENGANTAR vi**](#_Toc516342760)

[**DAFTAR ISI ix**](#_Toc516342761)

[**DAFTAR GAMBAR xii**](#_Toc516342762)

[**DAFTAR TABEL xiii**](#_Toc516342763)

[**BAB I PENDAHULUAN 1**](#_Toc516342764)

[1.1 Latar Belakang Masalah 1](#_Toc516342765)

[1.2 Rumusan Masalah 3](#_Toc516342766)

[1.3 Batasan Masalah 4](#_Toc516342767)

[1.4 Tujuan Penelitian 4](#_Toc516342768)

[1.5 Manfaat Penelitian 4](#_Toc516342769)

[1.6 Metode Penelitian 5](#_Toc516342770)

[1.7 Sistematika Penulisan 6](#_Toc516342771)

[**BAB II TINJAUAN PUSTAKA 8**](#_Toc516342772)

[2.1 Landasan Teori 8](#_Toc516342773)

[2.1.1 Sistem Informasi 8](#_Toc516342774)

[2.1.2 Arsip Digital 9](#_Toc516342775)

[2.1.3 Dokumen dan Akademik 11](#_Toc516342776)

[2.1.4 Sistem Informasi Arsip 11](#_Toc516342777)

[2.1.5 *Object Oriented Analysis and Design (OOAD)* 14](#_Toc516342778)

[2.1.6 Bahasa Pemrograman PHP 15](#_Toc516342779)

[2.1.7 Database MySQL 18](#_Toc516342780)

[2.1.8 Framework Codeigniter 19](#_Toc516342781)

[2.2 Penelitian Terkait 23](#_Toc516342782)

[2.3 Profil Objek Penelitian 27](#_Toc516342783)

[**BAB III ANALISIS DAN PERANCANGAN SISTEM 29**](#_Toc516342784)

[3.1 Analisis Sistem yang berjalan 29](#_Toc516342785)

[3.2 Analisis Kebutuhan Sistem 31](#_Toc516342786)

[3.2.1 Kebutuhan *Hardware* 31](#_Toc516342787)

[3.3 Rancangan Model Sistem 33](#_Toc516342788)

[3.4 Rancangan Basis Data 42](#_Toc516342789)

[3.5 Rancangan Antarmuka Masukan Sistem 46](#_Toc516342790)

[3.6 Rancangan Antarmuka Keluaran Sistem 50](#_Toc516342791)

[3.7 Tempat dan Jadwal Penelitian 55](#_Toc516342792)

[**BAB IV IMPLEMENTASI DAN PENGUJIAN 56**](#_Toc516342793)

[4.1 Spesifikasi Sistem 56](#_Toc516342794)

[4.2 Langkah – Langkah Pembuatan Sistem 57](#_Toc516342795)

[4.3 Hasil Tampilan Aplikasi 78](#_Toc516342796)

[4.4 Pengujian 91](#_Toc516342797)

[**BAB V PENUTUP 99**](#_Toc516342798)

[5.1 Kesimpulan 99](#_Toc516342799)

[5.2 Saran 100](#_Toc516342800)

[**DAFTAR PUSTAKA 101**](#_Toc516342801)

**LAMPIRAN**

# DAFTAR GAMBAR

[**Gambar 2. 1 Model Pengelolaan Arsip Berbasis TIK** 10](#_Toc516413614)

[**Gambar 3. 1 *Flow Of Document* Proses Dokumen Akademik Mahasiswa** 29](#_Toc516413636)

[**Gambar 3. 2 *Use Case Sistem*** 33](#_Toc516413637)

[**Gambar 3. 3 *Class Diagram* Sistem** 34](#_Toc516413638)

[**Gambar 3. 4 *Activity Diagram Login*** 35](#_Toc516413639)

[**Gambar 3. 5 *Activity Diagram* Prodi** 36](#_Toc516413640)

[**Gambar 3. 6 *Activity Diagram* Fakultas** 36](#_Toc516413641)

[**Gambar 3. 7 *Activity Diagram* Upload Audio** 37](#_Toc516413642)

[**Gambar 3. 8 *Activity Diagram* Upload File** 37](#_Toc516413643)

[**Gambar 3. 9 *Activity Diagram* Upload Gambar** 38](#_Toc516413644)

[**Gambar 3. 10 *Activity Diagram* Upload Note** 39](#_Toc516413645)

[**Gambar 3. 11 *Activity Diagram* Data Kelas** 39](#_Toc516413646)

[**Gambar 3. 12 *Sequence Diagram* Login** 40](#_Toc516413647)

[**Gambar 3. 13 *Sequence Diagram* Data *Management*** 41](#_Toc516413648)

[**Gambar 3. 14 Halaman Utama Sistem** 46](#_Toc516413649)

[**Gambar 3. 15 Halaman Masukan Data** 47](#_Toc516413650)

[**Gambar 3. 16 Antarmuka *Edit* Profil *User*** 48](#_Toc516413651)

[**Gambar 3. 17 Antarmuka *Edit Password*** 49](#_Toc516413652)

[**Gambar 3. 18 Antarmuka *Dashboard*** 50](#_Toc516413653)

[**Gambar 3. 19 Antarmuka Data *Management*** 51](#_Toc516413654)

[**Gambar 3. 20 Antarmuka Profil** 52](#_Toc516413655)

[**Gambar 3. 21 Antarmuka Halaman Help** 53](#_Toc516413656)

[**Gambar 4. 1 Alur Kerja *Framework* Codeigniter** 57](#_Toc516413680)

[**Gambar 4. 2 Halaman *Landing*** 77](#_Toc516413681)

[**Gambar 4. 3 Halaman *Input* Data Prodi** 78](#_Toc516413682)

[**Gambar 4. 4 Halaman *Input* Data Fakultas** 79](#_Toc516413683)

[**Gambar 4. 5 Halaman *Upload* Arsip Note** 79](#_Toc516413684)

[**Gambar 4. 6 Halaman *Upload* Arsip File** 80](#_Toc516413685)

[**Gambar 4. 7 Halaman *Upload* Arsip Gambar** 81](#_Toc516413686)

[**Gambar 4. 8 Halaman *Upload* Arsip Audio** 81](#_Toc516413687)

[**Gambar 4. 9 Halaman *Upload* Arsip Video** 82](#_Toc516413688)

[**Gambar 4. 10 Halaman Ganti *Password*** 83](#_Toc516413689)

[**Gambar 4. 11 Halaman *Dashboard Super\_User*** 83](#_Toc516413690)

[**Gambar 4. 12 Halaman *Dashboard User*** 84](#_Toc516413691)

[**Gambar 4. 13 Halaman *View* Data Prodi** 85](#_Toc516413692)

[**Gambar 4. 14 Halaman *View* Data Fakultas** 85](#_Toc516413693)

[**Gambar 4. 15 Halaman *View* Data Arsip Note** 86](#_Toc516413694)

[**Gambar 4. 16 Halaman *View* Data Arsip File** 87](#_Toc516413695)

[**Gambar 4. 17 Halaman *View* Data Arsip Gambar** 87](#_Toc516413696)

[**Gambar 4. 18 Halaman *View* Data Arsip Video** 88](#_Toc516413697)

[**Gambar 4. 19 Halaman *View* Data Arsip Video** 89](#_Toc516413698)

[**Gambar 4. 20 Halaman Profil** 89](#_Toc516413699)

[**Gambar 4. 21 Halaman *Help*** 90](#_Toc516413700)

[**Gambar 4. 22 *Flowchart Dan Flowgraph Menu* Pengarsipan Digital** 91](#_Toc516413701)

# DAFTAR TABEL

[**Tabel 2. 1 Macam Tag Untuk Memasukkan Kode PHP** 17](#_Toc516413983)

[**Tabel 3. 1 Tabel User** 42](#_Toc516413975)

[**Tabel 3. 2 Tabel Prodi** 42](#_Toc516413976)

[**Tabel 3. 3 Tabel Fakultas** 43](#_Toc516413977)

[**Tabel 3. 4 Tabel Note** 43](#_Toc516413978)

[**Tabel 3. 5 Tabel u\_audio** 44](#_Toc516413979)

[**Tabel 3. 6 Tabel u\_file** 44](#_Toc516413980)

[**Tabel 3. 7 Tabel u\_gambar** 45](#_Toc516413981)

[**Tabel 3. 8 Tabel u\_audio** 45](#_Toc516413982)

[**Tabel 4. 1 Tabel Struktur File** 56](#_Toc516413994)

[**Tabel 4. 2 List Hasil Pengujian *Black-Box*** 93](#_Toc516413995)

# BAB I PENDAHULUAN

## Latar Belakang Masalah

Arsip sebagai salah satu sumber data dan merupakan hal yang penting bagi sebuah lembaga pendidikan terutama bagi mahasiswa, diperlukan adanya pengelolaan yang baik terhadap arsip. Pengelolaan arsip tersebut meliputi proses pencatatan, proses pendistribusian, proses penyimpanan, proses perawatan, proses pengamanan, proses penyusutan hingga proses pengawasan. Pengelolaan arsip yang baik akan memudahkan penemuan kembali arsip ketika dibutuhkan sehingga informasi yang ingin diketahui dari arsip tersebut dapat diperoleh tanpa membutuhkan banyak waktu. Namun pada kenyataannya, pentingnya peranan arsip tidak diimbangi dengan pengelolaan arsip yang baik di organisasi-organisasi kelembagaan. Adanya keterbatasan sumber daya manusia yang berkompeten di bidang kearsipan menjadi kendala utama dalam pengelolaan arsip di setiap organisasi kelembagaan. Hal ini didasarkan pada persepsi masyarakat bahwa pengelolaan arsip dapat dilakukan oleh semua orang tanpa perlu memperdalam konsep dan teori tentang pengelolaan arsip. Atas dasar persepsi tersebut, tidak banyak orang yang benar - benar menekuni bidang kearsipan dan kurang bahkan tidak ada petugas khusus yang menangani arsip (arsiparis). Arsiparis yang tidak benar-benar menekuni bidang kearsipan menjadi kurang memiliki pemahaman tentang pengelolaan arsip sehingga pengelolaannya didasarkan pada kebiasaan tanpa adanya pengembangan dalam pelaksanaan pengelolaan arsip. Kurangnya pemahaman arsiparis tentang pengelolaan arsip akan mempengaruhi sistem penataan dan penyimpanan arsip. Sistem dan proses penyimpanan arsip yang kurang sistematis dapat menyebabkan penumpukkan arsip, terselipnya arsip di tempat yang tidak semestinya, atau bahkan hilang. Hilangnya arsip berarti juga kehilangan data, informasi, dan bukti tentang suatu hal atau kegiatan dalam suatu organisasi. Selain itu, penataan dan penyimpanan arsip yang tidak baik dapat menyebabkan tersebarnya informasi dari arsip kepada pihak-pihak yang tidak berkepentingan atau yang tidak bertanggung jawab atau yang memiliki maksud dan tujuan yang tidak baik terhadap suatu organisasi. Banyaknya pengelolaan arsip di suatu organisasi kelembagaan yang masih bersifat konvensional dihadapkan pada keterbatasan fasilitas yang mana fasilitas tersebut memiliki peran dan fungsi sebagai penunjang serta pendukung dalam proses pengelolaan arsip.

Dalam upaya penekanan kendala-kendala yang dihadapi dalam pengelolaan arsip yang mengakibatkan sulitnya pencarian arsip yang disebabkan penumpukkan arsip atau arsip yang tidak dikelola dengan sistem penyimpanan yang baik, penyebaran informasi dari arsip yang bersifat tertutup dan rahasia kepada orang-orang yang tidak berkepentingan, dan hilangnya arsip, serta sebagai upaya pemanfaatan teknologi maka pengelolaan arsip memerlukan suatu sistem yang digunakan untuk mempermudah pengelolaan arsip. Sistem yang tidak lagi bersifat konvensional tetapi telah memanfaatkan teknologi yaitu menggunakan media elektronik yang berbasiskan pada penggunaan komputer. Penggunaan media elektronik dalam pengelolaan arsip ini disebut dengan sistem kearsipan elektronik (digital).

Berdasarkan hal tersebut, peneliti tertarik untuk melakukan penelitian berkaitan dengan bidang kearsipan, khususnya sistem kearsipan elektronik (digital) dokumen akademik mahasiswa. Penelitian ini dilakukan di Universitas Islam Kalimantan Muhammad Arsyad Al Banjari Banjarmasin, di Fakultas Teknologi Informasi, peneliti tertarik untuk melihat gambaran nyata dari penerapan sistem kearsipan elektronik terutama dalam hal pengarsipan dokumen akademik Mahasiswa. Maka dari itu, peneliti melakukan penelitian dengan judul ”**Sistem Kearsipan Digital Dokumen Akademik Mahasiswa Berbasis Web Pada Fakultas Teknologi Informasi Universitas Islam Kalimantan Banjarmasin**”.

## Rumusan Masalah

Adapun rumusan masalah dalam penelitian tugas akhir ini antara lain sebagai berikut :

1. Bagaimana cara merancang sistem kearsipan yang pada Mahasiswa Fakultas Teknologi Informasi Universitas Islam Kalimantan Muhammad Arsyad Al Banjari?
2. Bagaimana cara membuat sistem kearsipan digital dokumen akademik Mahasiswa Fakultas Teknologi Informasi Universitas Islam Kalimantan Muhammad Arsyad Al Banjari?

## Batasan Masalah

Batasan masalah pada penelitian ini adalah :

1. Objek yang dijadikan oleh peneliti pada penelitian ini adalah pengarsipan yang dilakukan mahasiswa pada Fakultas Teknologi Informasi Universitas Islam Kalimantan Muhammad Arsyad Al-Banjari Banjarmasin ?
2. Penelitian berfokus hanya pada manajemen dokumen akademik mahasiswa, misal seperti berkas KHS (kartu hasil studi), KRS (kartu hasil studi), sertifikat – sertifikat dan lain sebagainya yang berhubungan dengan kegiatan akademik mahasiswa.

## Tujuan Penelitian

Berdasarkan rumusan masalah yang ada, maka tujuan dari penelitian ini adalah merancang sistem yang mempermudah kegiatan akademik mahasiswa khususnya dalam pengarsipan dokumen akademik Mahasiswa Fakultas Teknologi Informasi Universitas Islam Kalimantan yang bisa digunakan kapanpun, dan dimanapun serta *Compatible* dengan berbagai perangkat.

## Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah :

1. Secara teoritis, hasil penelitian ini diharapkan dapat memberikan sumbangan pengetahuan, wawasan, dan pemikiran yang bermanfaaat bagi perkembangan ilmu pengetahuan dan pendidikan.
2. Bagi objek penelitian, penelitian ini diharapkan dapat menjadi contoh dan memberikan inspirasi serta motivasi kepada lembaga pendidikan untuk dapat menerapkan sistem dengan memanfaatkan teknologi dalam pengelolaan arsip dokumen pendidikan.
3. Bagi Mahasiswa, hasil penelitian ini diharapkan dapat mempermudah manajemen pengarsipan dokumen akademik mahasiswa

## Metode Penelitian

Agar permasalahan yang dipaparkan dapat ditemukan pemecahan masalahnya maka diperlukan beberapa metode penelitian dari penelitian ini. Metode penelitian yang dimaksud yaitu metode pengumpulan data, analisa *system*, desain *system*, pemrograman dan implementasi.

1. **Metode Pengumpulan Data**
2. Observasi

Dalam observasi ini yang dilakukan yaitu meliat langsung kondisi tempat penelitian yang dalam hal ini yaitu Mahasiswa yang ada Fakultas Taknologi Informasi Universitas Islam Kalimantan Banjarmasin.

1. Wawancara

Melakukan wawancara kepada bagian Mahasiswa dalam manajemen dokumen akademik mahasiswa.

1. Studi Pustaka

Melakukan Studi kepustakaan dengan mengumpulkan data dan informasi mengenai prosedut sistem yang telah ada, analisa perancangan sistem berorientasi objek dan Bahasa pemrograman yang dapat dijadikan acuan pembahasan dalam masalah ini

1. Studi Sejenis

Dalam penelitian ini, diperlukan sebuah perbandingan studi literature sejenis yang erat hubungannya dengan tema penulisan skripsi ini. Perbandingan studi sejenis ini diperlukan agar nantinya penelitian ini dapat bermanfaat dan menjadi pelengkap dan penyempurna dari studi – studi literatur yang telah dilaksanakan sebelumnya

1. **Metodologi Pengembangan Sistem**

Proses pengembangan sistem kearsipan digital dokumen akademik mahasiswa ini menggunakan OOAD (*Object Oriented Analysis and Design*) yang terdiri dari metode analisis dan metode perancangan. (Nugroho : 2005). Alat bantu pemodelan sistem yang digunakan adalah *Unified Modelling Language* (*UML*).

## Sistematika Penulisan

Sistematika yang digunakan dalam penulisan Skripsi ini terdiri dari 5 Bab yang diuraikan sebagai berikut :

* + 1. BAB I : PENDAHULUAN Dalam bab ini akan dibahas tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian dan sistematika penulisan.
    2. BAB II : TINJAUAN PUSTAKA Dalam bab ini membahas tentang dasar-dasar teori yang melandasi penyusunan dan perancangan dalam pengembangan sistem perangkat lunak, hasil-hasil penelitian sebelumnya atau penelitian terkait dan profil objek penelitian
    3. BAB III : ANALISIS DAN PERANCANGAN SISTEM Dalam bab ini menguraikan tentang analisis sistem yang sedang berjalan, analisis kebutuhan sistem, rancangan model sistem, rancangan struktur basis data dan rancangan masukan dan Keluaran sistem.
    4. BAB IV : IMPLEMENTASI DAN PENGUJIAN Dalam bab ini membahas tentang spesifikasi sistem, langkah-langkah pembuatan sistem, tampilan aplikasi yaitu implementasi dari rancangan sistem yang telah dibuat dan pengujian.
    5. BAB V : PENUTUP Dalam bab ini membahas tentang kesimpulan serta saran untuk dilaksanakan lebih lanjut guna pengembangan penelitian.

# BAB II TINJAUAN PUSTAKA

## Landasan Teori

Untuk mendukung pembuatan laporan ini, maka perlu dikemukakan hal-hal atau teori-teori yang berkaitan dengan permasalahan dan ruang lingkup pembahasan sebagai landasan dalam pembuatan laporan ini.

### Sistem Informasi

Sistem adalah kumpulan dari elemen – elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. (Jogiyanto : 2005)

Menurut Jerry Fitzchald, Ardra F. FitzGerald dan Warren D. Stallings, Jr., suatu sistem adalah suatu jaringan dari perosedur – prosedur yang saling berhubungan, berkumpul bersama – sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. (Jogiyanto : 2005)

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sumber dari informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal datum atau data item. Data adalah kenyataan yang menggambarkan suatu kejadian – kejadian dan kesatuan nyata. Kejadian – kejadian (*events*) adalah sesuatu yang terjadi pada saat tertentu. Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. (Jogiyanto : 2005)

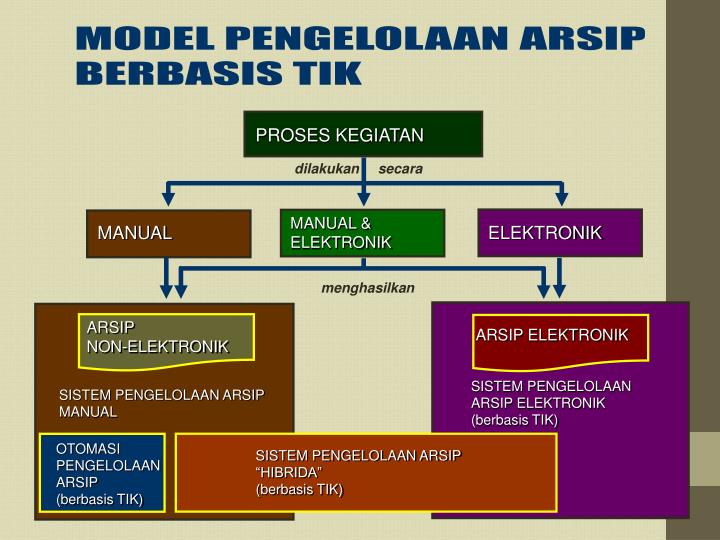
Informasi juga dapat menggambarkan kejadian nyata yang digunakan untuk pengambilan keputusan. Sumber dari informasi adalah data yang berbentuk huruf, simbol, alfabet, dan sebagainya.

### Arsip Digital

Arsip sangat penting dalam sebuah lembaga atau pun institusi karena arsip merupakan catatan yang fungsinya sebagai pengingat atau memori dari peristiwa yang pernah dialami dari waktu ke waktu. Isi dari arsip biasanya merupakan fakta-fakta dari identitas, harkat dan perwujudan aktivitas sebuah lembaga ataupun institusi yang akan disimpan dari generasi ke generasi selanjutnya menurut Pustaka Arsip Kampar (2015).

Menurut Undang-undang Nomor 43 Tahun 2009 pasal 1 bahwa arsip adalah rekaman kegiatan atau peristiwa dalam berbagai bentuk dan media sesuai dengan perkembangan teknologi informasi dan komunikasi yang dibuat dan diterima oleh lembaga negara, pemerintahan daerah, lembaga pendidikan, perusahaan, organisasi politik, organisasi kemasyarakatan, dan perseorangan dalam pelaksanaan kehidupan bermasyarakat, berbangsa, dan bernegara. Dan arsip perguruan tinggi adalah lembaga kearsipan berbentuk satuan organisasi perguruan tinggi, baik negeri maupun swasta yang melaksanakan fungsi dan tugas penyelenggaraan kearsipan di lingkungan perguruan tinggi.

Arsip digital berdasarkan Universitas Indonesia adalah arsip yang berbentuk eletronik dan cara menggunakan arsip tersebut juga dalam bentuk elektronik. Arsip digital ini berasal dari arsip yang berbentuk *hardcopy* kemudian dikonversi dalam bentuk digital. Untuk mengelola arsip digital juga harus sama dengan bentuk arsip yang mulanya berupa *hardcopy*. Dengan adanya arsip digital akan memberikan kemudahan dalam pencarian arsip walaupun arsip yang dicari sudah sangat lama sekali.



**Gambar 2. 1 Model Pengelolaan Arsip Berbasis TIK**

Menurut kesimpulan penelitian yang berjudul “Pengelolaan Arsip Digital” oleh Muhidin tahun 2016 bahwa arsip digital adalah Arsip yang dapat digunakan dan dimanfaatkan serta arsip yang sudah disimpan dalam bentuk kode-kode biner yang dapat dikelola seperti membuat, membuka, dan menghapus arsip tersebut dengan alat komputasi yang dapat membaca ataupun mengolah arsip. Penerapan penyimpanan arsip secara digital juga memberikan keuntungan seperti ruang penyimpanan tidak memakan tempat dan mempermudah pencarian berkas tersebut karena tidak melakukan pencarian secara manual, maksud dari manual disini adalah mencari dirak berkas satu-satu. Selain itu juga akan menghemat waktu.

### Dokumen dan Akademik

Dokumen atau sahifah adalah sebuah tulisan yang memuat informasi. Biasanya, dokumen ditulis di kertas dan informasinya ditulis memakai tinta baik memakai tangan atau memakai media elektronik (seperti printer). (Wikipedia, 2018)

Sedangkan Akademik digunakan untuk menggambarkan sesuatu yang behubungan dengan pekerjaan yang ada disekolah, kampus dan universitas, terutama pekerjaan yang menyangkut dengan belajar dan penalaran daripada keterampilan praktis dan teknis. (collinsdictionary, 2018).

### Sistem Informasi Arsip

Sistem informasi pada organisasi atau institusi yang didalamnya terjadi transaksi informasi dan juga pengolahannya berdasarkan alur sistem yang sudah dibuat disertai hasil pembuatan laporan-laporan transaksi informasi tersebut dengan pihak yang berkaitan . Kemudian sistem informasi tersebut diimplementasikan dengan komputerisasi sehingga pentingnya peran sumber daya yaitu manusia dan komputer. Manusia sebagai user untuk input informasi tersebut dan proses output dari komputerisasi guna mencapai hasil yang diinginkan (Unsri, 2017).

Sistem Informasi Kearsipan adalah suatu sistem informasi yang mengelola data yang menyangkut pengumpulan, pengelolaan, pemusnahan, pencetakan laporan dan pencarian kembali arsip yang berbasis komputer sehingga mampu mengelola arsip dengan lebih efektif dan efesien dan pada akhirnya dapat memberi masukan informasi secara aktual dan akurat tentang perumusan kebijakan, strategi dan program pembangunan Sistem Informasi Kearsipan merupakan perangkat-perangkat yang saling berinteraksi dalam pengolahan data kearsipan untuk mencapai tujuan dari fungsi kearsipan, yaitu penyimpanan, penataan, pengelompokan, pengendalian dan pemeliharaan kearsipan (Tatalia, 2016).

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Karakteristik sistem sebagai berikut :

* + 1. Memiliki komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak perduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan industri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, makaperusahaan dapat disebut sebagai subsistem. Demikian juga bila perusahaan dipandang sebagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

* + 1. Batas sistem (boundary)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

* + 1. Lingkungan luar sistem (environment)

Adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.

* + 1. Penghubung sistem (interface)

Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya.

* + 1. Masukan sistem (input)

Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam sistem komputer, program adalah maintanance input yangdigunakan untuk mengoperasikan komputernya dan data adalah signal input untuk diolah menjadi informasi.

* + 1. Keluaran sistem (Output)

Merupakan hasil dari energi yang diolah oleh sistem.

* + 1. Pengolah sistem (Process)

Adalah bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

* + 1. Sasaran sistem

Tentunya sistem harus memiliki sararan kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada guna (Sukana, 2016).

### *Object Oriented Analysis and Design (OOAD)*

OOAD adalah metode analisis yang memeriksa *requirements* dari sudut pandang kelas – kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek – objek sistem atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas.

OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOD). OOA adalah motode analisis yang memeriksa *requirement* (syarat / keperluan yang harus dipenuhi sebuah sistem) dari sudut pandang kelas – kelas dan objek – objek yang ditemui dalam ruang lingkup perusahaan. Sedangkan OOD adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek – objek sistem atau subsistem.

### Bahasa Pemrograman PHP

PHP singkatan dari *Hypertext Preprocessor* yaitu bahasa pemprograman web server-side yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada sever (*server side* HTML *embedded scripting*). PHP adalah script yang digunakan untuk membuat halamn website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta selalu yang terbaru. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan (Angar, 2010).

PHP adalah bahasa *open source* yang dapat digunakan diberbagai sistem informasi seperti: Linux, Unix, Macintosh dan Windows. PHP dapat dijalankan secara runtime melalui console serta dapat menjalan perintah - perintah sistem. Open source artinya kode kode PHP tebuka untuk umum dan kita tidak harus membayar biaya pembelian atas keaslian license yang biasanya cukup mahal. Web server yang mendukung PHP dapat ditemukan dimana-mana, mulai dari Apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah. PHP juga dilengkapi dengan berbagai macam pendukung lain seperti support langsung ke berbagai macam database yang populer, misal Oracle, PostgreSQL, MySQL dan lain-lain (Angar, 2010).

PHP adalah sebuah bahasa pemprograman scripting untuk membuat halaman web yang dinamis. Walaupun dikenal sebagai bahasa membuat halaman web, tapi sebenarnya juga dapat digunakan untuk membuat aplikasi command line dan juga GUI. Tidak seperti HTML biasa, PHP tidak akan diberikan oleh server secara langsung ketika ada permintaan dari client (yaitu browser), namun melalui pemprosesan dair sisi server, makanya PHP disebut skrip server-side. Kode PHP ddimasukkan kedalam kode HTML dengan cara menyelipkan di dalam kode HTML. Untuk membedakan kode PHP dengan kode HTML, didepan kode PHP tersebut diberi tag pembuka dan diakhir kode PHP diberi tag penutup. Kode PHP yang ada ditengah kode HTML akan diproses menggunakan mesin yang disebut PHP Parser. Mesin PHP Parser akan langsung melewatkan (tidak mengolah) kode HTML dan hanya mengolah isi kode PHP (Zaki dkk, 2008)

Dengan adanya kode PHP, sebuah halaman web bisa melakukan banyak hal yang dinamis, seperti mengakses database, membuat gambar, membaca dan menulisi file dan sebagainya. Hasil akhir pengolahan kode PHP akan dikembalikan lagi dalam bentuk kode HTML untuk ditampilkan di browser. Ada empat jenis tag yang bisa digunakan untuk memasukkan kode PHP (Zaki dkk, 2008).

**Tabel 2. 1 Macam Tag Untuk Memasukkan Kode PHP**

|  |  |  |
| --- | --- | --- |
| Jenis Tag | Tag Pembuka | Tag Penutup |
| Tag Standar | <?Php | ?> |
| Tag Pendek | <? | ?> |
| Tag ASP | <% | %> |
| Tag script | <script language=”php”> | </script> |

PHP adalah suatu bahasa pemprograman berbasiskan kode-kode (script) yangd digunakan untuk mengolah suatu data dan mengirimkannya kembali ke web browser menjadi kode HTML. Kode PHP mempunyai ciri-ciri khusus, yaitu :

* Hanya dapat dijalankan menggunakan web server, misalnya Apache
* Kode PHP diletakkan dan dijalan di web server
* Kode PHP dapat digunakan untuk mengakses database seperti MySQL, PostgreSQL, Oracle dan lain-lain.
* Merupakan software yang bersifat open source
* Gratis untuk di-download dan digunakan
* Memiliki sifat multiplatform artinya dapat dijalankan menggunakan sistem operasi apapun seperti Linux, Unix, Windows dan lain-lain (Oktavian, 2010).

### Database MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*database management system*), database ini *mulithread*, *multi user .* MySQL AB membuat MySQL tesedia sebagai perangkat lunak gratis dibawah lisensi GNU General Publik License (GPL), tetapi mereka juga dibawah lisensi komersial untuk kasus-kasus yang bersifat khusus. MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public Licence). MySQL merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Languase).* SQL adalah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Huda, 2010).

Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya, terutama dalam kecepatan. Berikut ini beberapa keistemewaan MySQL antara lain :

* 1. **Portability**, Mysql dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, SOlaris, Amiga dan lain-lainnya.
  2. **Multiuser**, Mysql dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
  3. **Security**, Mysql memiliki beberapa lapisan sekuritas seperti leve; subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenskripsi.
  4. **Scability** dan **limits**, Mysql mampu menangani database dalam skala besar, dengan jumlah record lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya (Huda, 2010).

### Framework Codeigniter

Framework memudahkan pembuatan aplikasi web karena hanya dengan menggunakan fungsi-fungsi yang sudah ada framework. Fungsi-fungsi tersebut disimpan dalam beberapa file yang terpisah dan juga memudahkan dalam penggunaan *source code* yang akan digunakan berulang-ulang. Codeigniter adalah framework hasil karya Rick Ellis yang merupakan CEO pMachine selain itu juga merupakan framework *open source* dan terus dikembangkan sampai sekarang (Jubilee, 2015). *Codeigniter* menurut halaman wesbite resminya *codeigniter* adalah framework dengan bahasa pemprograman PHP yang kuat dan dibangun untuk *developer web* untuk membuat fitur aplikasi web yang lengkap serta dengan *footprint* yang sangat kecil.

Adapun fitur utama framework codeigniter adalah mempercepat dan memudahkan dalam membuat aplikasi dengan bahasa pemprograman PHP dan dapat mengatur stuktur website serta dapat mengatasi masalah-masalah struktural. Framework codeigniter lebih fokus untuk kemudahan operasi-operasi dasar seperti :

* Ujicoba website
* Membuat halaman dan *form* serta dapat membuat validasi *input* form yang berkaitan dengan html
* Dapat mengelola *sesi* dan *cookies*
* Penggunaan FTP atau XMLRPC dalam komunikasi internet
* Akses untuk query dan database (Jubilee, 2015)

Codeigniter menyediakan berbagai macam *library* yang dapat digunakan untuk pembuatan aplikasi berbasis web, cara menggunakan *library* tersebut hanya dengan memanggil class library yang telah disediakan pada framework codeigniter. Adapun kelebihan codeigniter sebagai berikut :

1. Codeigniter gratis yang sudah dilisensikan oleh apache/BSD *style open source* dan dapat digunakan secara gratis dan bebas.
2. Codeigniter mendukung bahasa pemprograman PHP versi 4 dan versi 5 dan 7 (sekarang).
3. Ukuran framework codeigniter yang cukup kecil dan cepat, maskudnya disini adalah codeigniter hanya mengambil fungsi yang akan digunakan saja. Tidak seperti framework lain yang me-load semua fungsi walaupun ada beberapa fungsi yang tidak digunakan. Maka dari itu codeigniter memiliki akses yang ringan dan juga cepat.
4. Codeigniter memiliki dokumentasi lengkap yang sudah disediakan oleh pengembang codeigniter sehingga memudahkan pengguna codeigniter untuk mempelajari dan menggunakan codeigniter dalam membuat aplikasi web. Dan dokumentasi selalu disertakan oleh pengembang saat ada versi codeigniter yang baru.
5. Konsep MVC (Model View Controller) digunakan oleh framework codeigniter. Konsep ini memudahkan para programmer dalam pengerjaan layout dan logika dalam membuat aplikasi berbasis web karena dalam framework codeigniter telah dipisahkan (Widodo, 2013).

Banyak fitur yang tersedia pada codeigniter antara lain penggunaan framework yang ringan dan cepat, menggunakan konsep MVC (Model View Controller), dapat menggunakan berbagai macam basis data, dapat digunakan pada PHP versi 4 dan versi 5, adanya fitur form dan validasi, tersedia keamanan dan XSS filtering, mendukung active record database, adanya pengaturan session, mendukung class untuk email, juga mendukung lokalisasi bahasa, enkripsi data, caching, bencmarking, serta menghasilkan clean URL, tersedia helper dan library yang cukup banyak, adanya log error yang terjadi, URL routing fleksibel dan mendukung ekstensi class dan plugin, adanya class membuat template web, trackback, mengetahui user agent hingga kalender, memanipulasi gambar, mendukung transfer file via FTP dan upload file (Widodo, 2013).

## Penelitian Terkait

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **JUDUL** | **NAMA PENELITI dan TAHUN PENELITIAN** | **MASALAH** | **APLIKASI** | **KETIDAKSESUAIAN** |
| 1. | Rancang Bangung Sistem Informasi Manajemen Kearsipan Pemerintah Daerah Istimewa Yogyakarta | Reza Gama Hidayat (2017) | seringnya kehilangan, kesulitan dalam proses pencarian arsip/dokumen dan tersimpan dengan tidak teratur pada ruang pengarsipan | Sistem Informasi Manajemen Kearsipan Pemerintah Daerah Istimewa | Perlu pengoptimalan penggunaan sistem secara menyeluruh dan berkelanjutan3 |
| 2. | Sistem Informasi *e-document* pada Badan Penjamin Mutu Akademik Universitas Respati Yogyakarta | Sri Hasta Mulyani (2014) | seringnya kehilangan arsip/dokumen, kesulitan dalam proses pencarian arsip/dokumen, arsip/dokumen tersimpan dengan tidak teratur dan masih pada unit-unit yang terdapat dalam organisasi intitusi | Sistem Informasi *e-document* | Sistem dikembangkan dengan perangkat lunak pengembangan yaitu PHP, Drupal 7.20 |
| 3. | Implementasi Sistem Kearsipan Elektronik Di Sekretariat Daerah dan Dinas Pendidikan Pemuda dan Olahraga Daerah Istemewa Yogyakarta | Rina Suyuti (2013) | Pengelolaan arsip di organisasi kelembagaan termasuk organisasi kelembagaan pendidikan, masih dilakukan secara konvensional | Aplikasi Sistem Kearsipan Elekronik Sekretarian Daerah dan Dinas Pendidikan Pemuda dan Olahraga Daerah Istemewa Yogyakarta | Perlu regenerasi arsiparis dan petugas kearsipan secara merata dan sesuai kuota formasi agar beban kerja yang diberikan tidak terlalu tinggi |

Sri Hasta Mulyani (2014) melakukan penelitian tentang sistem informasi e-document pada badan penjamin mutu akademik universitas Respati Yogyakarta menyatakan bahwa okumen menjadi salah satu faktor penting dalam mengukur tingkat pencapaian kegiatan yang terorganisasi. Pentingnya sebuah informasi yang terkandung dalam dokumen dapat dijadikan acuan dalam pengambilan keputusan, penyusunan kebijakan, perencanaan, regulasi, monitoring atau memantau capaian kegiatan dan mengevaluasi kegiatan. Namun dalam pengelolaan dokumen masih ditemukan beberapa permasalahan, yaitu: seringnya kehilangan arsip/dokumen, kesulitan dalam proses pencarian arsip/dokumen, arsip/dokumen tersimpan dengan tidak teratur dan masih pada unit-unit yang terdapat dalam organisasi intitusi,berakibat pada terlambat dalam pembuatan laporan, membutuhkan waktu yang lama dalam pencarian arsip/dokumen, membutuhkan banyak tempat untuk penyimpanan arsip dan dokumen. Sistem dikembangkan dengan perangkat lunak pengembangan yaitu PHP, Drupal 7.20 dan Database Management System dan MySQL. Hasil dari penelitian diharapkan membantu perguruan tinggi dalam pengelolaan dokumen yang lebih baik dengan adanya penyimpanan salinan fisik dokumen ke dalam media elektronik, mempercepat proses pencarian dokumen yang dilakukan secara elektronik dengan akses melalui internet, efesiensi dalam penyediaan sarana dan prasarana dalam pentimpanan dokumen.

Reza Gama Hidayat (2017) melakukan penelitian tentang rancang bangun sistem informasi manajemen kearsipan pemerintah daerah istimewa Yogyakarta berbasis web dengan metode scrum menyatakan bahwa sistem informasi yang telah dibuat berhasil menggunakan metode scrum. Metode scrum sendiri adalah metode pengembangan perangkat lunak. Penelitian tersebut melalui tiga tahap pengujian yaitu pengujian *alpha,* pengujian *beta* dan pengujian fungsional. Pengujian *beta* dilakukan oleh peneliti dan staff kearsipan Badan Perpustakaan dan Arsip Daerah sedangkan hasil dari pengujian fungsional ada dua poin yang diambil, pertama adalah sistem informasi dapat berjalan baik menurut sebagian besar penguji adalah sebesar 70%. Dan yang kedua adalah penelitian ini tidak ada kesalahan sebesar 100% pada sistem informasi yang telah berjalan.

Aji Prihartantyasto (2016) melakukan penelitian tentang Perancangan Sistem Informasi Pengelohan Arsip Surat Masuk dan Surat Keluar menggunakan *Framework CodeIgniter* dengan studi kasus Disperindagko dan UMKM Salatiga menyatakan bahwa sistem yang dibangun untuk mempermudah pengarsipan surat masuk dan surat keluar pada lingkungan Disperindagko dan UMKM Salatiga. Salah satu kemudahan yang didapat adalah kepala dinas dapat melakukan disposisi dimanapun berada walaupun tidak dalam lingkungan kerja. Sistem ini dibuat berbasis web dengan menggunakan *framework codeigniter*. Kelebihan *framework* ini adalah banyak *library* yang digunakan untuk membangun sistem berjalan dengan cepat dan mudah. Untuk pengujian sistem menggunakan pengujian blackbox dan pengujian ke pengguna langsung menggunakan kuisioner. Hasil dari pengujian blackbox adalah setiap fungsi *valid*. Dan kesimpulan dari hasil pengujian pengguna melalui kuisioner melalui 6 pertanyaan adalah rata-rata mengatatakan sangat setuju sistem berjalan dengan sesuai.

Rina Suyuti (2013) melakukan penelitian tentang implementasi sistem kearsipan elektronik di sekretariat daerah dan dinas pendidikan pemuda dan olahraga Daerah Istemewa Yogyakarta menyatakan bahwa sistem yang dibuat berbasiskan web dengan jaringan intranet dan internet dan sistem diberi nama Sisminkada. Sistem kearsipan elektronik ini didukung dengan penggunaan perangkat keras dan perangkat lunak. Adapun proses dalam sistem kearsipan elektronik tersebut yaitu proses pencatatan, pengendalian, pendistribusian, penyimpanan dan penyusutan serta merupakan hasil sosialisasi dari sub instansi yang terkait di Sekretariat daerah yang lebih fokus pada proses *input* surat masuk dan surat keluar.

## Profil Objek Penelitian

Universitas Islam Kalimantan Muhammad Arsyad Al Banjari adalah PTS tertua dan terbesar di Kalimantan Selatan, didirikan oleh yayasan Universitas tersebut berdasarkan akte Notaris Bachtiar Nomor 6 tanggal 7 Juli 1981. Saat ini telah memiliki Program PACASARJANA, 9 (Sembilan) Fakultas, 17 (Tujuh Belas) Program Studi S-1. Berdasarkan Surat Keputusan Direktur Jenderal Pendidikan Tinggi Departemen Pendidikan Nasional Nomor : 1063/D/T/2008 tanggal 23 April 2008 tentang Ijin Penyelenggaraan Program-Program Studi Baru pada Universitas Islam Kalimantan Muhammad Arsyad Al Banjary Banjarmasin, dimana UNISKA diberikan ijin untuk menyelenggarakan program studi baru yaitu S1 Agribisnis dengan SK Nomor 4024/D/T/K-XI/2010 tanggal 18 Oktober 2010 dan Fakultas Hukum dengan konsentrasi S1 Ilmu Hukum berdasarkan SK Nomor 4025/D/T/K-XI/2010 tanggal 18 Oktober 2010 serta S.1 Program Studi Informatika dengan Surat Keputusan Nomor  4808/D/T/K-XI/2010 tanggal 10 Desember 2010. Selanjutnya pada tanggal 22 Oktober 2009 kembali UNISKA diberikan ijin untuk menyelenggarakan Program Studi Ekonomi Syariah berdasarkan Surat Ijin dari Dirjen Dikti No.: D.dj.I/614/2009 dan Program Studi Kimia sesuai dengan SK Kepmendiknas No. 204/E/O/2011 tanggal 21/09/2011.

Pada tahun 2015 Universitas Islam Kalimantan memperoleh akreditasi B dari BAN-PT. Kemudian disusul dengan 3 izin program studi baru, yaitu: 1) S1 Pendidikan Guru Madrasah Ibtidaiya (PGMI) di bawah naungan Fakultas Studi Islam dengan terbitnya Surat keputusan Dirjen Pendidikan Islam No. 361 Tanggal 20 Januari 2015, 2) S1 Sistem Informasi di bawah naungan Fakultas Teknologi Informasi berdasarkan SK Kemenristek No. 427/M/Kp/VII/ 2015, dan 3) S1 Teknik Sipil di bawah naungan Fakultas Teknik berdasarkan SK Kemenristek No. 431/M/Kp/VII/2015. Fakultas Teknologi Informasi memiliki dua program studi yaitu Teknik Informatika dan Sistem Informasi dengan SK BAN-PT No. 030/BAN-PT/Ak-XV/S1/X/2012 pada tanggal 18 Oktober 2012. Sedangkan program studi Sistem Informasi.

# BAB III ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini membahas tentang identifikasi permasalahan, analisis permasalahan, solusi permasalahan dan perancangan sistem dalam membangun sistem pengarsipan dokumen akademik mahasiswa pada Fakultas Teknologi Informasi. Adapun teknik dalam pengampulan data yaitu dengan cara melakukan wawancara dan observasi. Dari proses pengumpulan data ini akan didapatkan proses identifikasi dan analisis permasalahan.

## Analisis Sistem yang berjalan

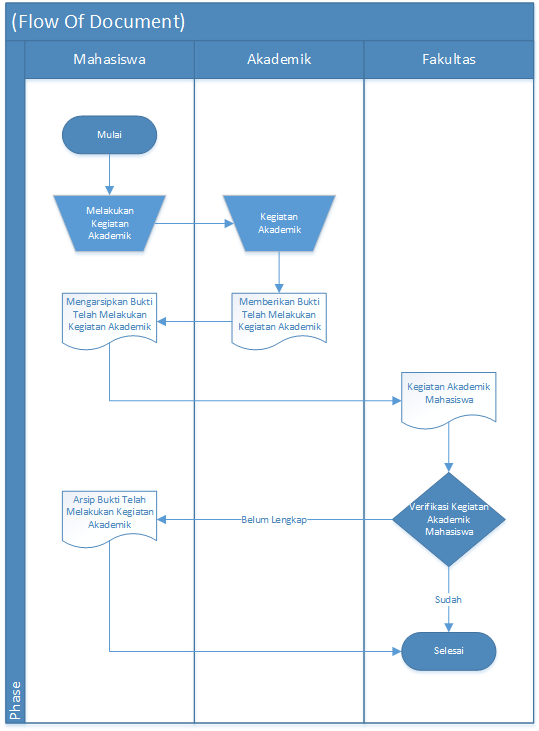
Identifikasi permasalahan dilakukan setelah proses wawancara dan observasi. Dari proses mengumpulkan data tersebut maka didapatkan suatu titik temu dari permasalahan yang ada. Proses identifikasi dilakukan dengan cara menganalisis setiap proses dari mahasiswa dalam melakukan proses penyimpanan dokumen akademik.

1. Input

Pada proses pengarsipan dokumen – dokumen akademik dilakukan dengan cara penyimpana berkas secara langsung.

1. Proses

Berkas Dokumen Akademik yang didapatkan oleh mahasiswa disimpan dengan penyimpanan secara langsung (dokumen asli disimpan didalam kumpulan dokumen Mahasiswa) dan penyimpanan yang dilakukan di *Hard-Drive, Flash-Drive* maupun media penyimpanan lainnya.



**Gambar 3. 1 Flow Of Document Proses Dokumen Akademik Mahasiswa**

1. Analisa Output

Adapun hasil (keluaran) dari proses ini yaitu berupa *Hard-Copy* Dokumen asli yang kemudian disimpan secara langsung oleh Mahasiswa.

## Analisis Kebutuhan Sistem

Tahapan ini merupakan tahapan yang dilakukan untuk menganalisis sistem secara lebih detail, baik proses, prosedur dan fungsi sesuai dengan data yang telah dikumpulkan. Di dalam perancangan aplikasi ini, kebutuhan sistem yang digunakan terdiri dari perangkat keras (*hardware)* dan perangkat lunak (*software*).

### Kebutuhan *Hardware*

*Hardware* minimum untuk pembuatan Sistem Pengarsipan Digitan Dokumen Akademik Mahasiswa pada Fakultas Teknologi Informasi Universitas Islam Kalimantan Banjarmasin ini adalah sebagai berikut :

1. *Server Processor* intel Pentium 4
2. *Harddisk* 500Gb SATA
3. *Memory* 1Gb
4. **Kebutuhan Software**

Sistem Kearsipan Digital Dokumen Akademik Mahasiswa pada Fakultas Teknologi Informasi Universitas Islam Kalimantan Banjarmasin ini dibuat menggunakan software sebagai berikut :

1. PHP – *Hypertext Preprocessor*

Bahasa pemrograman yang digunakan dalam pembuatan sistem ini yaitu adalah Bahasa Pemrograman php.

1. MySQL

MySQL digunakan sebagai database untuk sistem.

1. Sublime Text 3

Digunakan sebagai text editor untuk penulisan kode.

1. XAMPP

Sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL Database, dan penerjemah Bahasa yang ditulis dengan Bahasa pemrograman PHP dan Perl.

1. **Analisis Kebutuhan Proses**

*Unified Modelling Language* (UML) adalah Bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan mendokumentasikan dari sebuah sistem pengembangan *software.* UML juga memberikan standar penulisan sebuah *blue print system,* yang meliputi bisnis proses, penulisan kelas – kelas dalam Bahasa program yang spesifik, skema *database,* dan komponen – komponen yang diperlukan dalam *software system.*

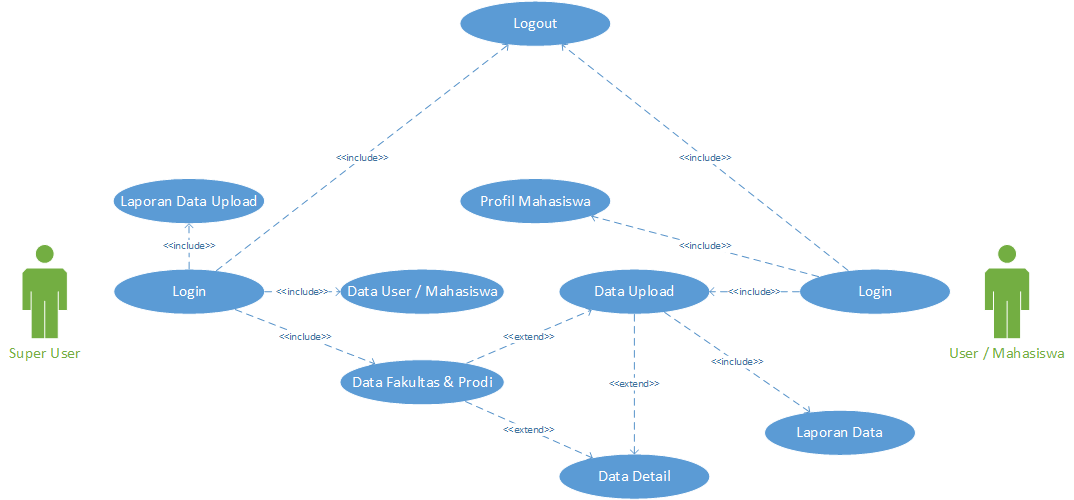
## Rancangan Model Sistem

Tahap ini merupakan tahap perancangan sistem. Desain sistem menggunakan UML (*Unified Model Language*), struktur table dalam *database* dan relasi antar table.

1. ***Use Case Diagram***

Berdasarkan kebutuhan sistem, maka dapat diasumsikan Sistem Pengarsipan Digital Dokumen Akademik Mahasiswa Pada Fakultas Teknologi Informasi Universitas Islam Kalimantan Banjarmasin adalah sebagai berikut :

1. Pihak yang berinteraksi dengan sistem informasi ini adalah User yang terdiri dari Administrator dan Mahasiswa.
2. Administrator dapat mengelola data dan biodata Mahasiswa pada sistem dan mengelola dokumen – dokumen yang telah diupload dan diarsipkan.



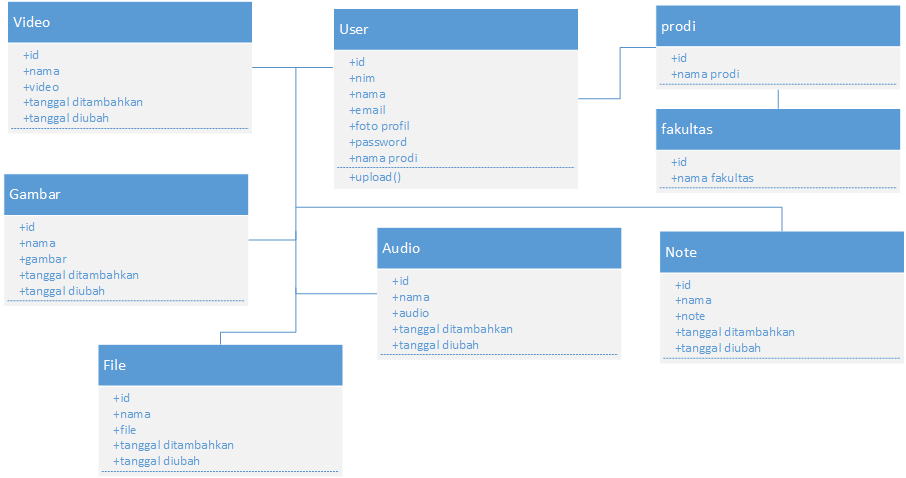
**Gambar 3. 2 Use Case Sistem**

Gambar *use case diagram* menunjukan *use case diagram* dari sistem pengarsipan digital dokumen akademik mahasiswa tersebut mempunyai dua actor yaitu *Admin* dan Mahasiswa.

*Activity diagram* merupakan teknik untuk menggambarkan logika prosedur, proses bisnis, dan jalur kerja. Pada tahap ini perancangan *Activity diagram* dibuat berdasarkan aktifitas yang dilakukan oleh Mahasiswa dan *Administrator.*

1. ***Class Diagram***

*Class* adalah sebuah spesifikasi yang jika instansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/property) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi), berikut gambar *Class Diagram* :



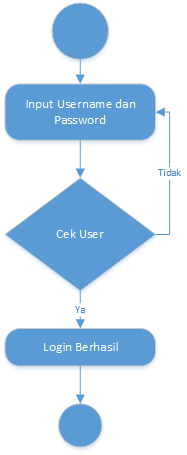
**Gambar 3. 3 Class Diagram Sistem**

1. ***Activity Diagram***

*Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang dirancang, bagaimana masing – masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses parallel yang mungkin terjadi pada beberapa eksekusi. Berikut adalah gambar *Activity diagram* dari sistem yang dirancang, yaitu :

* + - 1. ***Activity Diagram* Data Login**

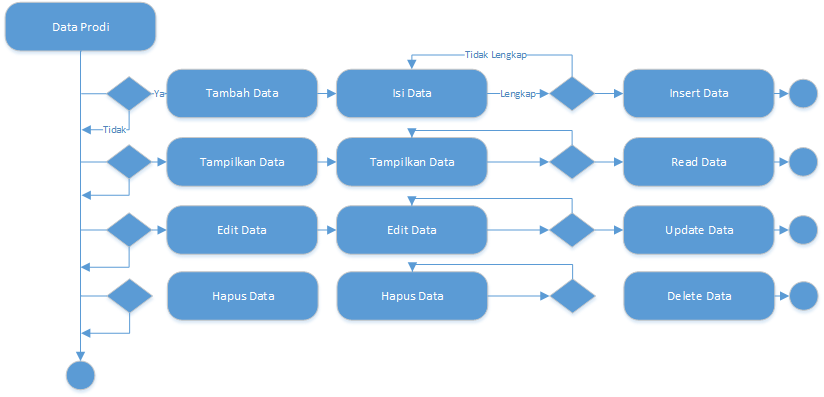
*Activity diagram* data *login* dapat dilihat pada gambar berikut :

**

**Gambar 3. 4 Activity Diagram Login**

* + - 1. ***Activity Diagram* Data Prodi**

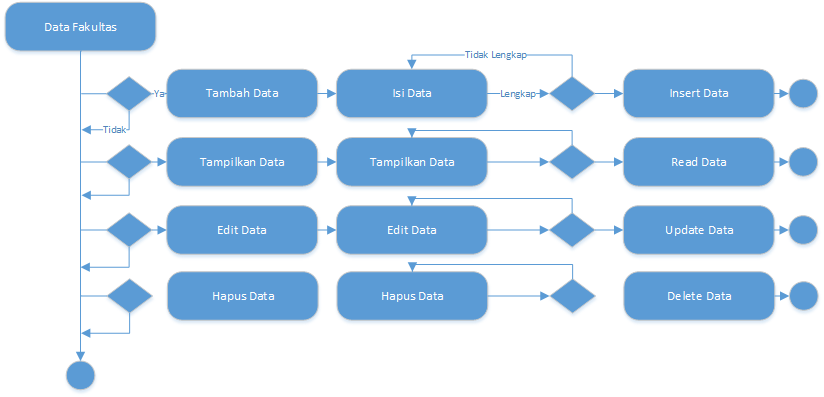
*Activity diagram* dataProdidapat dilihat pada gambar berikut :



**Gambar 3. 5 Activity Diagram Prodi**

* + - 1. ***Activity Diagram* Data Fakultas**

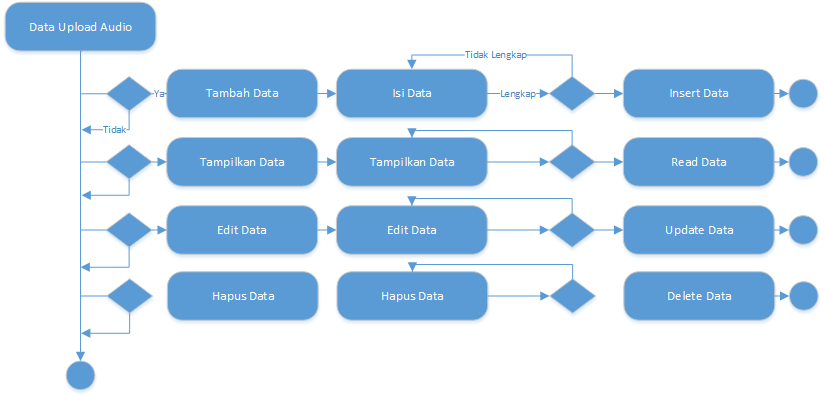
*Activity diagram* dataFakultasdapat dilihat pada gambar berikut :

******

**Gambar 3. 6 Activity Diagram Fakultas**

* + - 1. ***Activity Diagram* Data Upload Audio**

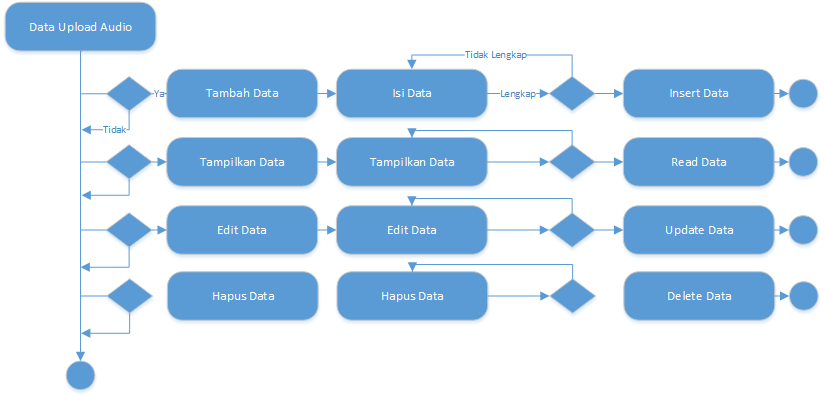
*Activity diagram* upload Audiodapat dilihat pada gambar berikut :

******

**Gambar 3. 7 Activity Diagram Upload Audio**

* + - 1. ***Activity Diagram* Data Upload File**

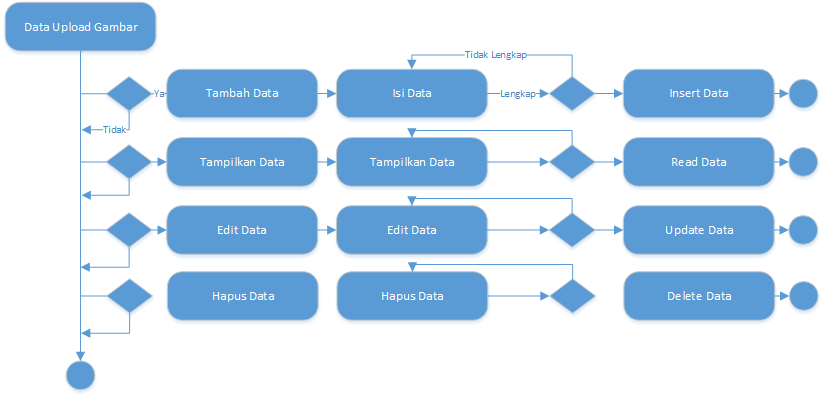
*Activity diagram* upload Filedapat dilihat pada gambar berikut :

******

**Gambar 3. 8 Activity Diagram Upload File**

* + - 1. ***Activity Diagram* Data Upload Gambar**

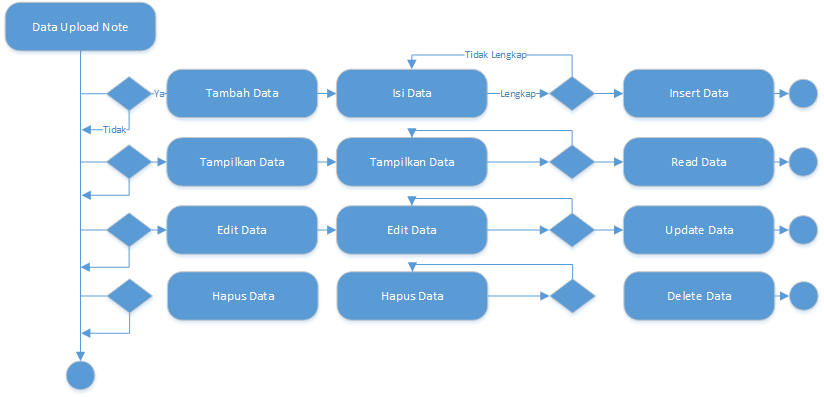
*Activity diagram* dataupload Gambardapat dilihat pada gambar berikut :

******

**Gambar 3. 9 Activity Diagram Upload Gambar**

* + - 1. ***Activity Diagram* Data Upload Note**

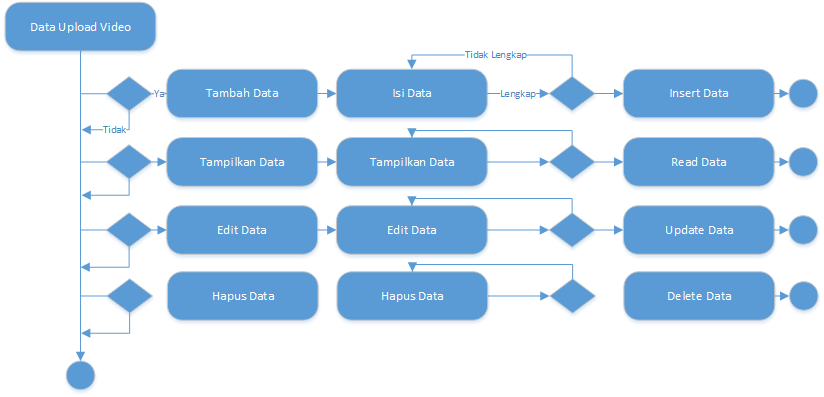
*Activity diagram* dataUpload Notedapat dilihat pada gambar berikut :

******

**Gambar 3. 10 Activity Diagram Upload Note**

* + - 1. ***Activity Diagram* Data Upload Video**

*Activity diagram* data upload Video dapat dilihat pada gambar berikut :

******

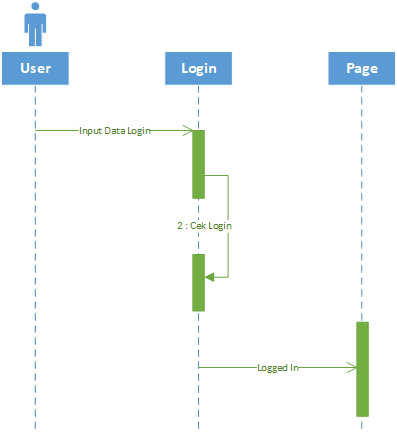
**Gambar 3. 11 Activity Diagram Data Kelas**

1. ***Sequence Diagram***

*Sequence Diagram* menggambarkan perilaku pada sebuah scenario, diagram ini menunjukan sejumlah contoh objek dan *message* (pesan) yang diletakkan diantara objek – objek ini di dalam *use case,* berikut gambar *sequence diagram* :

1. *Sequence Diagram Login*

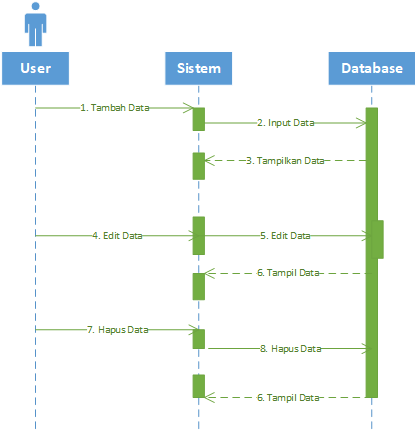
Berikut ini adalah gambaran scenario *sequence diagram login.*

**

**Gambar 3. 12 Sequence Diagram Login**

1. *Sequence Diagram Data Management*

Berikut ini adalah gambaran skenario *sequence diagram* Data*.*

****

**Gambar 3. 13 Sequence Diagram Data Management**

## Rancangan Basis Data

Pada perancangan objek sebelumnya telah dihasilkan diagram kelas yang menghasilkan hubungan antar objek dan kelas – kelas yang terdapat dalam aplikasi. Objek yang telah dihasilkan tersebut, kemudian dinyatakan dalam bentuk tabel – tabel. Pada bagian ini akan dijelaskan tabel – tabel basis data yang digunakan dalam perancangan sistem ini.

1. Tabel user

Tabel user merupakan tabel yang berfungsi untuk menampung data pengguna baik *admin*istrator maupun *user* biasa. Struktur tabel tb\_user dapat dilihat pada Tabel 3.1.

**Tabel 3. 1 Tabel User**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_user | Int (8) | Primary Key |
| 2 | nim | Varchar (10) | Nomor Induk Mahasiswa |
| 3 | nama | Varchar (100) | Nama Lengkap |
| 4 | email | Varchar (100) | Email |
| 5 | foto\_profil | Varchar (100) | Foto Profil |
| 6 | password | Varchar (255) | Password Pengguna |
| 7 | level | Enum (Super\_User, User) | Level Pengguna |
| 8 | id\_prodi | Int (3) | Foreign Key |

1. Tabel prodi

Tabel prodi merupakan tabel yang berfungsi untuk menampung data program studi *user* yang ada pada sistem. Struktur tabel prodi dapat dilihat pada Tabel 3.2.

**Tabel 3. 2 Tabel Prodi**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_prodi | Int (3) | Primary Key |
| 2 | nama\_prodi | Varchar (100) | Nama Program Studi |
| 3 | id\_fakultas | Int (3) | Foreign Key |

1. Tabel fakultas

Tabel fakultas merupakan tabel yang berfungsi untuk menyimpan data fakultas yang di masukkan oleh Super\_User. Struktur tabel fakultas dapat dilihat pada Tabel 3.3.

**Tabel 3. 3 Tabel Fakultas**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_fakultas | Int (3) | Primary Key |
| 2 | Nama\_fakultas | Varchar (100) | Nama Fakultas |

1. Tabel note

Tabel note merupakan tabel untuk menyimpan data catatan yang di upload oleh User maupun Super\_User. Struktur tabel note dapat dilihat pada Tabel 3.4.

**Tabel 3. 4 Tabel Note**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_note | Int (3) | Primary Key |
| 2 | note | Varchar (100) | Nama Kelas |
| 3 | date | date | Tanggal Catatan |
| 4 | added | timestamp | Waktu Ditambahkan |
| 5 | updated | timestamp | Waktu Diperbarui |
| 6 | id\_user | Int(8) | Foreign Key |

1. Tabel u\_audio

Tabel u\_audio merupakan tabel untuk penyimpanan data upload berupa file dengan bentuk audio. Struktur tabel u\_audio dapat dilihat pada Tabel 3.5.

**Tabel 3. 5 Tabel u\_audio**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_audio | Int (3) | Primary Key |
| 2 | nama\_audio | Varchar (100) | Nama Kelas |
| 3 | audio | date | Tanggal Catatan |
| 4 | added | timestamp | Waktu Ditambahkan |
| 5 | updated | timestamp | Waktu Diperbarui |
| 6 | id\_user | Int(8) | Foreign Key |

1. Tabel u\_file

Tabel u\_file merupakan tabel untuk penyimpanan data upload berupa file dengan bentuk dokumen (pdf) dan lain sebagainya. Struktur tabel u\_file dapat dilihat pada Tabel 3.5.

**Tabel 3. 6 Tabel u\_file**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_file | Int (3) | Primary Key |
| 2 | file | Varchar (100) | Nama Kelas |
| 3 | date | date | Tanggal Catatan |
| 4 | added | timestamp | Waktu Ditambahkan |
| 5 | updated | timestamp | Waktu Diperbarui |
| 6 | id\_user | Int(8) | Foreign Key |

1. Tabel u\_gambar

Tabel u\_gambar merupakan tabel untuk penyimpanan data upload berupa gambar dengan bentuk (.jpg, .jpeg, .png). Struktur tabel u\_gambar dapat dilihat pada Tabel 3.5.

**Tabel 3. 7 Tabel u\_gambar**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_note | Int (3) | Primary Key |
| 2 | note | Varchar (100) | Nama Kelas |
| 3 | date | date | Tanggal Catatan |
| 4 | added | timestamp | Waktu Ditambahkan |
| 5 | updated | timestamp | Waktu Diperbarui |
| 6 | id\_user | Int(8) | Foreign Key |

1. Tabel u\_video

Tabel u\_video merupakan tabel untuk penyimpanan data upload yang berupa file dengan bentuk video. Struktur tabel u\_video dapat dilihat pada Tabel 3.5.

**Tabel 3. 8 Tabel u\_audio**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama | Tipe | Keterangan |
| 1 | id\_video | Int (3) | Primary Key |
| 2 | nama\_video | Varchar (100) | Nama Kelas |
| 3 | video | date | Tanggal Catatan |
| 4 | added | timestamp | Waktu Ditambahkan |
| 5 | updated | timestamp | Waktu Diperbarui |
| 6 | id\_user | Int(8) | Foreign Key |

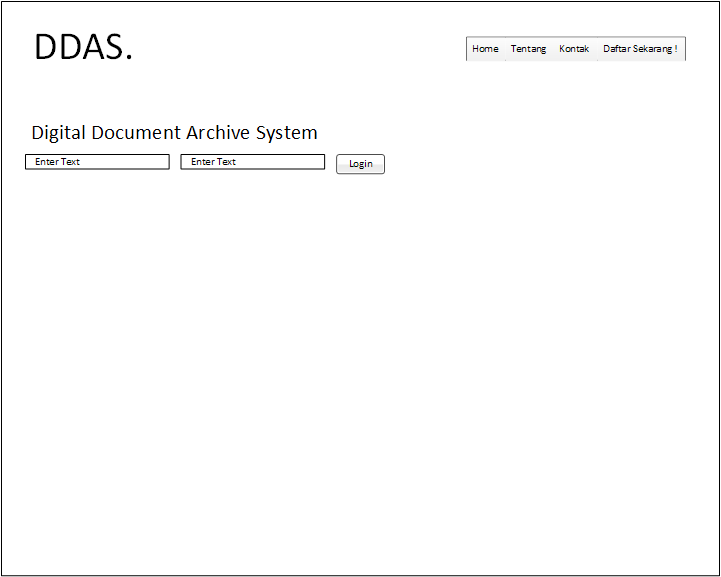
## Rancangan Antarmuka Masukan Sistem

*User Interface* sangatlah penting dalam suati aplikasi karena merupakan bagian dari perangkat lunak yang menjadi sarana komunikasi antar pengguna dengan sistem serta dapat memberikan kemudahan bagi pengguna dalam menggunakan fitur yang disediakan sistem.

Desain Masukan *(input)* adalah rancangan *form* masukan *(input)* yang digunakan dalam pembuatan sistem ini.

1. **Rancangan Antarmuka Halaman Utama**

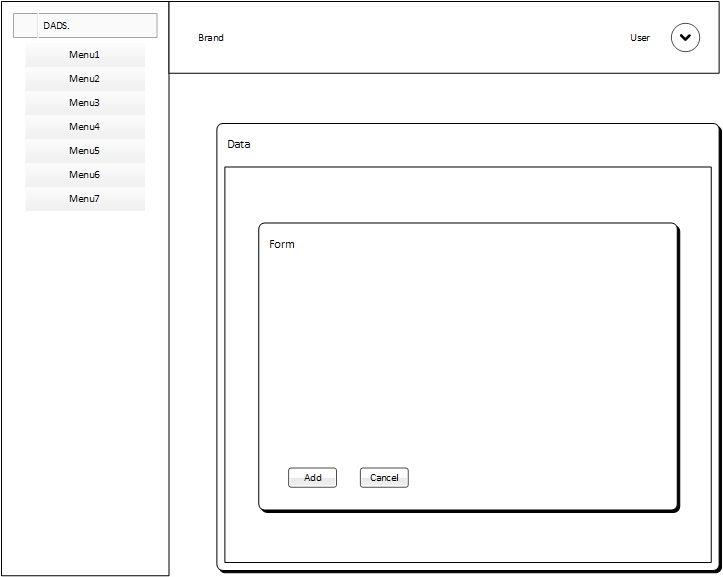
Rancangan antarmuka merupakan tampilan utama dari sistem yang dapat dilihat oleh Mahasiswa dan *Administrator.* Gambaran rancangan antarmuka halaman utama sistem dapat dilihat pada Gambar 3.11.



**Gambar 3. 14 Halaman Utama Sistem**

1. **Rancangan Antarmuka Masukan Data**

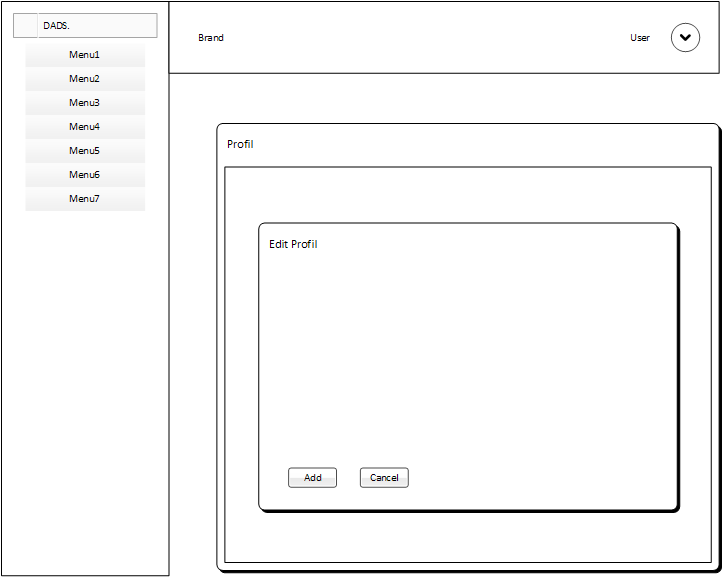
Rancangan antarmuka masukan data merupakan tampilan dari sistem yang dapat dilihat oleh Mahasiswa dan *Administrator* ketika hendak melakukan proses pengolahan data *(create, read, update, delete).* Gambaran rancangan antarmuka halaman utama sistem dapat dilihat pada Gambar 3.11.



**Gambar 3. 15 Halaman Masukan Data**

1. **Rancangan Antarmuka *Edit* Profil *User***

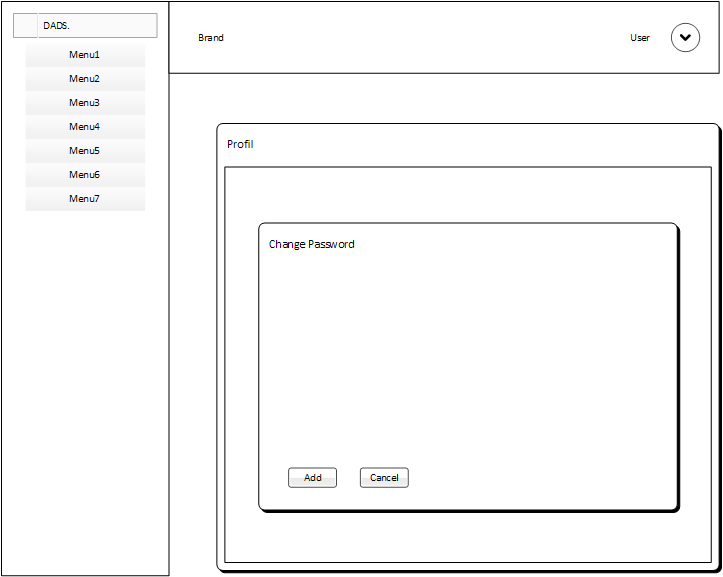
Rancangan antarmuka edit profil merupakan tampilan saat *user* berhasil melakukan proses *login* dan ingin melakukan perbaruan profil*.* Gambaran rancangan antarmuka Edit Profil *User* dapat dilihat pada Gambar 3.13.



**Gambar 3. 16 Antarmuka Edit Profil User**

1. **Rancangan Antarmuka *Edit Password***

Rancangan antarmuka *Edit Password* merupakan tampilan dari sistem yang dapat dilihat oleh Mahasiswa dan *Administrator* ketika hendak melakukan proses perbaruan *Password.* Gambaran rancangan antarmuka *Data Management* sistem dapat dilihat pada Gambar 3.14.



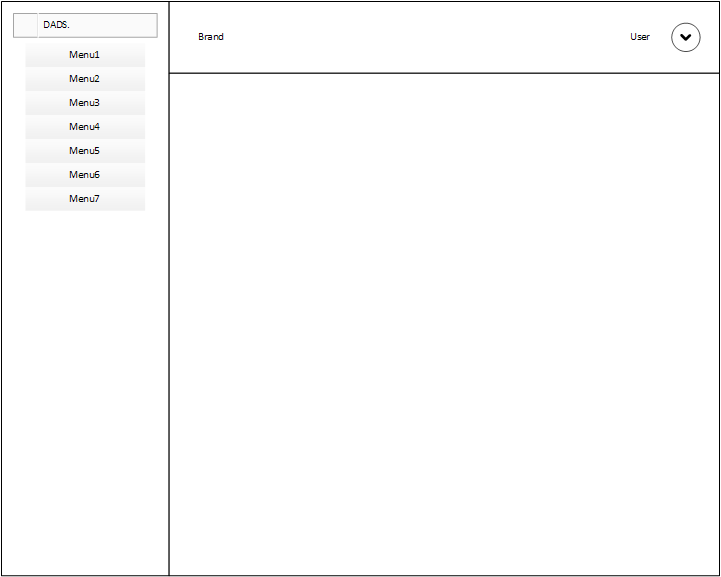
**Gambar 3. 17 Antarmuka Edit Password**

## Rancangan Antarmuka Keluaran Sistem

Berikut ini adalah perancangan hasil keluaran *(output)* dari pengolahan data yang ada pada sistem ini.

1. **Rancangan Antarmuka *Dashboard***

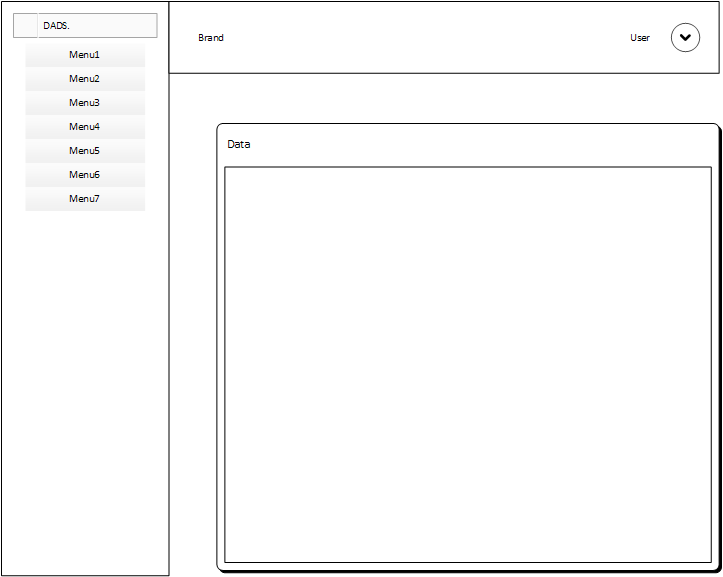
Rancangan antarmuka *Dashboard* merupakan tampilan dari sistem yang dapat dilihat oleh Mahasiswa dan *Administrator* setelah melakukan proses *login.* Gambaran rancangan antarmuka *Dashboard* sistem dapat dilihat pada Gambar 3.15.



**Gambar 3. 18 Antarmuka Dashboard**

1. **Rancangan Antarmuka *Data Management***

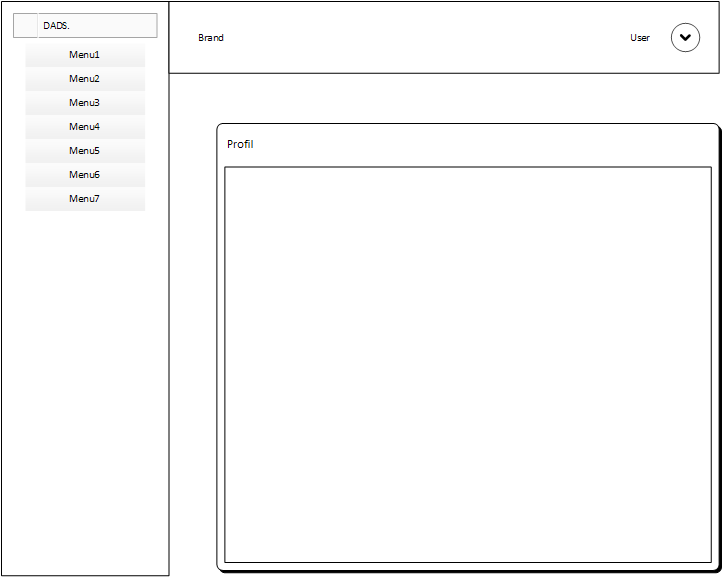
Rancangan antarmuka *Data Creation* merupakan tampilan dari sistem yang dapat dilihat oleh Mahasiswa dan *Administrator* saat akan melakukan proses pengolahan data *(create, read, update, delete).* Gambaran rancangan antarmuka *Data Creation* dapat dilihat pada Gambar 3.16.



**Gambar 3. 19 Antarmuka Data Management**

1. **Rancangan Antarmuka Profil**

Rancangan antarmuka Profilmerupakan tampilan dari sistem yang dapat dilihat oleh Mahasiswa dan *Administrator* saat melihat Profil *User.* Gambaran rancangan antarmuka Profil dapat dilihat pada Gambar 3.17.



**Gambar 3. 20 Antarmuka Profil**

1. **Rancangan Antarmuka Halaman *Help***

Rancangan antarmuka Halaman *Help* merupakan tampilan dari sistem ketika *User* membuka tampilan *Help.* Gambaran rancangan antarmuka *Help* dapat dilihat pada Gambar 3.18.



**Gambar 3. 21 Antarmuka Halaman Help**

## Tempat dan Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Uraian Kegiatan** | **Bulan** | | | | **Bulan** | | | | **Bulan** | | | | **Bulan** | | | | **Bulan** | | | | **Bulan** | | | |
|  |  | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| 1 | Persiapan Penelitian |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Analisis Permasalahan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Pembuatan Aplikasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Testing dan Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Dokumentasi dan Penulisan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# BAB IV IMPLEMENTASI DAN PENGUJIAN

## Spesifikasi Sistem

Pada sub bab sfesifikasi rancangan ini akan di bahas mengenai spesifikasi perangkat lunak dan spesifikasi perangkat keras.

1. **Spesifikasi Perangkat Lunak**

Spesifikasi perangkat lunak yang digunakan pada saat implementasi program, yaitu :

1. Sistem Operasi Windows 10
2. Xampp Control Panel v3.2.2
3. phpMyAdmin v4.7.0
4. Sublime Text 3
5. Brackets Text Editor
6. **Spesifikasi Perangkat Keras**

Spefikasi perangkat keras yang digunakan pada saat implementasi program, aplikasi ini adalah :

1. Laptop dengan *CPU* Amd A8
2. *Memory* 4gb
3. Kapasitas *Harddisk 500gb*

## Langkah – Langkah Pembuatan Sistem

Pada tahap ini dilaksanakan implementasi dari rancangan – rancangan, baik rancangan basis data, aplikasi maupun rancangan tampilan.

1. **Bahasa Pemrograman dan Komponen**

Bahasa pemrograman dan komponen – komponen yang digunakan dalam pengembangan sistem ini adalah :

1. Application server : Apache 2.4.26 (Win32)
2. PHP *version* 5.6.31
3. MariaDB *version* 10.1.25
4. *Framework* : Codeigniter 3.1.8
5. *CSS Bootstraps* V.4.0
6. **Struktur File**

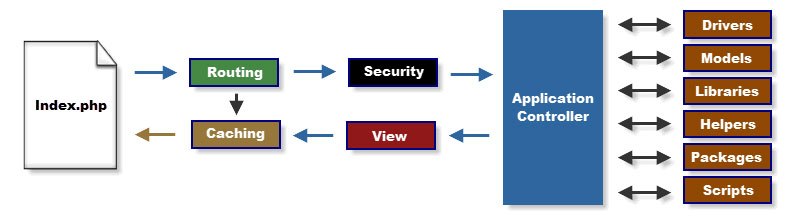
Karena aplikasi ini merupakan penerapan dari Codeigniter *framework*, maka dalam penyusunan struktur file juga harus mengikuti standar baku yang dimiliki Codeigniter itu sendiri.

**Tabel 4. 1 Tabel Struktur File**

|  |  |  |
| --- | --- | --- |
| No | Struktur file | Keterangan |
| 1 | ddas/application/config | Konfigurasi aplikasi, *server*, dan *database* |
| 2 | ddas/application/controllers | *Source code* logika control aplikasi |
| 3 | ddas/application/models | *Source code* fungsi – fungsi pengelolaan struktur data |
| 4 | ddas/application/views | *Source code* tampilan yang ada pada aplikasi |
| 5 | ddas/application/libraries | *Source code library* tambahan |

1. **Alur Kerja Codeigniter *Framework***

Secara umum alur kerja *framework* Codeigniter adalah sebagai berikut :



**Gambar 4. 1 Alur Kerja Framework Codeigniter**

*Index.php* merupakan *front controller* atau file yang berhubungan langsung dengan *user.* Saat *user* membuka *index.php* dan melakukan *request,* maka akan dicek melalui *router* alamat yang dimasukkan oleh *user* tersebut. Bila ada pada *cahche* data, maka akan langsung menuju ke *Caching* dan kembali lagi ke *user.* Bila pada *Cache* tidak ada maka akan diteruskan ke *security* untuk memvalidasi, menyaring, dan mengamankan informasi yang dimasukkan *user* (bila ada). Lalu dari *security* informasi tersebut dikirim ke *application controller.* Pada *application controller,* informasi dan data akan diolah. Bila diperlukan, *application controller* akan berhubungan dengan *models, libraries, helpers, plugins,* atau skrip lainnya. *Application controller* akan mengirimkan respon ke *view* untuk ditampilkan kepada *user. View* akan disimpan di *cache* bila berikutnya melakukan *request* yang sama. Sehingga mempercepat proses. Setelah di proses di *cache,* maka akan ditampilkan informasi kepada *user.*

Dalam implementasinya pada sistem kearsipan digital dokumen akademik mahasiswa dapat diuraikan sebagai berikut :

* 1. **Routing**

Ketika pertama kali mengakses aplikasi melalui *web browser*, maka halaman yang pertama kali dipanggil adalah halam *index.php* pada folder aplikasi. Setelah itu file *index.php* melakukan *routing* ke *controller* yang telah di *config* oleh *programmer* pada file ddas/application/config/routes.php dimana isinya adalah sebagai berikut :

<?php

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

$route['default\_controller'] = 'user';

$route['404\_override'] = 'page404';

$route['translate\_uri\_dashes'] = FALSE;

Dari code diatas, maka sistem akan menjalankan *controller* User.php yang berisi :

<?php

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

class User extends CI\_Controller {

public function \_\_construct() {

parent::\_\_construct();

$this->load->model('User\_model');

}

public function index() {

if ($this->session->userdata('is\_login')==TRUE) {

redirect('user/dashboard/','refresh');

} else {

$this->load->view('landing');

}

}

public function login() {

$this->load->view('landing');

}

public function blank() {

$data = $this->User\_model->get\_user\_data();

$this->load->view('user/blank', $data);

}

public function register() {

$this->load->view('register');

}

public function help() {

$data = $this->User\_model->get\_user\_data();

$this->load->view('user/help', $data);

}

Dari *code controller* User.php pada baris *function index(),* sistem memanggil file ddas/application/views/landing.php yang berisi :

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="description" content="">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<!-- The above 4 meta tags \*must\* come first in the head; any other head content must come \*after\* these tags -->

<!-- Title -->

<title>DDAS. App - Landing Page</title>

<!-- Favicon -->

<link rel="icon" href="<?php echo base\_url('assets');?>/landing/img/core-img/favicon.ico">

<!-- Core Stylesheet -->

<link href="<?php echo base\_url('assets');?>/landing/style.css" rel="stylesheet">

<!-- Responsive CSS -->

<link href="<?php echo base\_url('assets');?>/landing/css/responsive.css" rel="stylesheet">

</head>

<body>

<!-- Preloader Start -->

<div id="preloader">

<div class="colorlib-load"></div>

</div>

<!-- \*\*\*\*\* Header Area Start \*\*\*\*\* -->

<header class="header\_area animated">

<div class="container-fluid">

<div class="row align-items-center">

<!-- Menu Area Start -->

<div class="col-12 col-lg-10">

<div class="menu\_area">

<nav class="navbar navbar-expand-lg navbar-light">

<!-- Logo -->

<a class="navbar-brand" href="#">DDAS.</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#ca-navbar" aria-controls="ca-navbar" aria-expanded="false" aria-label="Toggle navigation"><span class="navbar-toggler-icon"></span></button>

<!-- Menu Area -->

<div class="collapse navbar-collapse" id="ca-navbar">

<ul class="navbar-nav ml-auto" id="nav">

<li class="nav-item active"><a class="nav-link" href="#home">Home</a></li>

<li class="nav-item"><a class="nav-link" href="#about">Tentang</a></li>

<li class="nav-item"><a class="nav-link" href="#contact">Kontak</a></li>

</ul>

<div class="sing-up-button d-lg-none">

<a href="<?php echo base\_url('user/register') ?>">Daftar Sekarang !</a>

</div>

</div>

</nav>

</div>

</div>

<!-- Signup btn -->

<div class="col-12 col-lg-2">

<div class="sing-up-button d-none d-lg-block">

<a href="<?php echo base\_url('user/register') ?>">Daftar Sekarang !</a>

</div>

</div>

</div>

</div>

</header>

<!-- \*\*\*\*\* Header Area End \*\*\*\*\* -->

<!-- \*\*\*\*\* Wellcome Area Start \*\*\*\*\* -->

<section class="wellcome\_area clearfix" id="home">

<div class="container h-100">

<div class="row h-100 align-items-center">

<div class="col-12 col-md">

<div class="wellcome-heading">

<h2>Digital Documents Archive System</h2>

<h3>D</h3>

<p>Semua file yang anda perlukan, Upload, Simpan, Selesai, Gunakan, Kapanpun dan Dimanapun</p>

<div class="get-start-area">

<?php if($this->session->flashdata('pesan') == TRUE): ?>

<div class="alert alert-danger alert-dismissible" role="alert">

<button type="button" class="close" data-dismiss="alert" aria-label="Close"><span aria-hidden="true">&times;</span></button><?php echo $this->session->flashdata('pesan'); ?>

</div>

<?php endif; ?>

<?php if($this->session->flashdata('sukses') == TRUE): ?>

<div class="alert alert-success alert-dismissible" role="alert">

<button type="button" class="close" data-dismiss="alert" aria-label="Close"><span aria-hidden="true">&times;</span></button><?php echo $this->session->flashdata('sukses'); ?>

</div>

<?php endif; ?>

<!-- Form Start -->

<form action="<?php echo base\_url('user/proses\_login') ?>" method="post" class="form-inline">

<input type="email" name="email" class="form-control mb-2 mr-1 ml-1" placeholder="Alamat Email Anda">

<input type="password" name="password" class="form-control mb-2 mr-1 ml-1" placeholder="Password anda">

<input type="submit" class="submit" value="Login">

</form>

<!-- Form End -->

</div>

</div>

</div>

</div>

</div>

</section>

<!-- \*\*\*\*\* Wellcome Area End \*\*\*\*\* -->

<!-- \*\*\*\*\* Special Area Start \*\*\*\*\* -->

<section class="special-area bg-white section\_padding\_100" id="about">

<div class="container">

<div class="row">

<div class="col-12">

<!-- Section Heading Area -->

<div class="section-heading text-center">

<h2>Tentang Aplikasi</h2>

<div class="line-shape"></div>

</div>

</div>

</div>

<div class="row">

<!-- Single Special Area -->

<div class="col-12 col-md-4">

<div class="single-special text-center wow fadeInUp" data-wow-delay="0.2s">

<div class="single-icon">

<i class="ti-mobile" aria-hidden="true"></i>

</div>

<h4>User Friendly</h4>

<p>Sistem dibangun menggunakan template bootstrap V.4 yang responsive sehingga memudahkan akses pada berbagai perangkat.</p>

</div>

</div>

<!-- Single Special Area -->

<div class="col-12 col-md-4">

<div class="single-special text-center wow fadeInUp" data-wow-delay="0.4s">

<div class="single-icon">

<i class="ti-ruler-pencil" aria-hidden="true"></i>

</div>

<h4>Arsitektur Sistem</h4>

<p>Sistem dibuat menggunakan Framework Codeigniter yang membuat fungsi menjadi lebih cepat dan mudah digunakan.</p>

</div>

</div>

<!-- Single Special Area -->

<div class="col-12 col-md-4">

<div class="single-special text-center wow fadeInUp" data-wow-delay="0.6s">

<div class="single-icon">

<i class="ti-settings" aria-hidden="true"></i>

</div>

<h4>User Preference</h4>

<p>Sistem dibuat dengan menyesuaikan kebutuhan pengguna, yaitu <b>Mahasiswa</b> maupun <b>Dosen</b>.</p>

</div>

</div>

</div>

</div>

</section>

<!-- \*\*\*\*\* Special Area End \*\*\*\*\* -->

<!-- \*\*\*\*\* App Screenshots Area Start \*\*\*\*\* -->

<section class="app-screenshots-area bg-white section\_padding\_0\_100 clearfix" id="screenshot">

<div class="container">

<div class="row">

<div class="col-12 text-center">

<!-- Heading Text -->

<div class="section-heading">

<h2>Screenshot Aplikasi</h2>

<div class="line-shape"></div>

</div>

</div>

</div>

</div>

<div class="container-fluid">

<div class="row">

<div class="col-12">

<!-- App Screenshots Slides -->

<div class="app\_screenshots\_slides owl-carousel">

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (1).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (2).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (3).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (4).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (5).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (6).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (7).png" alt="">

</div>

<div class="single-shot">

<img src="<?php echo base\_url('assets');?>/landing/img/scr-img/app (1).png" alt="">

</div>

</div>

</div>

</div>

</div>

</section>

<!-- \*\*\*\*\* App Screenshots Area End \*\*\*\*\*====== -->

<!-- \*\*\*\*\* CTA Area Start \*\*\*\*\* -->

<section class="our-monthly-membership section\_padding\_50 clearfix">

<div class="container">

<div class="row align-items-center">

<div class="col-md-8">

<div class="membership-description">

<h2>Daftarkan Akun Sebagai Dosen?</h2>

<p><strong>Note :</strong> Bagi Dosen yang ingin mendaftarkan akun harap hubungi Administrator terlebih dahulu untuk membuat akun dengan fitur dosen.</p>

</div>

</div>

<div class="col-md-4">

<div class="get-started-button wow bounceInDown" data-wow-delay="0.5s">

<a href="#">Daftar?</a>

</div>

</div>

</div>

</div>

</section>

<!-- \*\*\*\*\* CTA Area End \*\*\*\*\* -->

<!-- \*\*\*\*\* Contact Us Area Start \*\*\*\*\* -->

<section class="footer-contact-area section\_padding\_100 clearfix" id="contact">

<div class="container">

<div class="row">

<div class="col-md-6">

<!-- Heading Text -->

<div class="section-heading">

<h2>Kontak Kami Segera!</h2>

<div class="line-shape"></div>

</div>

<div class="footer-text">

<p>Silahkan hubungi Administrator untuk dengan mengisi form disamping ini.</p>

</div>

<div class="address-text">

<p><span>Alamat :</span> Jl. HKSN Komplek. AMD Kayutangi Banjarmasin</p>

</div>

<div class="phone-text">

<p><span>Telepon :</span> +62-853-4788-9990</p>

</div>

<div class="email-text">

<p><span>Email:</span> rifqi.mulyawan@gmail.com</p>

</div>

</div>

<div class="col-md-6">

<!-- Form Start-->

<div class="contact\_from">

<form action="mailto:rifqi.mulyawan@gmail.com" method="post" enctype="text/plain">

<!-- Message Input Area Start -->

<div class="contact\_input\_area">

<div class="row">

<!-- Single Input Area Start -->

<div class="col-md-12">

<div class="form-group">

<input type="text" class="form-control" name="name" id="name" placeholder="Nama Anda" required>

</div>

</div>

<!-- Single Input Area Start -->

<div class="col-md-12">

<div class="form-group">

<input type="email" class="form-control" name="email" id="email" placeholder="Alamat Email Anda" required>

</div>

</div>

<!-- Single Input Area Start -->

<div class="col-12">

<div class="form-group">

<textarea name="message" class="form-control" id="message" cols="30" rows="4" placeholder="Pesan Anda \*" required></textarea>

</div>

</div>

<!-- Single Input Area Start -->

<div class="col-12">

<button type="submit" class="btn submit-btn">Kirim Sekarang !</button>

</div>

</div>

</div>

<!-- Message Input Area End -->

</form>

</div>

</div>

</div>

</div>

</section>

<!-- \*\*\*\*\* Contact Us Area End \*\*\*\*\* -->

<!-- \*\*\*\*\* Footer Area Start \*\*\*\*\* -->

<footer class="footer-social-icon text-center section\_padding\_70 clearfix">

<!-- footer logo -->

<div class="footer-text">

<h2>DADS.</h2>

<p>Digital Archive Documents System | DADS</p>

</div>

<!-- social icon-->

<div class="footer-social-icon">

<a href="#"><i class="fa fa-facebook" aria-hidden="true"></i></a>

<a href="#"><i class="active fa fa-twitter" aria-hidden="true"></i></a>

<a href="#"> <i class="fa fa-instagram" aria-hidden="true"></i></a>

<a href="#"><i class="fa fa-google-plus" aria-hidden="true"></i></a>

</div>

<div class="footer-menu">

<nav>

<ul>

<li><a href="#">Tentang</a></li>

<li><a href="#">Syarat &amp; Kondisi</a></li>

<li><a href="#">Kebijakan Pribadi</a></li>

</ul>

</nav>

</div>

<!-- Foooter Text-->

<div class="copyright-text">

<!-- \*\*\*\*\* Removing this text is now allowed! This template is licensed under CC BY 3.0 \*\*\*\*\* -->

<p>Copyright ©2017 Ca. Designed by <a href="https://colorlib.com" target="\_blank">Colorlib</a> | for DADS. Customized by NJ©</p>

</div>

</footer>

<!-- \*\*\*\*\* Footer Area Start \*\*\*\*\* -->

<!-- Jquery-2.2.4 JS -->

<script src="<?php echo base\_url('assets');?>/landing/js/jquery-2.2.4.min.js"></script>

<!-- Popper js -->

<script src="<?php echo base\_url('assets');?>/landing/js/popper.min.js"></script>

<!-- Bootstrap-4 Beta JS -->

<script src="<?php echo base\_url('assets');?>/landing/js/bootstrap.min.js"></script>

<!-- All Plugins JS -->

<script src="<?php echo base\_url('assets');?>/landing/js/plugins.js"></script>

<!-- Slick Slider Js-->

<script src="<?php echo base\_url('assets');?>/landing/js/slick.min.js"></script>

<!-- Footer Reveal JS -->

<script src="<?php echo base\_url('assets');?>/landing/js/footer-reveal.min.js"></script>

<!-- Active JS -->

<script src="<?php echo base\_url('assets');?>/landing/js/active.js"></script>

</body>

</html>

Jadi singkatnya proses *routing framework codeigniter* yaitu ketika *user* mengakses aplikasi pada *web browser,* maka sistem akan memanggil *controller* yang telah di set pada file *config routes.* Setelah itu *controller* memanggil file *view,* jadi halaman yang tampil pada *web browser* adalah file *view* yang ditentukan pada *controller.*

* 1. **Sistem Data**

Dalam sebuah sistem informasi tentunya ada proses *create, update, delete* dan *view* data. Pada penerapannya dengan *framework codeigniter*, keempat proses tersebut tidak jauh berbeda karena sama – sama melibatkan *controller* sebagai logika sistem, *model* sebagai *query database* dan *view* data sebagai tampilan, sehingga penulis hanya akan menjelaskan proses *view* data dan sebagai contoh pada saat *view* data note dengan aktornya adalah *Super\_User.*

Saat akan melakukan view data, maka *user* sebagai *super\_user* harus login terlebih dahulu. Setelah login, *super\_user* memilih *menu,* kemudian mengklik menu “note archive”. Pada menu *view* data note, *web browser* mendapatkan *link* “…/ddas/note”, dari *link* tersebut dapat dijelaskan bahwa sistem akan menuju “*function index()*” yang berada dalam file *controller “note.php”,* yang mana listing codenya sebagai berikut :

<?php

if (!defined('BASEPATH'))

exit('No direct script access allowed');

class Note extends CI\_Controller

{

function \_\_construct()

{

parent::\_\_construct();

$this->load->model('Note\_model');

$this->load->library('form\_validation');

}

public function index()

{

$q = urldecode($this->input->get('q', TRUE));

$start = intval($this->input->get('start'));

if ($q <> '') {

$config['base\_url'] = base\_url() . 'note/index.html?q=' . urlencode($q);

$config['first\_url'] = base\_url() . 'note/index.html?q=' . urlencode($q);

} else {

$config['base\_url'] = base\_url() . 'note/index.html';

$config['first\_url'] = base\_url() . 'note/index.html';

}

$config['per\_page'] = 10;

$config['page\_query\_string'] = TRUE;

$config['total\_rows'] = $this->Note\_model->total\_rows($q);

$note = $this->Note\_model->get\_limit\_data($config['per\_page'], $start, $q);

$this->load->library('pagination');

$this->pagination->initialize($config);

$data\_user = $this->Note\_model->get\_user\_data();

$data = array(

'note\_data' => $note,

'q' => $q,

'pagination' => $this->pagination->create\_links(),

'total\_rows' => $config['total\_rows'],

'start' => $start,

);

$this->load->view('note/note\_list', $data+$data\_user);

}

Pada baris “$this->Note\_model>get\_limit\_data($config['p

er\_page'], $start, $q);” , fungsi *index()* pada *controller note* ini akan memanggil fungsi *get\_limit\_data* pada file *Note\_model.php* yang terletak di ddas/application/models/… yang isinya :

function get\_limit\_data($limit, $start = 0, $q = NULL) {

$id = $this->session->userdata('is\_id');

$this->db->join('user','user.id\_user=note.id\_user');

if ($this->session->userdata('is\_level')== 'Super\_User'):

$this->db->order\_by($this->id, $this->order);

$this->db->like('id\_note', $q);

$this->db->or\_like('note', $q);

$this->db->or\_like('date', $q);

$this->db->or\_like('added', $q);

$this->db->or\_like('updated', $q);

$this->db->limit($limit, $start);

endif;

Kembali lagi pada fungsi *index()* pada *controller note,* setelah *query database* dilanjutkan lagi memanggil file *view* pada baris “$this->load->view('note/note\_list', $data+$data\_user);” baris tersebut memanggil file *note\_list.php* pada folder “ddas/application/view/note/” yang nantinya akan ditampilkan pada *web browser* yang berisi *code* sebagai berikut :

<?php $this->load->view('layout/head') ?>

<body>

<div class="container-fluid" id="wrapper">

<div class="row">

<?php $this->load->view('layout/nav') ?>

<main class="col-xs-12 col-sm-8 col-lg-9 col-xl-10 pt-3 pl-4 ml-auto">

<?php $this->load->view('layout/header') ?>

<section class="row">

<div class="col-sm-12">

<!-- paste what you wanted to display here -->

<ol class="breadcrumb">

<li class="breadcrumb-item"><a href="<?php echo base\_url('user/dashboard') ?>">Note Archive</a></li>

<li class="breadcrumb-item active">List</li>

</ol>

<div class="row" style="margin-bottom: 10px">

<div class="col-md-4 mb-2">

<?php echo anchor(site\_url('note/create'),'Add <span class="fa fa-plus-circle"></span>', 'class="btn btn-primary"'); ?>

</div>

<div class="col-md-4 text-center">

<div style="margin-top: 8px" id="message">

<?php echo $this->session->userdata('message') <> '' ? $this->session->userdata('message') : ''; ?>

</div>

</div>

<div class="col-md-1 text-right">

</div>

<div class="col-md-3 text-right">

<?php if ($this->session->userdata('is\_level') == 'Super\_User'):?>

<form action="<?php echo site\_url('note/index'); ?>" class="form-inline" method="get">

<div class="input-group">

<input type="text" class="form-control" name="q" value="<?php echo $q; ?>" placeholder="Search ...">&nbsp;

<span class="input-group-btn">

<?php

if ($q <> '')

{

?>

<a href="<?php echo site\_url('note'); ?>" class="btn btn-default">Reset</a>

<?php

}

?>

<button class="btn btn-primary" type="submit">Search <span class="fa fa-search"></span></button>

</span>

</div>

</form>

<?php endif ?>

</div>

</div>

<table class="table table-bordered table-striped" style="margin-bottom: 10px">

<thead>

<tr>

<th>No</th>

<th>Note</th>

<th>Date</th>

<th>Added</th>

<th>Updated</th>

<th>Uploaded By</th>

<th>Action</th>

</tr></thead><?php

foreach ($note\_data as $note)

{

?>

<tr>

<td><?php echo ++$start ?></td>

<td><?php echo $note->note ?></td>

<td><?php echo $note->date ?></td>

<td><?php echo $note->added ?></td>

<td><?php echo $note->updated ?></td>

<td><?php echo $note->nama ?></td>

<td style="text-align:center" width="200px">

<?php

echo anchor(site\_url('note/read/'.$note->id\_note),'<button class="btn btn-secondary btn-circle margin" type="button"><span class="fa fa-eye"></span></button>');

echo ' ';

echo anchor(site\_url('note/update/'.$note->id\_note),'<button class="btn btn-secondary btn-circle margin" type="button"><span class="fa fa-pencil-square-o"></span></button>');

echo ' ';

echo anchor(site\_url('note/delete/'.$note->id\_note),'<button class="btn btn-secondary btn-circle margin" type="button"><span class="fa fa-trash"></span></button>','onclick="javasciprt: return confirm(\'Anda Yakin ?\')"');

?>

</td>

</tr>

<?php

}

?>

</table>

<div class="row">

<div class="col-md-6">

<a href="#" class="btn btn-primary">Total Record : <?php echo $total\_rows ?></a>

<?php if ($this->session->userdata('is\_level') == 'Super\_User'):?>

<?php echo anchor(site\_url('note/word'), '<span class="fa fa-file-word-o"></span>', 'class="btn btn-primary"'); ?>

<?php endif ?>

</div>

<div class="col-md-6 text-right">

<?php echo $pagination ?>

</div>

</div>

<?php $this->load->view('layout/footer') ?>

</div>

</section>

</main>

</div>

</div>

</body>

<?php $this->load->view('layout/js') ?>

Pada akhirnya *user* sebagai *super\_user* akan melihat data – data list *note* yang telah dimasukkan kedalam sistem.

* 1. **Penerapan Keamanan**

Pada sisi keamanan, *Framework Codeigniter* menyarankan beberapa metode yang kemudian di dalam aplikasi ini yaitu XSS *filtering* dan anti SQL *injection* menggunakan *escaping query.*

* + - 1. XSS *filtering*

XSS merupakan kependekan yang digunakan untuk istilah *cross site scripting*. XSS merupakan salah satu jenis setangan injeksi *code* (*code injection attack*). XSS dilakukan oleh penyetang dengan cara memasukkan kode HTML atau *client script code* lainnya ke suatu situs. Akibat setangan ini antara lain yaitu penyerang dapat mem-*bypass* keamanan di sisi klien, mendapatkan informasi sensitif, atau menyimpan aplikasi berbahaya. Untuk menghindari hal tersebut maka *codeigniter* menawarkan suatu settingan konfigurasi XSS *filtering.* Cara mengaktifkannya adalah dengan merubah baris *code* pada “ddas/application/config/config.php” yang isi nya :

$config['global\_xss\_filtering'] = FALSE; Menjadi

$config['global\_xss\_filtering'] = TRUE;

* + - 1. Anti SQL *Injection*

SQL *Injection* adalah sebuah taknik yang menyalahgunakan sebuah celah keamanan yang terjadi dalam lapisan basis data sebuah aplikasi. Celah ini terjadi ketika masukan pengguna tidak disaring secara benar dari karakter – karakter pelolos bentukan string yang diimbuhkan dalam pernyataan SQL atau masukan pengguna tidak bertipe kuat dan karenanya ketika dijalankan tidak sesuai harapan. Ini sebenarnya adalah sebuah contoh dari sebuah kategori celah keamanan yang lebih umum yang dapat terjadi setiap kali sebuah Bahasa pemrograman atau skrip diimbuhkan di dalam bahasa yang lain.

Untuk menghindarinya maka digunakan *escaping query.* Dalam penerapannya di aplikasi ini akan digunakan contoh proses pengambilan data note berdasarkan parameter id nya. Berikut adalah conton penggunaan *escaping query* :

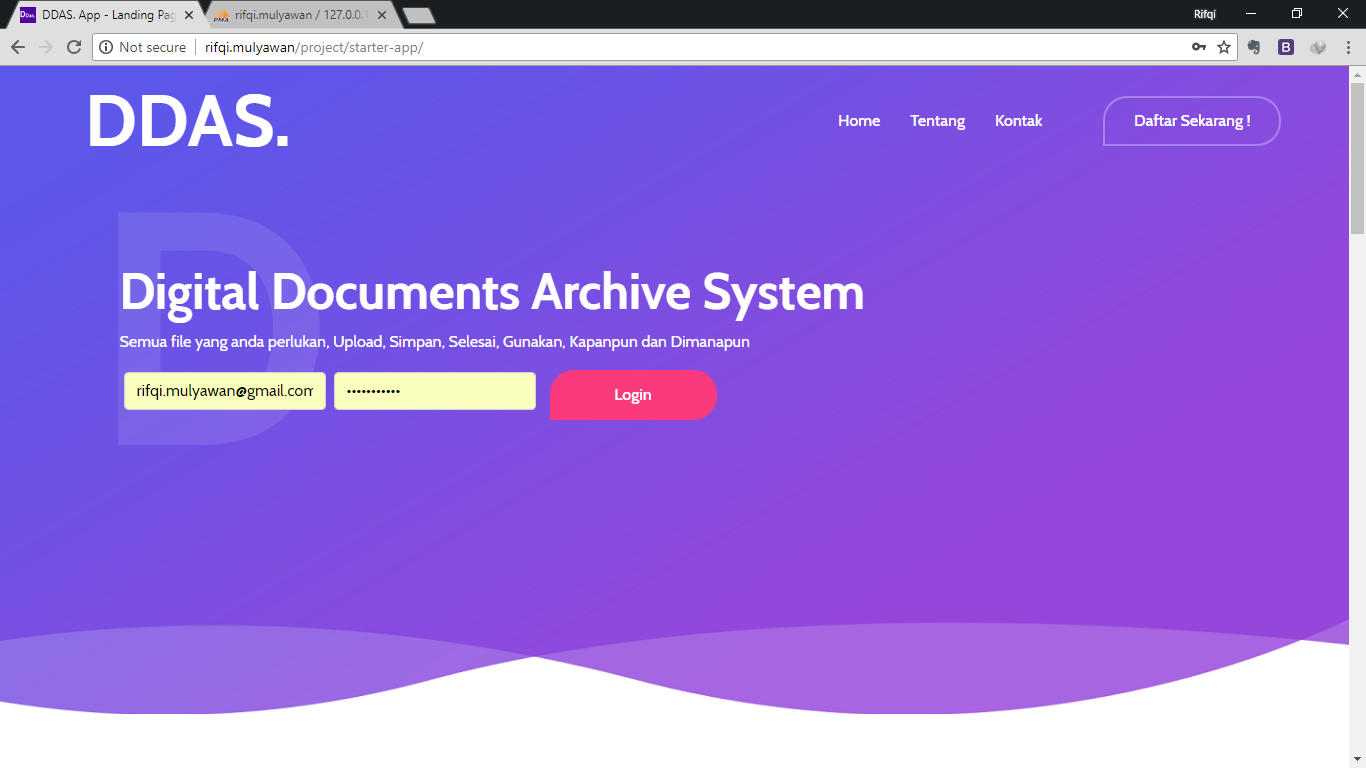
$this->db->escape($id);

## Hasil Tampilan Aplikasi

Berikut adalah hasil tampilan aplikasi berdasarkan rancangan sistem yang dibuat :

1. **Tampilan Antarmuka Masukan Sistem**
   1. Halaman *Landing*

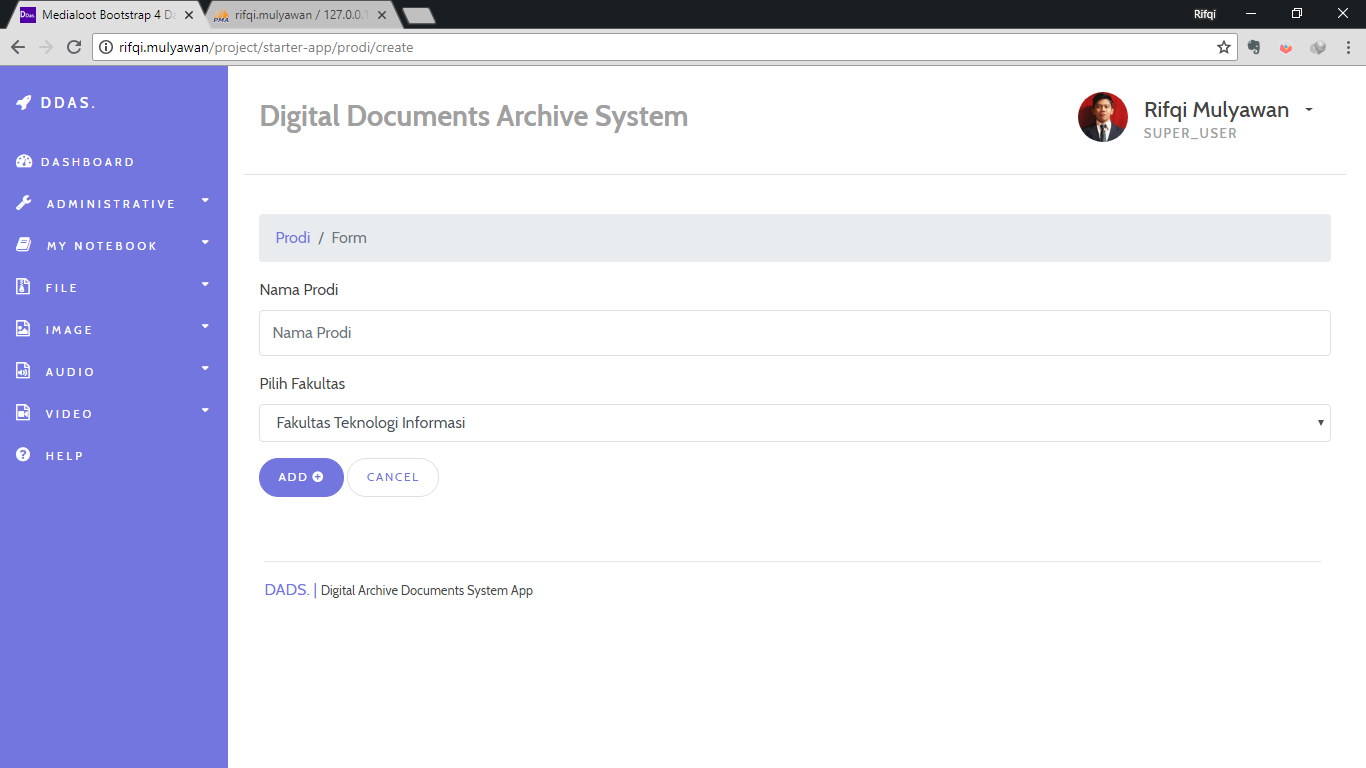
Halaman ini merupakan halaman utama sistem yang mana merupakan halaman *One Page View* (Halaman memuat beberapa halaman lainnya), yang berisi tentang sistem dan kontak sistem.



**Gambar 4. 2 Halaman Landing**

* 1. Halaman Input Data Prodi

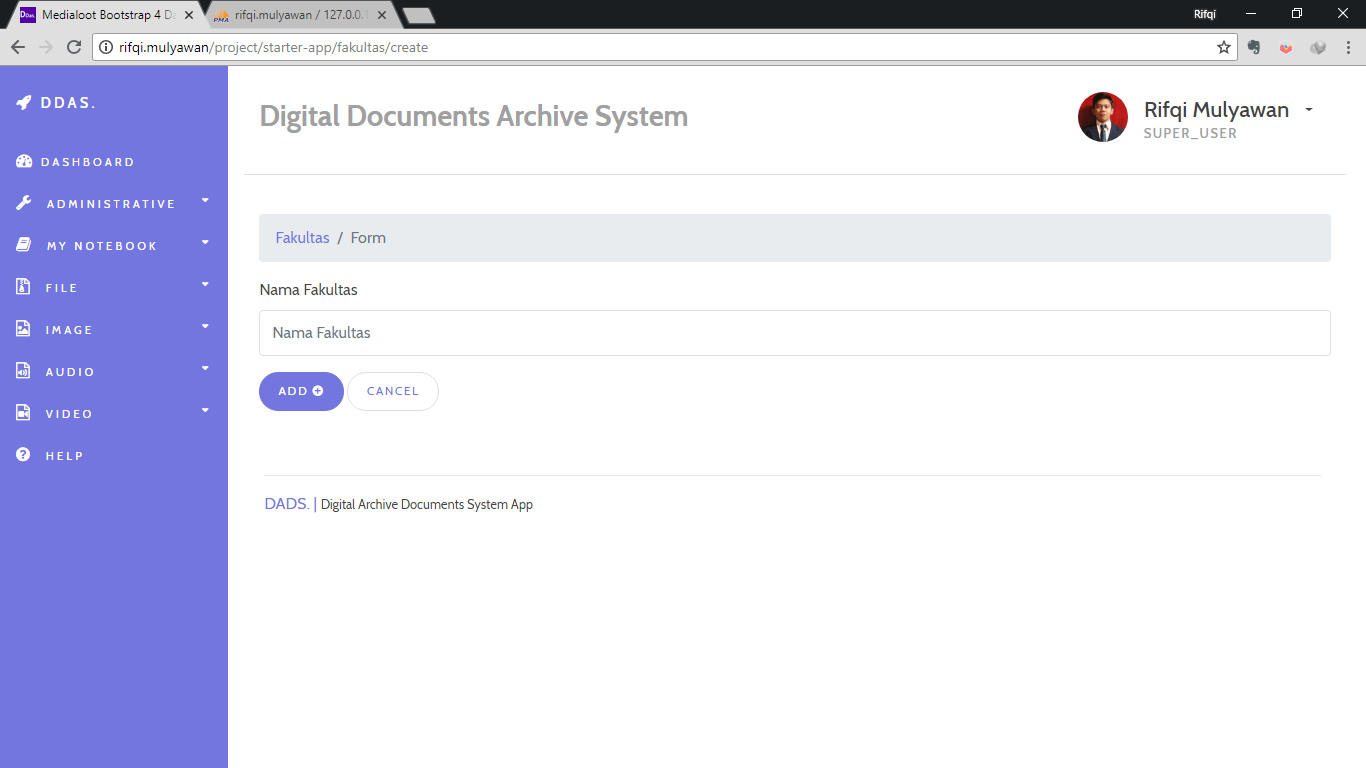
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* ketika hendak melakukan masukan data Prodi.



**Gambar 4. 3 Halaman Input Data Prodi**

* 1. Halaman Input Data Fakultas

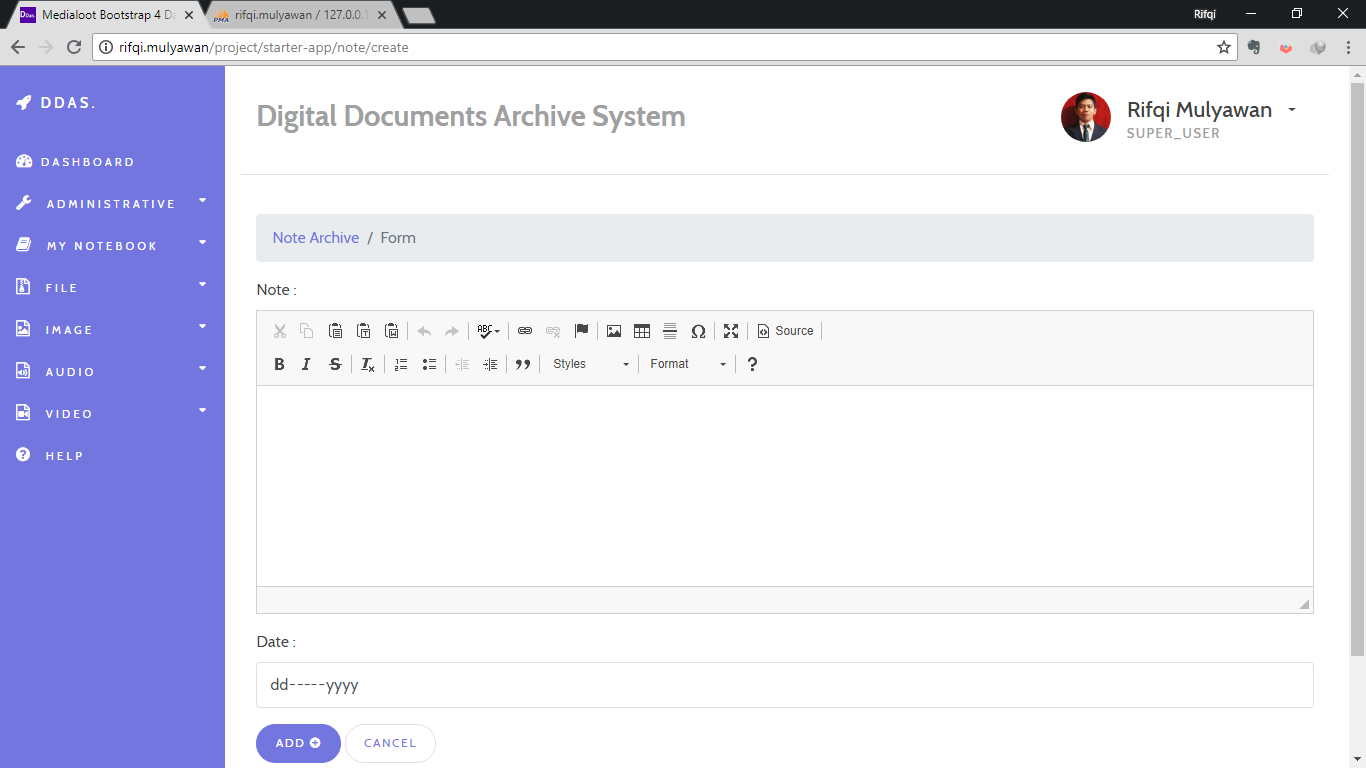
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* ketika hendak melakukan masukan data Fakultas.



**Gambar 4. 4 Halaman Input Data Fakultas**

* 1. Halaman *Upload* Arsip *Note*

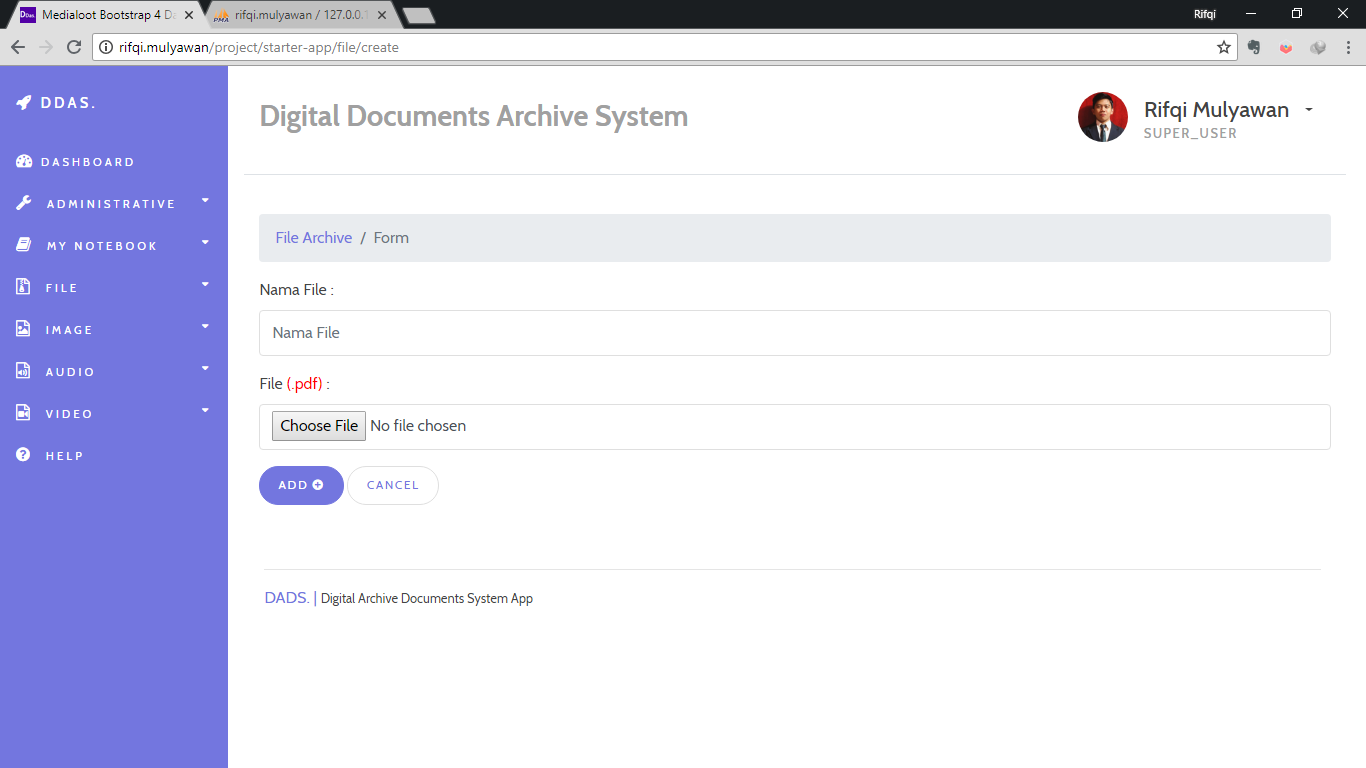
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* maupun *“User”* ketika hendak melakukan masukan data Arsip Catatan/*Note*.



**Gambar 4. 5 Halaman Upload Arsip Note**

* 1. Halaman *Upload* Arsip *File*

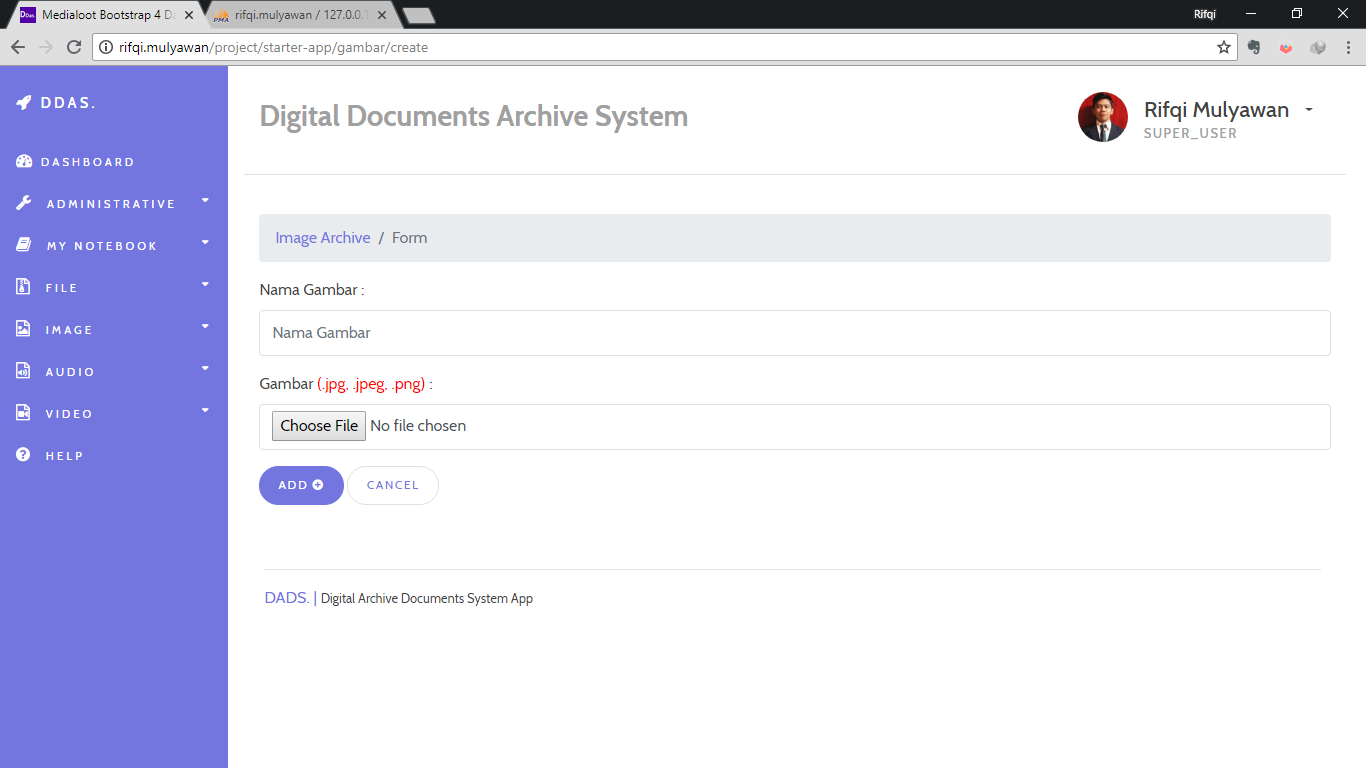
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* maupun *“User”* ketika hendak melakukan masukan data Arsip *File*.



**Gambar 4. 6 Halaman Upload Arsip File**

* 1. Halaman *Upload* Arsip Gambar

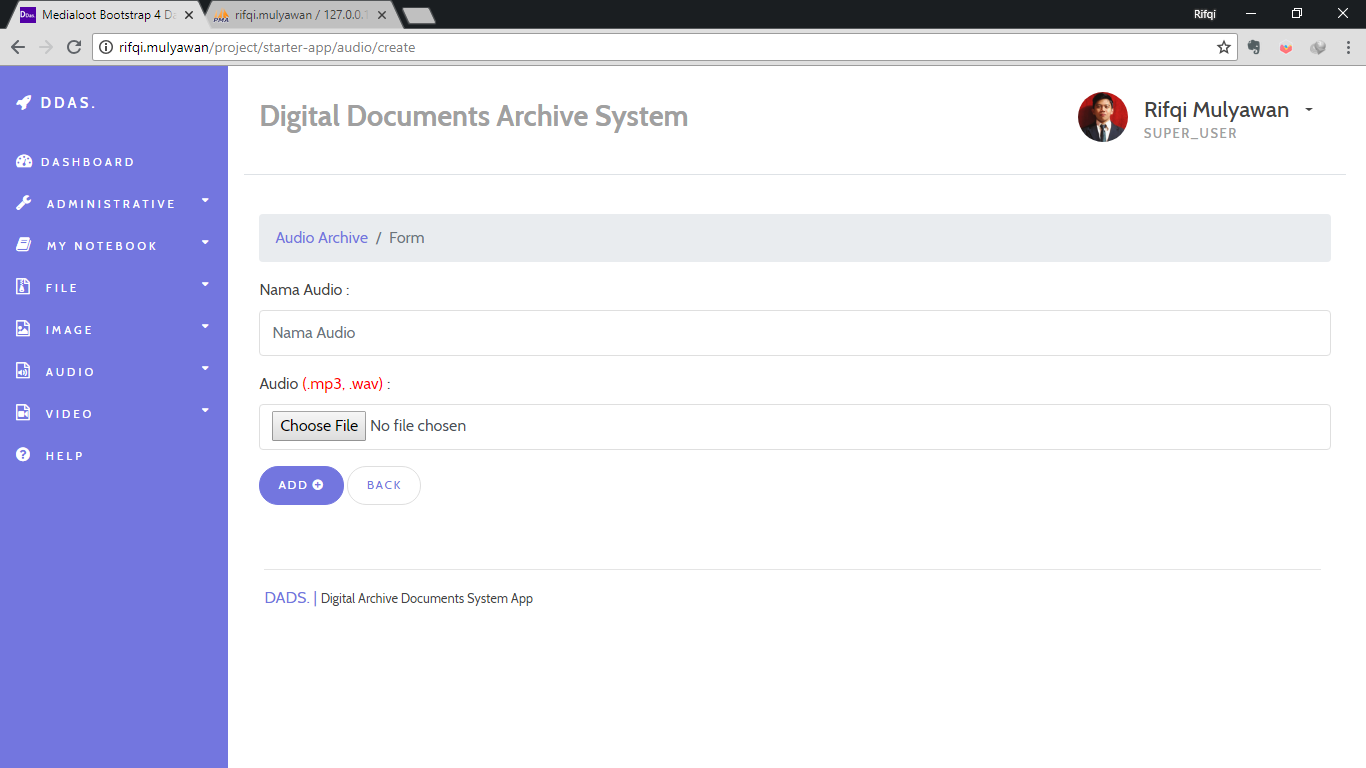
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* maupun *“User”* ketika hendak melakukan masukan data Arsip Gambar.



**Gambar 4. 7 Halaman Upload Arsip Gambar**

* 1. Halaman *Upload* Arsip Audio

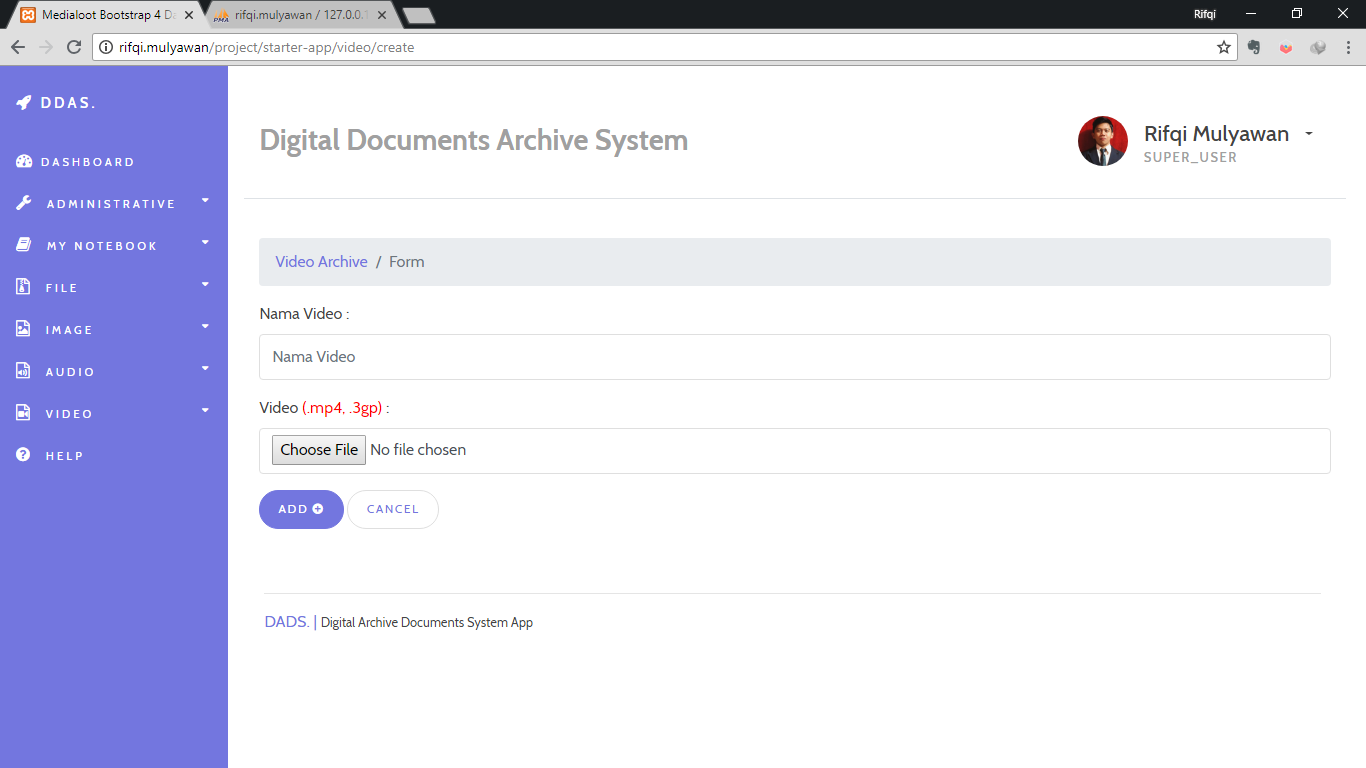
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* maupun *“User”* ketika hendak melakukan masukan data Arsip Audio.



**Gambar 4. 8 Halaman Upload Arsip Audio**

* 1. Halaman *Upload* Arsip Video

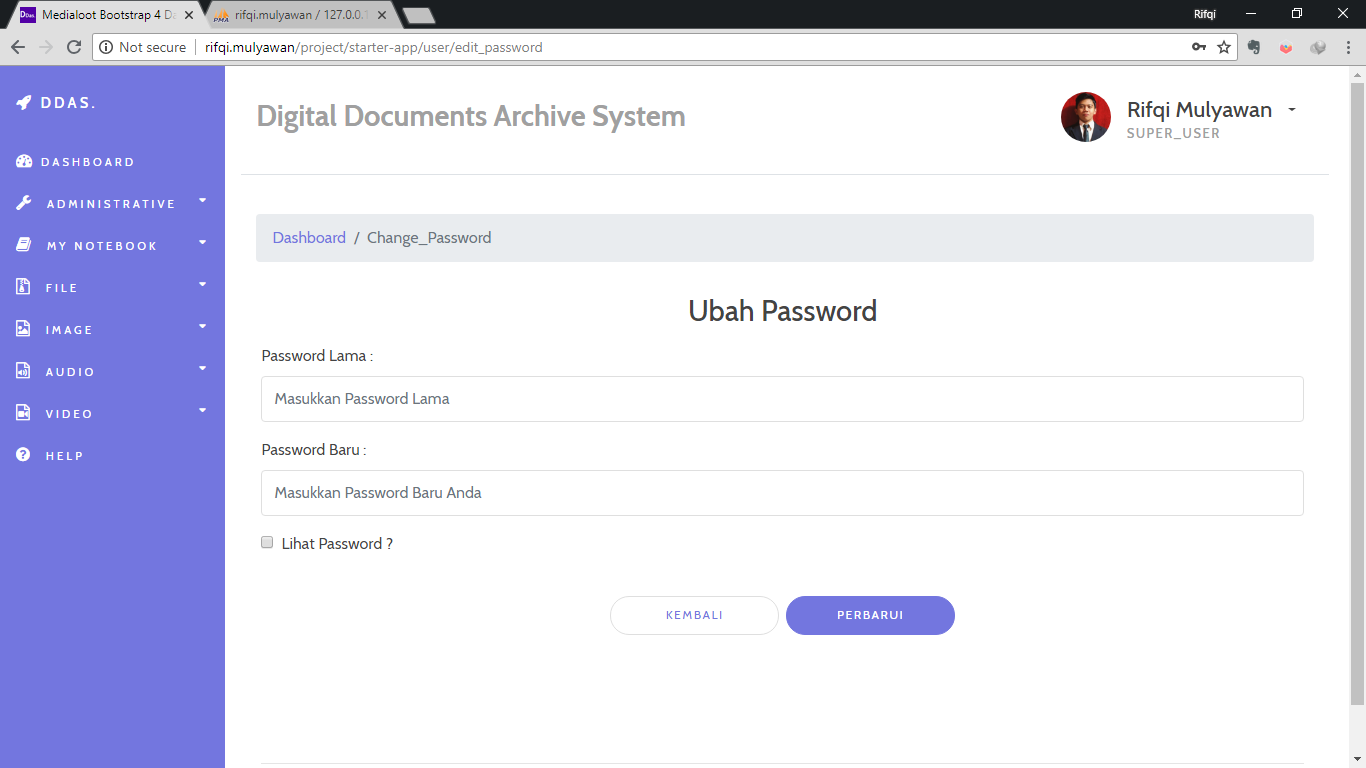
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* maupun *“User”* ketika hendak melakukan masukan data Arsip Video.



**Gambar 4. 9 Halaman Upload Arsip Video**

* 1. Halaman Ganti *Password*

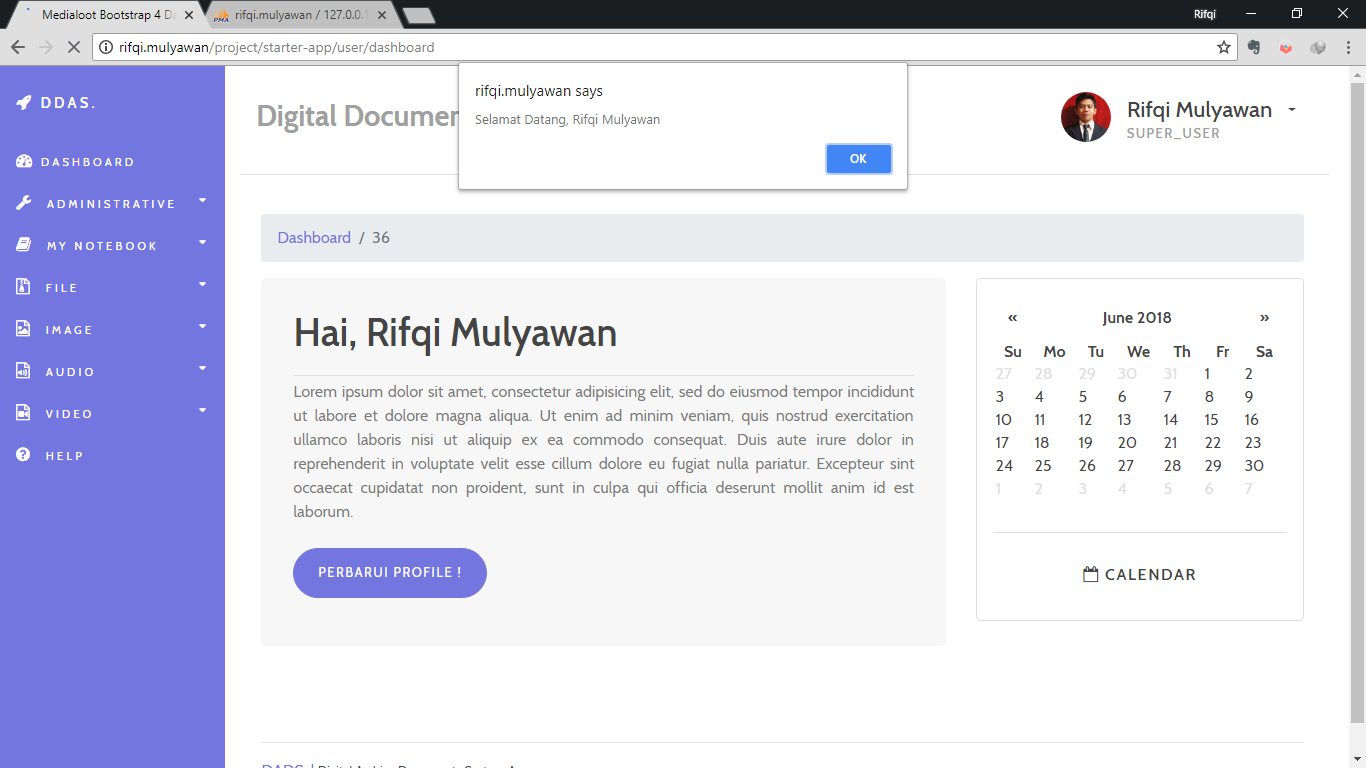
Halaman ini merupakan tampilan kepada *User* yang memiliki *Level “Super\_User”* maupun *“User”* ketika hendak melakukan proses perubahan *Password User*.



**Gambar 4. 10 Halaman Ganti Password**

1. **Tampilan Antarmuka Keluaran Sistem**
   1. Halaman *Dashboard User* Sebagai *Super\_User*

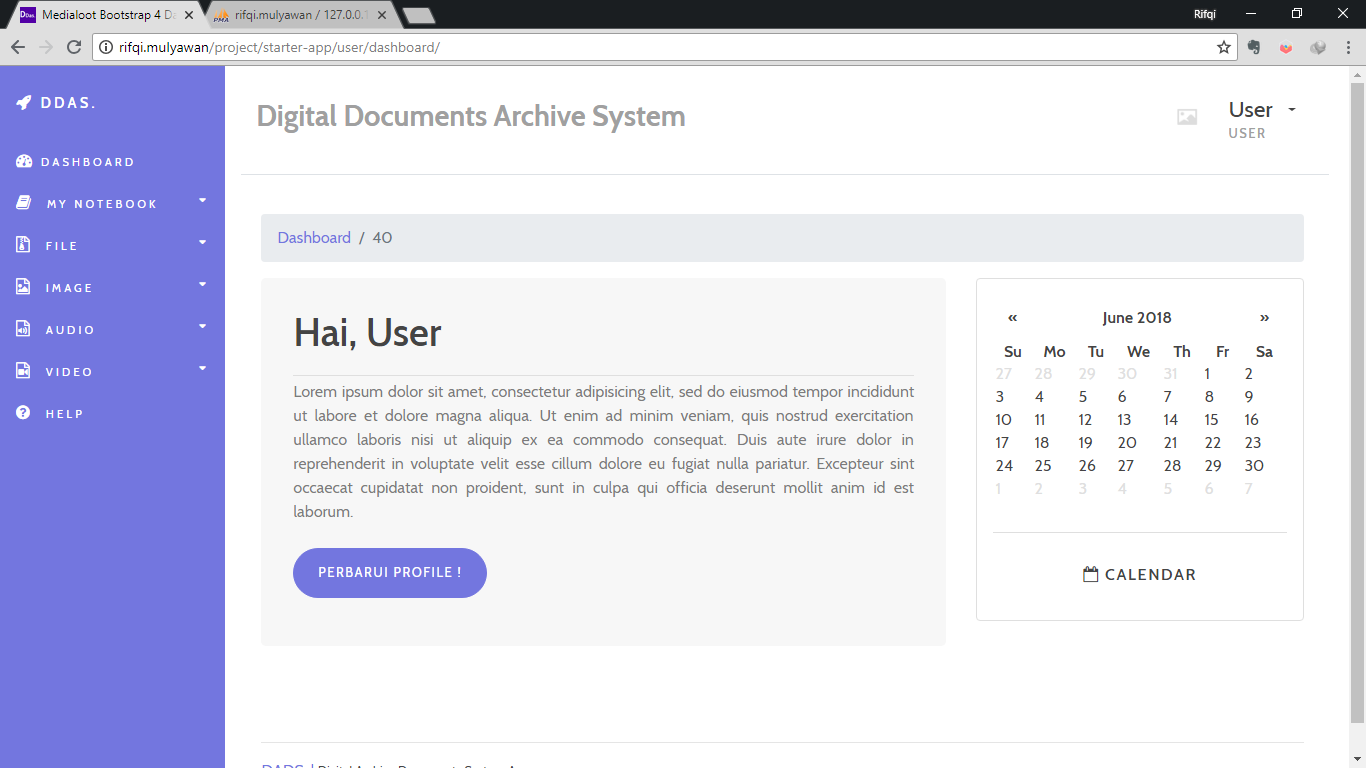
Setelah mengakses halaman *Landing* yang mana terdapat login sistem *user* akan dicek databasenya sesuai dengan level *user* tersebut dan kemudian langsung diarahkan ke halaman *Dashboard User* berdasarkan *Level User.*



**Gambar 4. 11 Halaman Dashboard Super\_User**

* 1. Halaman *Dashboard User* Sebagai *User*

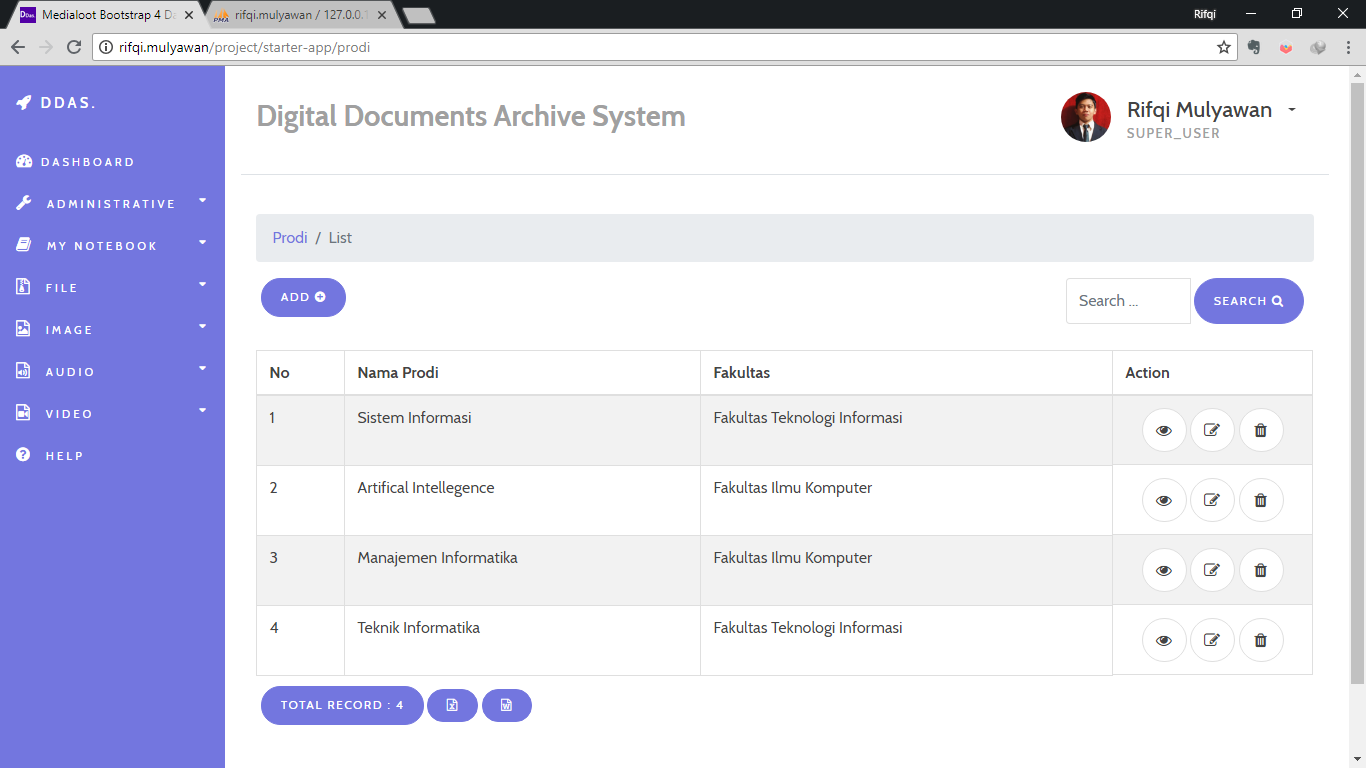
Halaman ini merupakan tampilan *User* yang telah *Login* pada sistem dengn level *User* yaitu *“User”.*



**Gambar 4. 12 Halaman Dashboard User**

* 1. Halaman *View* Data Prodi

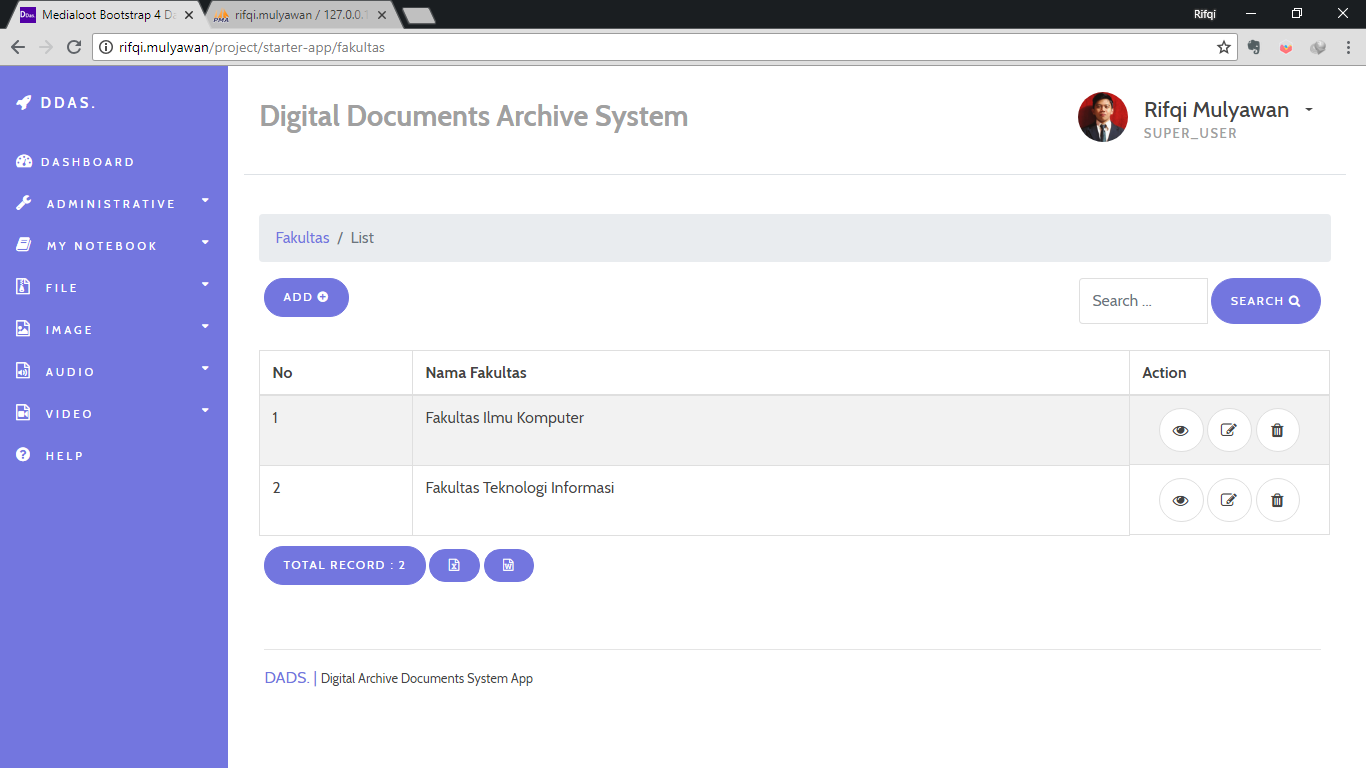
Halaman ini merupakan halaman yang dapat diakses oleh *User* dengan *Level “Super\_User”.* Berikut adalah tampilan *View* Prodi :



**Gambar 4. 13 Halaman View Data Prodi**

* 1. Halaman *View* Data Fakultas

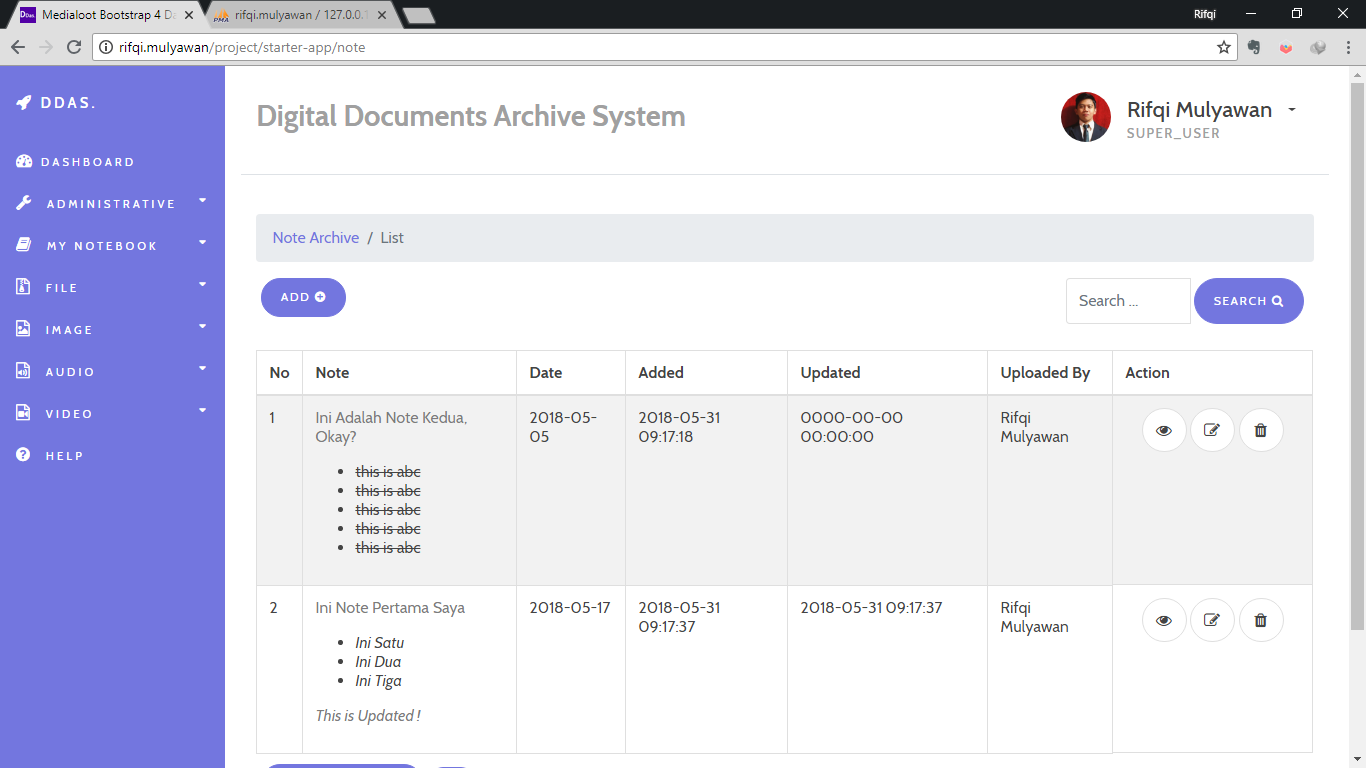
Halaman ini merupakan halaman yang dapat diakses oleh *User* dengan *Level “Super\_User”.* Berikut adalah tampilan *View* Data Fakultas :



**Gambar 4. 14 Halaman View Data Fakultas**

* 1. Halaman *View* Data Arsip *Note*

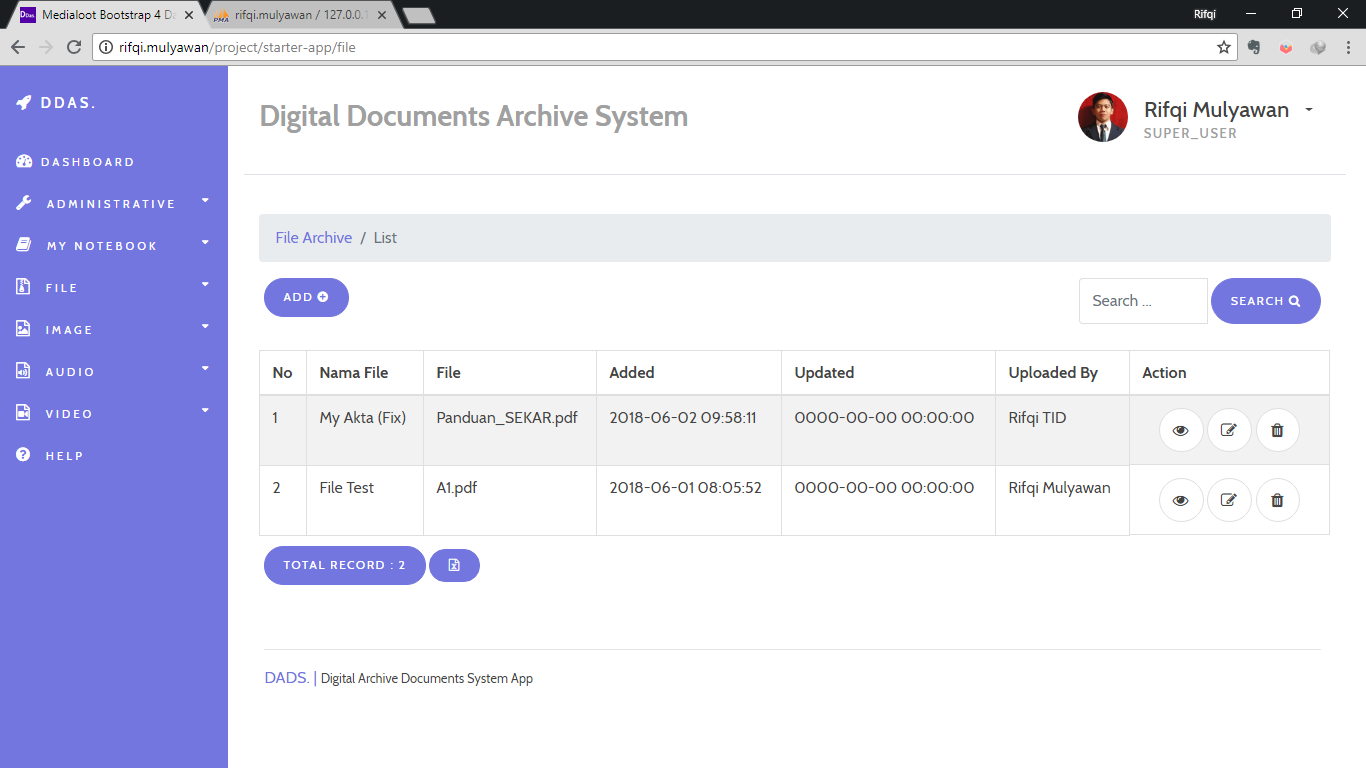
Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman *View* Data Arsip *Note* :



**Gambar 4. 15 Halaman View Data Arsip Note**

* 1. Halaman *View* Data Arsip *File*

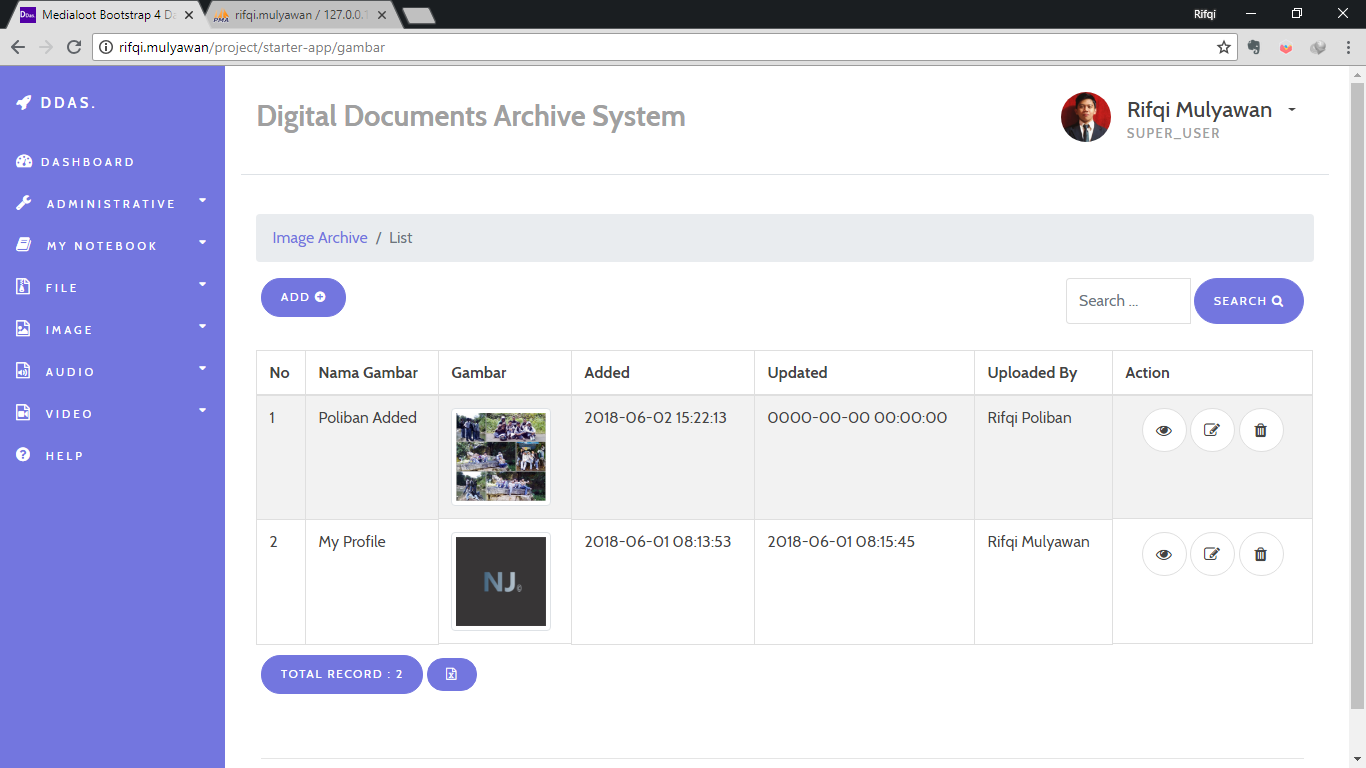
Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman *View* Data Arsip *File* :



**Gambar 4. 16 Halaman View Data Arsip File**

* 1. Halaman *View* Data Arsip Gambar

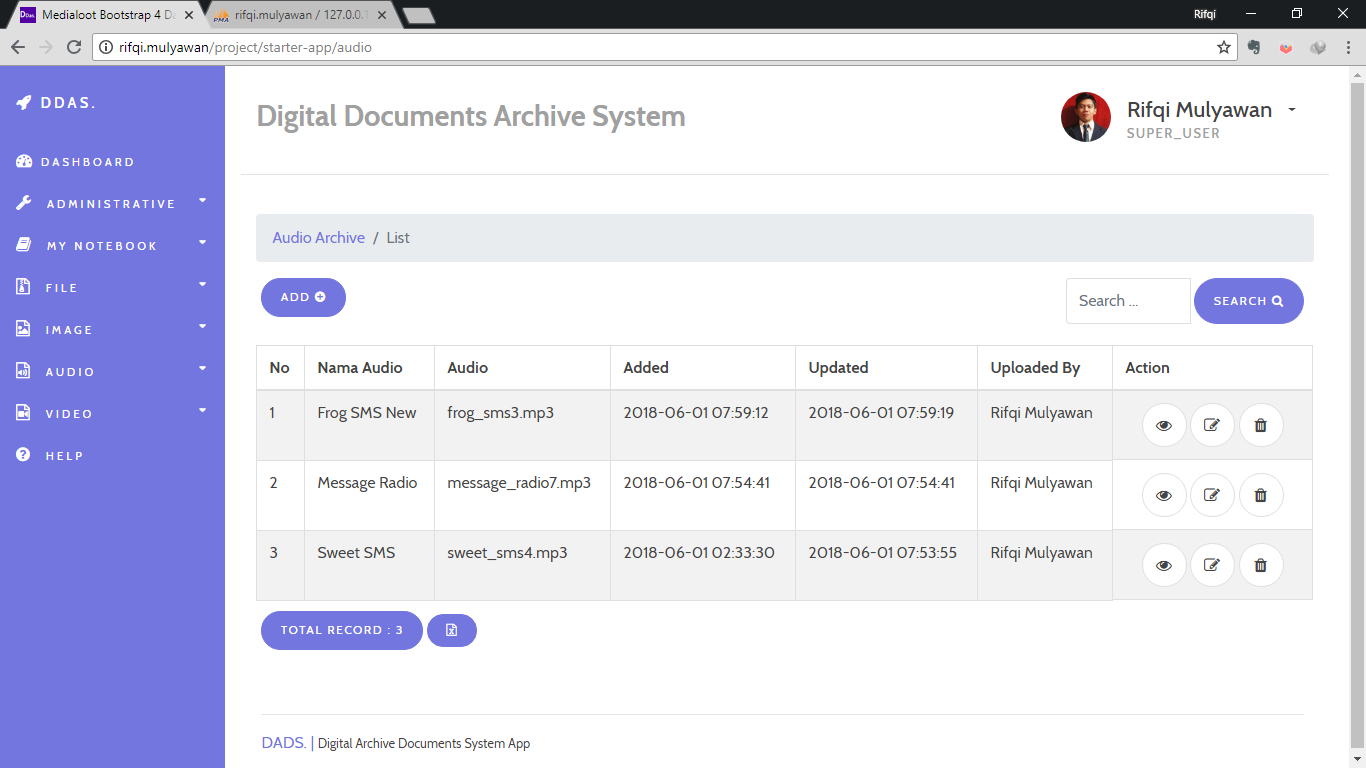
Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman *View* Data Arsip Gambar :



**Gambar 4. 17 Halaman View Data Arsip Gambar**

* 1. Halaman *View* Data Arsip Audio

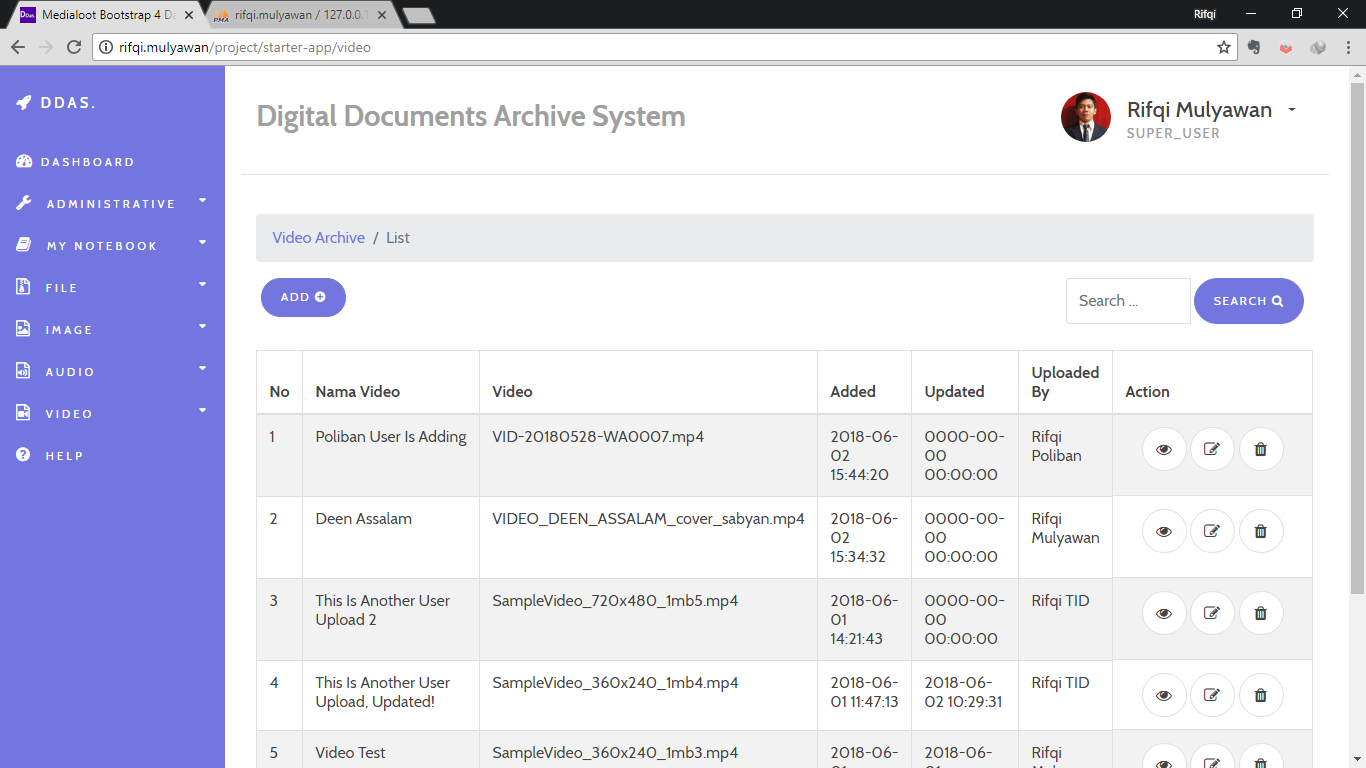
Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman *View* Data Arsip Audio :



**Gambar 4. 18 Halaman View Data Arsip Video**

* 1. Halaman *View* Data Arsip Video

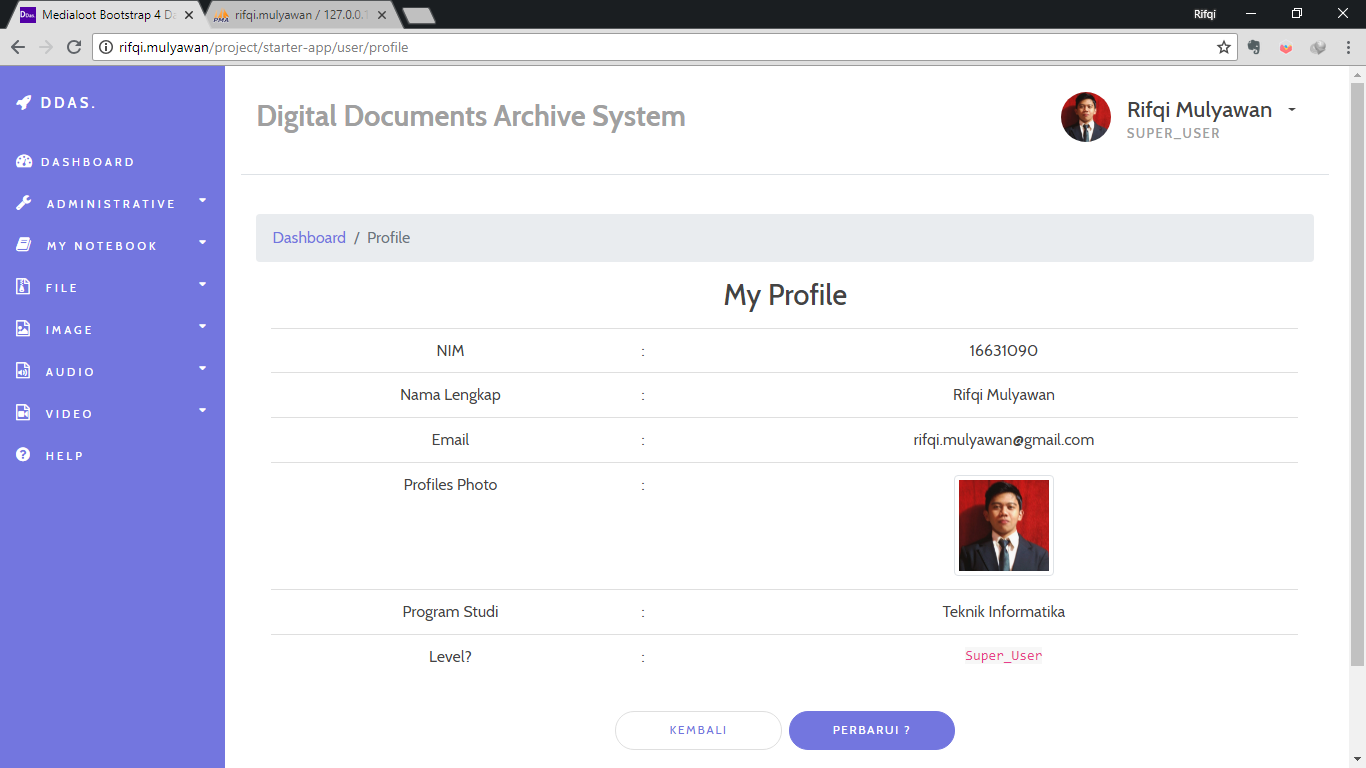
Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman *View* Data Arsip Video :



**Gambar 4. 19 Halaman View Data Arsip Video**

* 1. Halaman Profil

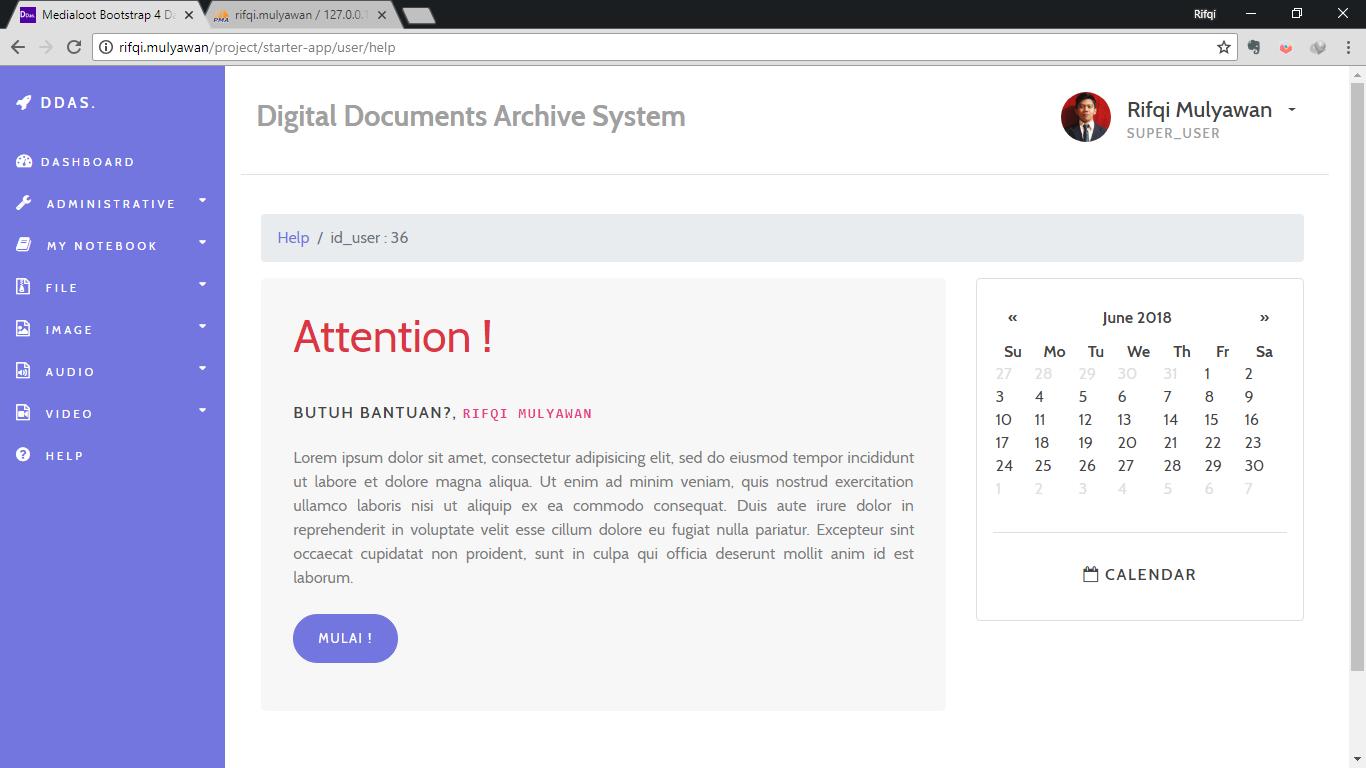
Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman Profil :



**Gambar 4. 20 Halaman Profil**

* 1. Halaman *Help*

Halaman ini adalah halaman yang dapat dilihat oleh semua *User.* Berikut adalah tampilan halaman *Help* :



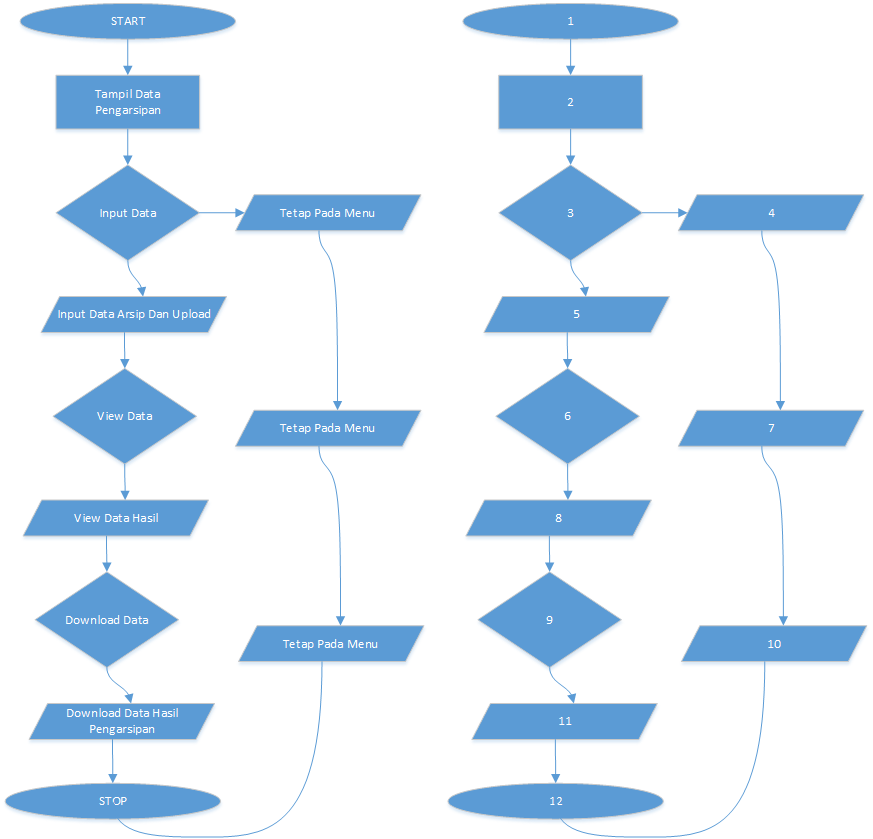
**Gambar 4. 21 Halaman Help**

## Pengujian

Pengujian pada sistem yang baru dibangun dilakukan melalui tahap pengujian *white-box,* dan pengujian *black-box.*

* + - 1. Pengujian White-Box

Pengujian *white-box* merupakan metode perancangan *test case* yang menggunakan struktur control dari perancangan procedural dalam mendapatkan *test case.* Adapun metode yang digunakan dalam pengujian *white-box* ini adalah metode basis *path.* Metode *basis path* mengijinkan pendesain kasus uji untuk membuat perkiraan logis yang kompleks dari desain procedural dan penggunaan perkiraan ini untuk mendefinisikan aliran eksekusi.



**Gambar 4. 22 Flowchart Dan Flowgraph Menu Pengarsipan Digital**

Dari *flowgraph* Halaman Login dapat diketahui :

* + - * 1. Cyclomatic Complexity

E (edge) = 12

N (Node) = 10

V (G) = E – N + 2

= 12 – 10 + 2

= 4

* + - * 1. Predicate Node (P)

V (G) = P + 1

= 3 + 1

= 4

* + - * 1. Jumlah Region (R) *Flowgraph* Pengarsipan adalah 4 Region.
        2. Path – path yang terdapat pada *Flowgraph* Pengarsipan adalah :
* Path 1 : 1 – 2 – 3 – 5 – 6 – 8 – 9 – 11 – 12
* Path 1 : 1 – 2 – 3 – 4 – 5 – 6 – 8 – 9 – 11 – 12
* Path 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 11 – 12
* Path 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12

Dapat disimpulkan dari perhitungan *Flowgraph* pengarsipan terdapat *Cyclomatic Complexity =* 4, *Predicate Node =* 4, *Region =* 4 dan *Independent Path =* 4. Karena nilai yang dihasilkan dari keempat parameter tersebut hasilnya sama, maka dapat disimpulkan bahwa rancangan program bebas dari kesalahan logika program.

* + - 1. Pengujian Black-Box

Pengujian *black-box,* yaitu suatu pendekatan untuk menguji apakah setiap fungsi di dalam program dapat berjalan dengan benar. Berikut beberapa proses yang dilakukan penulis dalam pengujian ini, yaitu :

Fungsi – fungsi yang tidak benar, baik input maupun output

Kesalahan Interface

Kesalahan dalam struktur data atau akses database

Di bawah ini merupakan table hasil pengujian dari aplikasi kearsipan digital dokumen akademik mahasiswa :

**Tabel 4. 2 List Hasil Pengujian Black-Box**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | RancangAn | Hasil Diharapkan | Hasil | Keterangan |
| 1 | Awal akses aplikasi | Masuk Halaman Landing | Sesuai |  |
| 2 | Memasukkan Data Login Super\_User | Masuk Halaman Dashboard Super\_User | Sesuai |  |
| 3 | Memasukkan Data Login User | Masuk Halaman Dashboard User | Sesuai |  |
| 4 | Klik Tab Navigasi | Masuk Kehalaman Yang Dituju | Sesuai |  |
| 5 | Isian Input Registrasi | Data Tersimpan Di Database | Sesuai | Nilai Input Valid |
| 6 | Input Data Upload | Tersimpan Di Database | Sesuai | Nilai Input Valid |
| 7 | Pennginputan Data Yang Tidak Lengkap | Data Belum Diubah Dan Belum Tersimpan Di Database | Sesuai | Nilai Input Invalid |
| 8 | Edit Data kemudian Klik Perbarui | Data Tersimpan Di Database | Sesuai | Nilai Input Valid |
| 9 | Klik Lihat Profil | Masuk Ke Halaman Profil User | Sesuai |  |
| 10 | Klik Pergantian Password | Menuju Ke Halaman Edit Password | Sesuai |  |
| 11 | Perbarui Password | Data Diubah Dan Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 12 | Edit Password Belum Lengkap | Data Belum Berubah | Sesuai |  |
| 13 | Tambah Data Prodi | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 14 | Edit Data Prodi Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Invalid |
| 15 | View Data Prodi | Menampilkan Halaman View Prodi | Sesuai |  |
| 16 | Tambah Data Fakultas | Data Tersimpan Di Database | Sesuai | Nilai Input Valid |
| 17 | Edit Data Fakultas Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 18 | View Data Fakultas | Menampilkan Halaman View Fakultas | Sesuai |  |
| 19 | Tambah Data Note | Data Berhasil Tersimpan Di Database | Sesuai | Nilai Input Valid |
| 20 | Tambah Data Note Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 21 | Edit Data Note Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 22 | Edit Data Note Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 23 | Hapus Data Note | Data Pada Database Terhapus | Sesuai |  |
| 24 | Tambah Data Arsip Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 25 | Tambah Data Arsip File Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 26 | Edit Data Arsip File Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 27 | Edit Data Arsip File Belum Lengkap | Data Belum Berubah Pada Database | Sesuai | Nilai Input Tidak Valid |
| 28 | View Data Hasil Pengarsipan File | Menuju Kehalaman Tampil Data Berdasarkan Id | Sesuai |  |
| 29 | Hapus Data Pengarsipan File | Data Terhapus Pada Database | Sesuai |  |
| 30 | Tambah Data Arsip Gambar Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 31 | Tambah Data Arsip Gambar Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 32 | Edit Data Arsip Gambar Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 33 | Edit Data Arsip Gambar Belum Lengkap | Data Belum Berubah Pada Database | Sesuai | Nilai Input Tidak Valid |
| 34 | View Data Hasil Pengarsipan Gambar | Menuju Kehalaman Tampil Data Berdasarkan Id | Sesuai |  |
| 35 | Hapus Data Pengarsipan Gambar | Data Terhapus Pada Database | Sesuai |  |
| 36 | Tambah Data Arsip Audio Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 37 | Tambah Data Arsip Audio Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 38 | Edit Data Arsip Audio Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 39 | Edit Data Arsip Audio Belum Lengkap | Data Belum Berubah Pada Database | Sesuai | Nilai Input Tidak Valid |
| 40 | View Data Hasil Pengarsipan Audio | Menuju Kehalaman Tampil Data Berdasarkan Id | Sesuai |  |
| 41 | Hapus Data Pengarsipan Audio | Data Terhapus Pada Database | Sesuai |  |
| 42 | Tambah Data Arsip Video Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 43 | Tambah Data Arsip Video Dengan Ukuran <20Mb | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 44 | Tambah Data Arsip Video >20Mb | Data Tidak Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 45 | Tambah Data Arsip Video Belum Lengkap | Data Belum Tersimpan Pada Database | Sesuai | Nilai Input Tidak Valid |
| 46 | Edit Data Arsip Video Lengkap | Data Tersimpan Pada Database | Sesuai | Nilai Input Valid |
| 47 | Edit Data Arsip Video Belum Lengkap | Data Belum Berubah Pada Database | Sesuai | Nilai Input Tidak Valid |
| 48 | View Data Hasil Pengarsipan Video | Menuju Kehalaman Tampil Data Berdasarkan Id | Sesuai |  |
| 49 | Hapus Data Pengarsipan Video | Data Terhapus Pada Database | Sesuai |  |

# BAB V PENUTUP

## Kesimpulan

Dari penelitian dan tulisan yang telah diuraikan, maka dapat ditarik kesimpulan sebagai berikut :

* + - 1. Aplikasi yang dikembangkan ini diusulkan untuk membantu proses kegiatan mahasiswa dalam mengarsipkan dokumen – dokumen akademik.
      2. Semua data tersimpan di dalam sistem secara terpusat dan terkomputerisasi sehingga proses pencarian dapat dilakukan dengan cepat berdasarkan waktu pengarsipan terbaru.
      3. Pengembangan aplikasi dengan menggunakan *Codeigniter Framework* dapat dilaksanakan sesuai dengan analisis dan perancangan yang telah dilakukan dengan menggunakan UML sebagai *tools*, yang juga sesuai dengan kebutuhan penggunanya.
      4. Penerapan *Codeigniter Framework* pada pengembangan sistem kearsipan digital dokumen akademik mahasiswa ini, membuat sistem memiliki beberapa keunggulan sebagai berikut :
         1. Performa yang lebih handal dan cepat karena adanya fitur *Caching* saat menjelajah data dalam sistem serta struktur file yang lebih rapi.
         2. Penulisan *Source Code* yang lebih efektif, efisien dan tertata rapi, sehingga dapat memudahkan dalam pemeliharaan dan pengembangan lebih lanjut dikemudian hari sesuai dengan perkembangan kebutuhan penggunanya.
         3. Penulisan struktur *Source Code* berdasarkan konsep *Model View Controller* (MVC), mengoptimalkan proses *coding,* contoh pada saat perancangan tata tampilan yang *user friendly* dengan CSS *Framework* Bootstraps*,* tanpa mengubah logika kontrol aplikasi dan *query database.*
         4. Konfigurasi keamanan yang lebih mudah dengan menggunakan fitur standar bawaab *Framework Codeigniter* dengan metode XSS *Filtering* dan Anti *SQL Injection.*

## Saran

Aplikasi ini tentu masih saja belum sempurna. Masih banyak hal yang dapat dilakukan untuk mengembangkan aplikasi ini agar menjadi lebih baik lagi, antara lain :

* + - 1. Penulis mengharapkan adanya penambahan fitur – fitur pada sistem, misalnya seperti fitur *chatting* antar user dan juga fitur – fitur lain yang dapat memudahkan keperluan mahasiswa dalam kegiatan akademik.
      2. Pengembangan juga dapat ditingkatkan dengan melakukan peningkatan keamanan yang lebih baik dan kompleks lagi, dikarenakan keamanan pada aplikasi ini hanya memanfaatkan fungsi keamanan standar dari *Framework Codeigniter.*

# DAFTAR PUSTAKA

Collins, ‘English Dictionary‘, Definition of ‘Academic’, 17 Juni 2016, <//www.collinsdictionary.com/English/academic > [diakses 09 Juni 2018]

Arief, M.Rudianto. Pemrograman Web Dinamis menggunakan Php dan Mysql.Yogyakarta : Andi Offset, 2011.

Bhasin, Hars, dkk. “Black Box Testing based on Requirement Analysis and Design Specifications”, International Journal of Computer Applications, vol. 87 no.018.http://research.ijcaonline.org/volume87/number18/pxc3894024.pdf (30 mei 2016).

Daft, Richard L. Manajemen Edisi Kelima Jilid Satu. Jakarta: Erlangga, 2002.

Gassing, Qadir dan Wahyuddin Halim. Pedoman Penulisan Karya Tulis Ilmiah Makalah Skripsi Tesis dan Disertasi. Makassar: Alauddin Press, 2014.

Ladjamudin, A,B. Analisis dan Desain Sistem Informasi. Yogyakarta. Graha Ilmu, 2005.

Mujianto, Ade. “Pembangunan Sistem Informasi Administrasi dan Penjadwalan Berbasis Dekstop di SMA Negeri 1 Cilacap”. Skripsi. Bandung: Universitas Komputer Indonesia, 2012.

Mendikbud. Pedoman Ejaan yang disempurnakan. Yogyakarta: Pustaka Widyatama, 2006.

Myer, Thomas, “Professional CodeIgniter”, Wiley Publishing; 2008.

Pressman, R.S., Rekayasa Perangkat Lunak, Buku Satu, Terj. Harnaningrum L.N., Yogyakarta: Andi Offset,2002.

Salim, Rin Rin Meilani, “Pengembangan Sistem Informasi Klinik Berbasis Web”. Skripsi. Medan: STIMIK Mikroskil, 2013.

Scribd.com. Jurnal Sistem Informasi Manajemen. https://www.scribd.com, (16 Agustus 2016).

Upton, David,“CodeIgniter for Rapid PHP Application Development”, Packt Publishing; 2007.