**PENJADWALAN PENDADARAN OTOMATIS MENGGUNAKAN METODE MATRIKS**

Acc Ujian Pendadaran

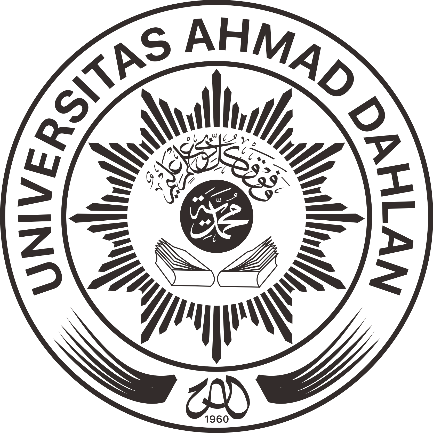
04/09/2020

Ardiansyah, S.T., M.Cs

**SKRIPSI**

**Untuk Memenuhi Sebagian Persyaratan**

**Mencapai Derajat Sarjana Komputer**

****

**Disusun Oleh :**

**Muhammad Nur Widya Luthfiantoro**

**1600018080**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS AHMAD DAHLAN**

**YOGYAKARTA**

**2020**

# 

# DAFTAR ISI

[DAFTAR ISI ii](#_Toc50379413)

[DAFTAR TABEL iv](#_Toc50379414)

[DAFTAR GAMBAR v](#_Toc50379415)

[DAFTAR LISTING vi](#_Toc50379416)

[ABSTRAK vii](#_Toc50379417)

[BAB I PENDAHULUAN 1](#_Toc50379418)

[1.1. Latar Belakang Masalah 1](#_Toc50379419)

[1.2. Identifikasi Masalah 2](#_Toc50379420)

[1.3. Batasan Masalah 3](#_Toc50379421)

[1.4. Rumusan Masalah 3](#_Toc50379422)

[1.5. Tujuan Penelitian 3](#_Toc50379423)

[1.6. Manfaat Penelitian 4](#_Toc50379424)

[BAB II LANDASAN TEORI 5](#_Toc50379425)

[2.1. Kajian Pustaka 5](#_Toc50379426)

[2.2. Kajian Teori 9](#_Toc50379427)

[2.2.1. Tugas Akhir 9](#_Toc50379428)

[2.2.2. Penjadwalan 9](#_Toc50379429)

[2.2.3. Metode Matriks 10](#_Toc50379430)

[BAB III METODOLOGI PENELITIAN 14](#_Toc50379431)

[3.1. Obyek Penelitian 14](#_Toc50379432)

[3.2. Metode Pengumpulan Data 14](#_Toc50379433)

[3.3. Alat Penelitian 14](#_Toc50379434)

[3.3.1. Perangkat Keras 14](#_Toc50379435)

[3.3.2. Perangkat Lunak 15](#_Toc50379436)

[3.4. Metode Penelitian 15](#_Toc50379437)

[BAB IV HASIL DAN PEMBAHASAN 18](#_Toc50379438)

[4.1. Analisis Sistem 18](#_Toc50379439)

[4.2. Perancangan Sistem 20](#_Toc50379440)

[4.3. Implementasi 29](#_Toc50379441)

[4.4. Pengujian Fungsional 32](#_Toc50379442)

[BAB V KESIMPULAN DAN SARAN 34](#_Toc50379443)

[5.1. Kesimpulan 34](#_Toc50379444)

[5.2. Saran 34](#_Toc50379445)

[DAFTAR PUSTAKA 36](#_Toc50379446)

# DAFTAR TABEL

[Tabel 2.1. : Perbandingan penelitian terdahulu 8](#_Toc50381180)

[Tabel 2.2. : Tabel jadwal dosen mengajar 10](#_Toc50381181)

[Tabel 2.3. : Tabel hasil rekomendasi 13](#_Toc50381182)

[Tabel 4.1. : Tabel jadwal rekomendasi 24](#_Toc50381185)

[Tabel 4.2. :Daftar Tabel Pertanyaan Black Box Test 33](#_Toc50381186)

# DAFTAR GAMBAR

[Gambar 4.1. : Jadwal Dosen 20](#_Toc50381223)

[Gambar 4.2. : Rancangan Halaman Login 25](#_Toc50381224)

[Gambar 4.3. : Rancangan daftar menu 26](#_Toc50381225)

[Gambar 4. 4. : Rancangan halaman daftar pendadaran 27](#_Toc50381226)

[Gambar 4.5. : Rancangan halaman data dosen 28](#_Toc50381227)

[Gambar 4.6. : Halaman Login Pengguna 29](#_Toc50381228)

[Gambar 4. 7. : Halaman Pendaftaran Pendadaran 30](#_Toc50381229)

[Gambar 4.8. : Halaman Data Pendadaran 32](#_Toc50381230)

# DAFTAR LISTING

[Listing 4.1. : Query login pengguna 36](#_Toc50379339)

[Listing 4.2 : Script rekomendasi pendadaran 38](#_Toc50379340)

[Listing 4.3 : Query pendaftaran pendadaran 38](#_Toc50379341)

[Listing 4. 4: Query data pendadaran 39](#_Toc50379342)

**PENJADWALAN PENDADARAN OTOMATIS MENGGUNAKAN METODE MATRIKS**

**Muhammad Nur Widya Luthfiantoro**

**1610008080**

# ABSTRAK

Tugas akhir merupakan salah satu mata kuliah dalam kurikulum pendidikan perguruan tinggi dan menempati posisi sangat strategis dalam pencapaian kualitas lulusan yang memadai. Universitas telah mengalokasikan sumber daya mereka untuk mengelola tugas akhir. Pengelolaan tugas akhir di beberapa universitas masih manual, biasanya ditulis di kertas atau ditulis di *Microsoft Excel*. Salah satu kelemahan menggunakan *Microsoft Excel* adalah tidak bisa mencegah atau mendeteksi jadwal yang bentrok. Jadwal yang bentrok adalah jadwal dosen mengajar dengan jadwal dosen menguji dan jadwal pemakaian ruangan dengan jadwal ruangan yang lain.

Penelitian ini menggunakan metode matriks untuk menghasilkan jadwal pendadaran yang optimal sehingga tidak terjadi bentrok antara jadwal dosen mengajar dengan jadwal dosen menguji dan jadwal pemakaian ruangan dengan jadwal ruangan yang lain . Semua data akan tersimpan di dalam database MySql dan menggunakan Bahasa pemrograman PHP untuk melakukan penjadwalan tersebut.

Penelitian ini bertujuan mengembangkan aplikasipengelolaan Tugas Akhir pada program studi Teknik Informatika Universitas Ahmad Dahlan. Hasil dari penelitian ini adalah aplikasi penjadwalan pendadaran otomatis yang memudahkan koordinator tugas akhir dalam mengatur jadwal pendadaran.

Kata Kunci : Tugas Akhir, Web,Penjadwalan,Metode Matriks

BAB I **PENDAHULUAN**

## 1.1. Latar Belakang Masalah

Tugas akhir merupakan salah satu mata kuliah dalam kurikulum pendidikan perguruan tinggi yang menempati posisi sangat strategis dalam pencapaian kualitas lulusan yang memadai (Nugroho, 2016). Universitas telah mengalokasikan sumber daya mereka untuk mengelola tugas akhir seperti pendaftaran tugas akhir, penjadwalan, dan juga pelaksanaan pendadaran. Dari beberapa jurnal pengelolaan tugas akhir masih secara manual, biasanya ditulis di kertas atau ditulis di *Microsoft Excel*.

Pengelola data tugas akhir adalah koordinator tugas akhir. Koordinator tugas akhir memiliki tugas yang sama di setiap universitas. Di program studi Teknik Informatika UAD pendaftaran tugas akhir, penjadwalan seminar proposal, dan penjadwalan pendadaran masih menggunakan *Microsoft Excel*. Salah satu kelemahan menggunakan *Microsoft Excel* adalah tidak bisa mencegah atau mendeteksi jadwal yang bentrok. Jadwal yang bentrok adalah jadwal dosen mengajar dengan jadwal dosen menguji dan jadwal pemakaian ruangan  dengan jadwal ruangan yang lain.

Ada beberapa metode penjadwalan yang biasa digunakan beberapa di antaranya yaitu algoritma genetika, algoritma *Particle Swarm Optimization,* dan metode matriks. Metode matriks merupakan metode yang paling sederhana dibandingkan metode lainnya karena mudah dipahami dan langsung jelas menemukan solusinya sehingga paling cocok untuk melakukan penjadwalan tugas akhir.

Aplikasi yang sudah ada sebelumnya sudah dapat mengelola berkas tugas akhir yaitu pembagian dosen pembimbing dan pembagian jadwal pendadaran, akan tetapi aplikasi tersebut masih memiliki kekurangan yaitu penjadwalan yang dilakukan masih berpotensi menghasilkan jadwal yang bentrok karena penjadwalannya masih manual, yaitu koordinator tugas akhir masih mengecek jadwal tiap tiap anggota tim penguji. Hal ini tentu saja tidak efektif dan efisien.

Pada penelitian ini akan membuat fitur penjadwalan otomatis  menggunakan metode matriks khususnya di penjadwalan pendadaran dan penjadwalan ruangan. Fitur penjadwalan otomatis yang dibuat akan diintegrasikan ke *website* pengelolaan tugas akhir yang telah ada.

## 1.2. Identifikasi Masalah

Tugas akhir pada program studi Teknik Informatika UAD masih dikelola secara manual menggunakan *Microsoft Excel* sehingga tidak bisa mencegah atau mendeteksi jadwal yang bentrok dan menyebabkan koordinator tugas akhir sulit untuk menentukan jadwal pendadaran dan penjadwalan ruangan.

## 1.3. Batasan Masalah

Dalam penelitian ini akan difokuskan pada pengembangan aplikasi pengelolaan Tugas Akhir pada program studi Teknik Informatika UAD khususnya di bagian penjadwalan pendadaran dan penjadwal ruangan menggunakan metode matriks berbasis web dengan bahasa pemrograman PHP dan basis data MySQL.

## 1.4. Rumusan Masalah

Berdasarkan latar belakang masalah, maka didapatkan rumusan masalah yaitu bagaimana menerapkan metode matriks dalam fitur penjadwalan pendadaran dan penjadwalan ruangan sehingga dapat mengotomatisasi koordinator dalam mengelola tugas akhir dan mengurangi potensi jadwal yang bentrok.

## 1.5. Tujuan Penelitian

Tujuan dari penelitian ini adalah menerapkan metode matriks pada penjadwalan pendadaran dan penjadwalan ruangan dengan metode matriks di aplikasipengelolaan Tugas Akhir di Universitas Ahmad Dahlan.

## 1.6. Manfaat Penelitian

Manfaat yang ingin diperoleh dari penelitian ini adalah tersedianya fitur penjadwalan pendadaran dan jadwal ruangan otomatis menggunakan metode matriks.

# BAB II

**LANDASAN TEORI**

## 2.1. Kajian Pustaka

Penelitian ini mengacu pada penelitian terdahulu untuk menggali informasi baik mengenai kelebihan atau kekurangan yang sudah ada dari penelitian-penelitian sebelumnya. Selain penelitian sebelumnya, akan disajikan pula tentang kajian pendukung yang digunakan dalam penelitian ini.

Kajian terdahulu mengacu pada penelitian yang dilakukan oleh Nurdewanto (2016) yang membahas tentang sistem penjadwalan rapat pimpinan Universitas dengan metode matriks. Penelitian ini dilakukan di Universitas Merdeka Malang. Metode yang digunakan adalah metode matriks yang cara kerjanya mengubah jadwal dosen menjadi sebuah matriks kemudian diakumulasikan sehingga dapat menentukan waktu yang tepat untuk rapat. Metode matriks mudah dipahami dan paling mudah untuk diimplementasikan menjadi sebuah aplikasidengan Bahasa pemrograman PHP.

Penelitian selanjutnya dilakukan oleh Laksono, et al. (2018) yang membahas tentang sistem penjadwalan kuliah menggunakan metode algoritma genetika. Penelitian ini dilakukan di Fakultas Kedokteran dan Kesehatan Universitas Muhammadiyah Jakarta. Metode yang digunakan adalah algoritma genetika yang cara kerjanya mengubah waktu ketersediaan mengajar dosen menjadi beberapa gen dan disatukan menjadi sebuah kromosom, kemudian kromosom tersebut akan diproses menggunakan algoritma genetika sehingga mendapatkan jadwal dosen mengajar. Metode ini cocok jika digunakan dalam melakukan penjadwalan kuliah. Akan tetapi metode ini tidak cocok jika diimplementasikan dalam penjadwalan tugas akhir.

Penelitian selanjutnya dilakukan oleh Nugraha, et al. (2019) yang membahas tentang sistem penjadwalan perkuliahan menggunakan algoritma particle swarm optimization pada jurusan teknologi informasi Fakultas Teknik Universitas Tadulako. Sistem ini menggunakan algoritma *Particle Swarm Optimization* dalam melakukan penjadwalan cara kerja algoritma tersebut adalah terus mengupdate solusi hingga menemukan hasil yang optimal. Algoritma ini cocok jika diimplementasikan dalam penjadwalan kuliah. Akan tetapi algoritma ini sulit untuk diimplementasikan di Bahasa pemrograman PHP.

Penelitian selanjutnya dilakukan oleh Saifullah & Hermawan (2016) yang membahas tentang pengembangan sistem penjadwalan kuliah menggunakan algoritma *steepest ascent hill climbing*. Penelitian ini dilakukan di Fakultas Sains dan Teknologi Universitas Teknologi Yogyakata. Cara kerja algoritma *steepest ascent hill climbing* adalah melakukan penjadwalan secara acak dan menentukan *successor* yang akan dibandingkan dengan *successor* yang lain. *Successor* yang paling optimal akan dijadikan sebagai jadwal kuliah. Algoritma ini dapat menghasilkan solusi yang optimal, namun proses penjadwalan ini memakan waktu yang cukup lama karena penjadwalan benar benar dilakukan secara acak.

Penelitian selanjutnya dilakukan oleh Sugiarto, et al. (2015) yang membahas tentang penjadwalan perkuliahan otomatis berbasis *fuzzy logic* dan *genetic algorithm*. Penelitian ini dilakukan di Universitas Dian Nuswantoro. Cara kerja algoritma genetika dan *fuzzy logic* adalah menambahkan aturan *fuzzy logic* ke dalam *crossover* dan mutasi yang ada di dalam algoritma genetika yang berfungsi untuk mempercepat proses mencapai *global optimum*. Kemudian akan diproses dan mendapatkan jadwal kuliah. Dengan menambahkan fuzzy logic dapat mempercepat proses penjadwalan, Akan tetapi prosesnya tetap panjang dan susah untuk diimplementasikan menjadi sebuah program.

Penelitian ini akan menggunakan metode matriks karena algoritma ini paling mudah untuk diimplementasikan dalam penjadwalan pendadaran dan penjadwalan ruangan. Metode matriks dapat menemukan solusi secara langsung melalui matriks akumulasi yang didapat dari jadwal dosen penguji dan jadwal dosen pembimbing.

Tabel 2.1. : Perbandingan penelitian terdahulu

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Penulis** | **Nurdewanto (2016)** | **Laksono, et al. (2018)** | **Nugraha, et al. (2019)** | **Saifullah dan Hermawan (2016)** | **Sugiarto, et al. (2015)** |
| **Tujuan** | Menghasilkan sistem penjadwalan rapat pimpinan universitas dengan metode matriks | Menghasilkan sistem penjadwalan kuliah menggunakan metode algoritma genetika | Menghasilkan sistem penjadwalan perkuliahan menggunakan algoritma particle swarm optimization | Menghasilkan sistem penjadwalan kuliah menggunakan algoritma steepest ascent hill climbing | Menghasilkan sistem penjadwalan perkuliahan otomatis berbasis fuzzy logic dan genetic algorithm |
| **Metode** | Metode Matriks | Metode Algoritma genetika | Algoritma particle swarm optimization | Algoritma steepest ascent hill climbing | *Fuzzy logic dan genetic algorithm* |
| **Hasil** | Hasil dari penelitian menghasilkan aplikasi penjadwalan rapat pimpinan universitas dengan metode matriks | Hasil penelitian menghasilkan fungsi aplikasi penjadwalan kuliah | Hasil penelitian menghasilkan fungsi aplikasi penjadwalan kuliah | Hasil penelitian menghasilkan fungsi aplikasi penjadwalan kuliah | Hasil penelitian menghasilkan fungsi aplikasi penjadwalan kuliah |

## 2.2. Kajian Teori

### 2.2.1. Tugas Akhir

Tugas akhir adalah karya tulis ilmiah yang menjadi salah satu syarat kelulusan. Universitas telah mengalokasikan sumber daya mereka untuk mengelola tugas akhir seperti pendaftaran tugas akhir, penjadwalan, dan juga pelaksanaan pendadaran. Dari beberapa jurnal pengelolaan tugas akhir masih secara manual, biasanya ditulis di kertas atau ditulis di *Microsoft Excel*. Tahapan tahapan dalam menjalankan tugas akhir adalah memilih dosen pembimbing, pengajuan proposal, seminar proposal, seminar tugas akhir, dan revisi.

### 2.2.2. Penjadwalan

Penjadwalan adalah proses perencanaan waktu untuk setiap sumber daya agar kegiatan yang dilakukan terstruktur dan berjalan dengan lancar. Penjadwalan tugas akhir sangat diperlukan dikarenakan keterbatasan ruangan, jadwal dosen penguji dan dosen pembimbing yang berbeda sehingga perlu dilakukan penjadwalan.

### 2.2.3. Metode Matriks

Metode matriks adalah metode yang memungkinkan kita mendapatkan solusi paling optimal berdasarkan nilai terkecil dari elemen – elemen matriks yang telah diakumulasikan.

Cara kerja metode matriks adalah mengubah data menjadi sebuah matriks. Contohnya seperti jadwal dosen mengajar pada Tabel 2.2.

Tabel 2.2. : Tabel jadwal dosen mengajar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Hari  Jam | Senin | Selasa | Rabu | Kamis | Jumat | Sabtu |
| 7.00 | 1 | 0 | 0 | 0 | 0 | 0 |
| 9.00 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11.00 | 0 | 0 | 1 | 0 | 0 | 0 |
| 13.00 | 0 | 1 | 0 | 0 | 0 | 1 |

Angka 1 pada Tabel 2.2 berarti dosen tersebut mengajar pada waktu tersebut. Sedangkan angka 0 berarti dosen tidak mengajar pada waktu tersebut.

Kemudian data tersebut diubah menjadi matriks dengan ordo 4 X 6 seperti berikut:

Dalam penjadwalan pendadaran di aplikasi ini dibutuhkan 6 matriks yang didapat dari jadwal mengajar dan jadwal menguji dosen pembimbing, dosen penguji 1, dan dosen penguji 2.

Matriks diatas adalah matriks dari jadwal dosen pembimbing yang bernama Ardiansyah, S.T., M.Cs. yang mengajar pada jam ke 1,2,3,7,8, dan 9 di hari selasa dan jam ke 1,2, dan 3 di hari kamis.

Matriks diatas adalah matriks dari jadwal dosen penguji 1 yang bernama Supriyanto, S.T., M.T. yang mengajar pada jam ke 11 dan 12 di hari selasa, jam ke 1,2,11, dan 12 di hari rabu, jam ke 7 dan 8 di hari kamis, dan jam ke 1,2,5, dan 6 di hari sabtu.

Matriks diatas adalah matriks dari jadwal dosen penguji 2 yang bernama Murinto, S.Si., M.Kom. yang mengajar pada jam ke 3,4,5,6,11, dan 12 di hari senin dan jam ke 10,11, dan 12 di hari jumat.

Matriks diatas adalah matriks akumulasi hasil dari akumulasi matriks jadwal dosen pembimbing ,dosen penguji 1, dan dosen penguji 2. Matriks jadwal pendadaran diasumsikan masih kosong sehingga tidak perlu ditambahkan.

Dari matriks akumulasi yang semula ber ordo 6 x 15 disederhanakan menjadi ber ordo 6 x 11 mengikuti jadwal pendadaran yang hanya sampai pada jam ke 11.

Prinsip metode matriks yang digunakan dalam mencari jadwal pendadaran ini adalah jika dalam suatu baris dalam matriks terdapat nilai 0 secara berurutan berjumlah 3 kali, maka di jam tersebut dapat disimpulkan bahwa ketiga dosen tersebut tidak mengajar dan jam tersebut dapat dijadikan rekomendasi pendadaran.

Dari matriks akumulasi yang telah diubah, metode matriks mendapatkan beberapa hasil yang dapat dijadikan sebagai rekomendasi jadwal penadadaran sebagai berikut.

Tabel 2.3. : Tabel hasil rekomendasi

|  |  |  |
| --- | --- | --- |
| No | Hari | Jam |
| 1 | Senin | (7,8,9) dan (8.9.10) |
| 2 | Selasa | (4,5,6) |
| 3 | Rabu | (3,4,5), (4,5,6), (5,6,7), (6,7,8), (7,8,9) dan (8,9,10) |
| 4 | Kamis | (4,5,6) dan (9,10,11) |
| 5 | Jumat | (1,2,3), (2,3,4), (3,4,5), (4,5,6), (5,6,7), (6,7,8) dan (7.8.9) |
| 6 | Sabtu | (6,7,8),(7,8,9),(8,9,10) dan (9,10,11) |

# BAB III

**METODOLOGI PENELITIAN**

## 3.1. Obyek Penelitian

Obyek yang akan dibahas pada penilitian ini adalah pengelolaan Tugas Akhir pada Program Studi Teknik Informatika Universitas Ahmad Dahlan.

## 3.2. Metode Pengumpulan Data

Metode pengumpulan data dilakukan untuk mengumpulkan data-data yang dibutuhkan untuk penelitian. Metode yang digunakan pada penelitian adalah studi dokumen.

Metode ini dilakukan untuk mendapatkan rincian dari obyek penelitian. Dokumen yang digunakan pada penelitian ini adalah dokumen yang didapat dari koordinator tugas akhir Teknik Informatika UAD dan juga jadwal dosen yang di dapat dari website SIMERU UAD.

## 3.3. Alat Penelitian

Kebutuhan Perangkat Lunak dan Perangkat Keras yang digunakan pada penilitian ini sebagai berikut :

### 3.3.1. Perangkat Keras

Spesifikasi perangkat keras yang digunakan pada penlitian ini adalah :

* 1. Laptop Asus dengan spesifikasi :
     + - 1. Intel Core i5-7200U CPU @ 2.50GHz
         2. RAM 4GB DDR4 Memory
  2. Mouse
  3. Wifi
  4. Flashdisk 16 GB

### 3.3.2. Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan pada penilitian ini adalah :

1. Sistem Operasi Windows 10
2. Browser Google Chrome
3. Microsoft Word 2016
4. Sublime Text 3
5. XAMPP V3.2.2
6. phpMyAdmin 5.0.2
7. Bootstrap
8. jQuery

## 3.4. Metode Penelitian

Pada penelitian ini menggunakan beberapa tahapan:

1. Tahapan Analisis Sistem

Pada tahapan analisis sitem dilakukan identifikasi kebutuhan pengguna dalam aplikasi ini. Identifikasi kebutuhan aplikasi disusun menjadi 2 bagian yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional merupakan kebutuhan yang berisikan fungsi yang dapat dilakukan oleh sistem.Kebutuhan non-fungsional merupakan kebutuhan yang mendukung fungsi dari suatu sistem.

1. Desain Aplikasi

Pada tahapan ini dilakukan pembuatan model dari aplikasi. Terdapat beberapa tahapan dalam melakukan desain aplikasi yaitu model pemakaian, model domain dan model antarmuka. Model pemakaian digunakan untuk mengetahui alur sistem yang digunakan oleh user. Model yang digunakan adalah user stories. User Stories adalah gambaran fitur yang terdapat dalam sistem yang disusun bertingkat berdasarkan prioritasnya. Model domain digunakan untuk mengetahui tipe entitas dasar dari konseptual sistem. Pemodelan yang digunakan dalam aplikasi ini adalah Class Diagram. Model antarmuka digunakan untuk menyesuaikan keinginan pemilik aplikasi dengan yang akan dibuat oleh pengembang dalam bentuk sketsa, purwarupa kertas, dan purwarupa nyata.

1. Tahapan Implementasi

Pada tahapan ini fitur tersebut akan dijadikan Bahasa yang dapat dimengerti komputer. Maka fitur tersebut akan diimplementasikan pada pemrograman web yang menggunakan Bahasa pemrograman PHP.

1. Tahapan Pengujian

Pada tahapan ini pengujian akan dilakukan menggunakan metode *black box testing*. Pengujian ini akan menentukan apakah *input* dan *output* pada aplikasi penjadwalan pendadaran telah melakukan apa yang seharusnya dilakukan sesuai dengan kebutuhan fungsional yang telah ada.

**BAB IV**  
HASIL DAN PEMBAHASAN

## Analisis Sistem

1. **Analisis Kebutuhan Pengguna**
2. Pengguna dapat mengelola jadwal dosen.
3. Pengguna dapat memilih jadwal pendadaran berdasarkan hasil rekomendasi.
4. Pengguna dapat memilih tanggal sesuai dengan yang diinginkan.
5. Pengguna dapat memilih ruangan sesuai dengan yang direkomendasikan.
6. Pengguna dapat melakukan pendaftaran pendadaran.
7. **Analisis Kebutuhan Fungsional**
   * + 1. Mengelola Data Dosen
   1. Sistem mampu menyimpan jadwal Dosen.
   2. Sistem mampu menyimpan perubahan jadwal Dosen.
      * 1. Merekomendasikan jadwal pendadaran
8. Sistem mampu merekomendasikan jadwal pendadaran berdasarkan jadwal dosen dan jadwal pendadaran yang ada.
   * + 1. Merekomendasikan ruangan pendadaran
9. Sistem mampu merekomendasikan ruangan pendadaran berdasarkan jadwal pendadaran yang telah ada.
   * + 1. Menambahkan Data Mahasiswa Pendadaran
10. Sistem mampu menampilkan data ( nama, judul tugas akhir, pembimbing, penguji 1 dan penguji 2) dari mahasiswa yang mendaftar pendadaran.
11. Sistem mampu menampilkan pilihan ruang untuk melakukan ujian pendadaran.
12. Sistem mampu menampilkan pilihan jam yang telah direkomendasikan untuk melakukan ujian pendadaran.
13. Sistem mampu menyimpan informasi mahasiswa yang akan melakukan ujian pendadaran.
    * + 1. Sistem mampu mendeteksi tanggal merah sehingga tidak bisa memilih tanggal tersebut untuk melakukan ujian pendadaran.
        2. Sistem mampu merekomendasikan jam mulai pendadaran dari jam 08.00 - 15.00.
        3. Sistem mampu merekomendasikan 7 hari selepas pendaftaran pendadaran sebagai hari jadwal ujian pendadaran.
        4. Sistem mampu mengirimkan jadwal pendadaran ke email tim penguji dan mahasiswa yang terlibat.
        5. Merubah jadwal pendadaran
           1. Sistem mampu merubah jadwal pendadaran melalui email yang dikirimkan
           2. Sistem mampu membatasi perubahan jadwal pendadaran 2 hari sebelum ujian dimulai

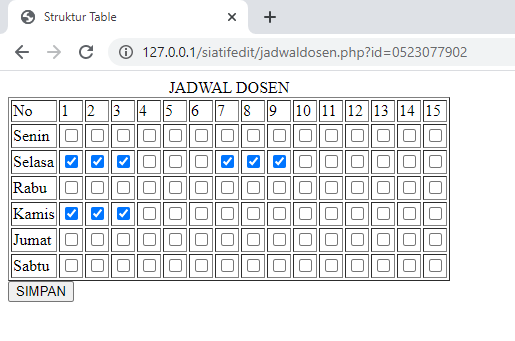
## Perancangan Sistem

Perancangan perangkat lunak merupakan tahapan yang berfungsi untuk memperoleh hasil aplikasi yang sesuai dengan kebutuhan dan kegunaan pengguna.

Pengguna aplikasi ini merupakan koordinator tugas akhir program studi Teknik Informatika. Aplikasi pengelolaan tugas akhir ini berbasis *website* dan dapat diakses melalui *web browser* apapun.

* 1. Data Jadwal Mengajar

Jadwal mengajar dosen akan dijadikan sebagai data yang nantinya akan diproses pada metode matriks, ketika dosen mengajar maka akan diberi tanda centang di setiap *checkbox* yang tersedia seperti pada Gambar 4.1 di bawah.



Gambar 4.1. : Jadwal Dosen

Gambar 4.1 menunjukkan bahwa dosen mengajar pada jam ke 1,2,3,7,8, dan 9 di hari selasa dan jam ke 1,2, dan 3 di hari kamis.

* 1. *Mengubah jadwal menjadi matriks*

Untuk dapat menjalankan algoritma matiks jadwal dosen seperti pada Gambar 4.1 harus diubah menjadi matriks terlebih dahulu.

Baris matriks diatas menandakan banyaka jumlah hari yang digunakan untuk mengajar sedangkan kolom matriks diatas menandakan banyaknya jam pelajaran dalam satu hari.

Nilai dari maktriks didapat dari jadwal dosen mengajar, matriks akan bernilai 1 jika dosen ada jadwal mengajar, sedangkan jika dosen tidak ada jadwal mengajar maka akan bernilai 0.

* 1. *Membuat matriks akumulasi*

Untuk mencari jadwal yang kosong menggunakan metode matriks dibutuhkan matriks akumulasi yang di dapat dari jadwal mengajar dosen pembimbing, penguji 1, dan penguji 2.

Matriks diatas adalah matriks dari jadwal dosen pembimbing yang mengajar pada jam ke 1,2,3,7,8, dan 9 di hari selasa dan jam ke 1,2, dan 3 di hari kamis.

Matriks diatas adalah matriks dari jadwal dosen penguji 1 yang mengajar pada jam ke 11 dan 12 di hari selasa, jam ke 1,2,11, dan 12 di hari rabu, jam ke 7 dan 8 di hari kamis, dan jam ke 1,2,5, dan 6 di hari sabtu.

Matriks diatas adalah matriks dari jadwal dosen penguji 2 yang mengajar pada jam ke 3,4,5,6,11, dan 12 di hari senin dan jam ke 10,11, dan 12 di hari jumat.

Matriks diatas adalah matriks akumulasi hasil dari akumulasi matriks jadwal dosen pembimbing ,dosen penguji 1, dan dosen penguji 2.

* 1. Mencari jadwal kosong

Prinsip metode matriks yang digunakan dalam mencari jadwal pendadaran ini adalah jika dalam suatu baris dalam matriks terdapat nilai 0 secara berurutan berjumlah 3 kali, maka di jam tersebut dapat disimpulkan bahwa ketiga dosen tersebut tidak mengajar dan jam tersebut dapat dijadikan rekomendasi pendadaran.

Dari matriks akumulasi yang semula ber ordo 6 x 15 disederhanakan menjadi ber ordo 6 x 11 mengikuti jadwal pendadaran yang hanya sampai pada jam ke 11.

Dari matriks akumulasi yang telah diubah, metode matriks mendapatkan beberapa hasil yang dapat dijadikan sebagai rekomendasi jadwal penadadaran sebagai berikut.

Tabel 4.1. : Tabel jadwal rekomendasi

|  |  |  |
| --- | --- | --- |
| No | Hari | Jam |
| 1 | Senin | (7,8,9) dan (8.9.10) |
| 2 | Selasa | (4,5,6) |
| 3 | Rabu | (3,4,5), (4,5,6), (5,6,7), (6,7,8), (7,8,9) dan (8,9,10) |
| 4 | Kamis | (4,5,6) dan (9,10,11) |
| 5 | Jumat | (1,2,3), (2,3,4), (3,4,5), (4,5,6), (5,6,7), (6,7,8) dan (7.8.9) |
| 6 | Sabtu | (6,7,8),(7,8,9),(8,9,10) dan (9,10,11) |

* 1. Rancangan User Interface

Rancangan *user interface* merupakan rancangan tampilan yang akan ditampilkan didalam aplikasi pengelolaan tugas akhir. Tampilan yang ada dalam aplikasi tersebut antara lain :

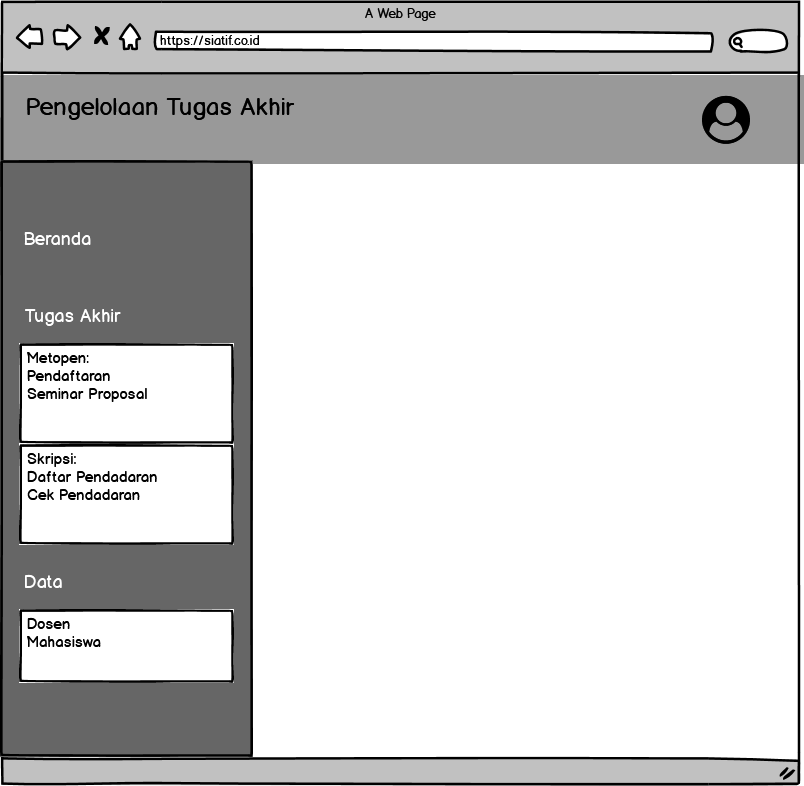
* + - 1. Rancangan Halaman *Login*



Gambar 4.2. : Rancangan Halaman Login

Gambar 4.2 merupakan rancangan halaman login. Pengguna diharuskan melakukan login pada halaman ini terlebih dahulu sebelum dapat menggunakan fitur didalam halaman utama.

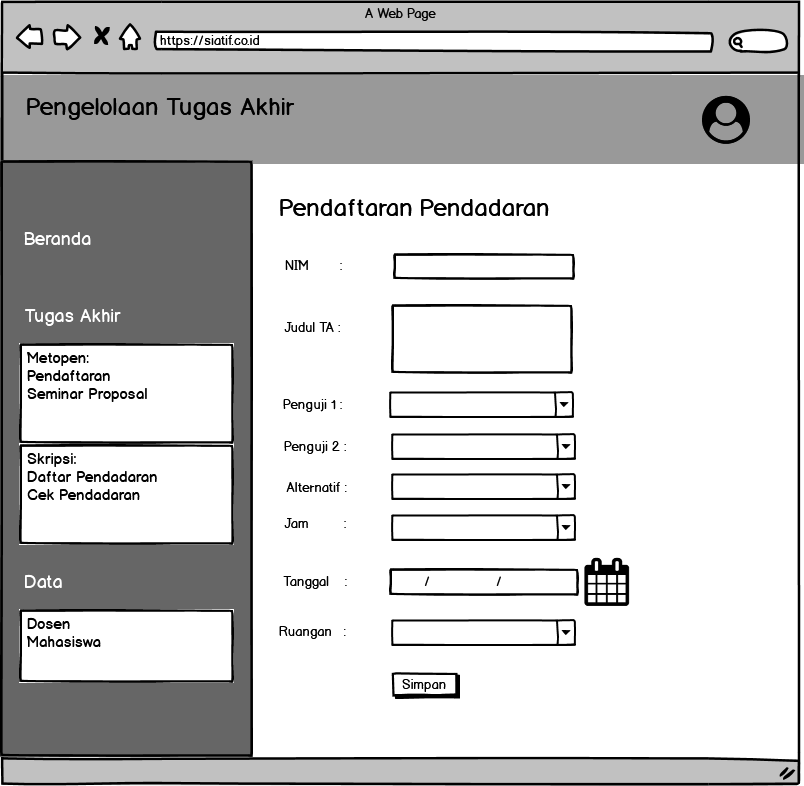
* + - 1. Rancangan Daftar Menu



Gambar 4.3. : Rancangan daftar menu

Gambar 4.3. merupakan rancangan daftar menu yang terdapat didalam sistem. Terdapat beberapa menu yang dapat digunakan oleh pengguna, yaitu menu tugas akhir dan menu data. Menu tugas akhir memiliki submenu yaitu pendaftaran metopen, seminar proposal, pendaftaran pendadaran, dan cek pendadaran. Sedangkan menu data memiliki submenu yaitu data mahasiswa dan data dosen.

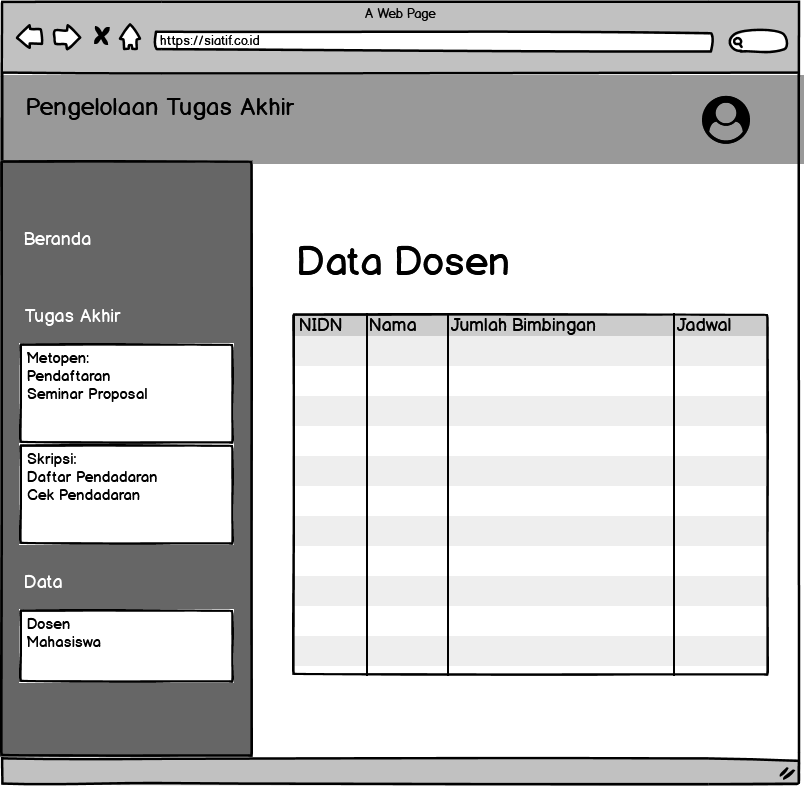
* + - 1. Rancangan Halaman Pendaftaran Pendadaran



Gambar 4. 4. : Rancangan halaman daftar pendadaran

Gambar 4.4. merupakan rancangan halaman daftar pendadaran untuk memasukkan data mahasiswa. Pengguna memasukkan nim dari mahasiswa yang mendaftar pendadaran untuk memunculkan data nama mahasiswa, judul, pembimbing, dan penguji 1 dari mahasiswa tersebut. Pada halaman ini pengguna juga dapat memilih penguji 2 sesuai hasil rekomendasi yang didapatkan serta mengatur jadwal pendadaran sesuai rekomendasi yang diberikan.

* + - 1. Rancangan Halaman Daftar Dosen

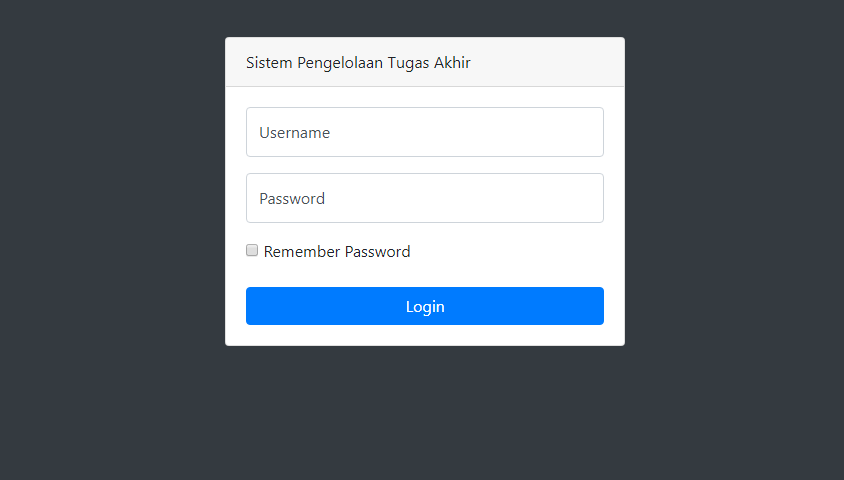


Gambar 4.5. : Rancangan halaman data dosen

Gambar 4.5. merupakan rancangan halaman data dosen untuk melihat data dosen berupa NIDN, nama, jumlah bimbingan dan jadwal yang sudah terdaftar di sistem.

## Implementasi

1. *Login*



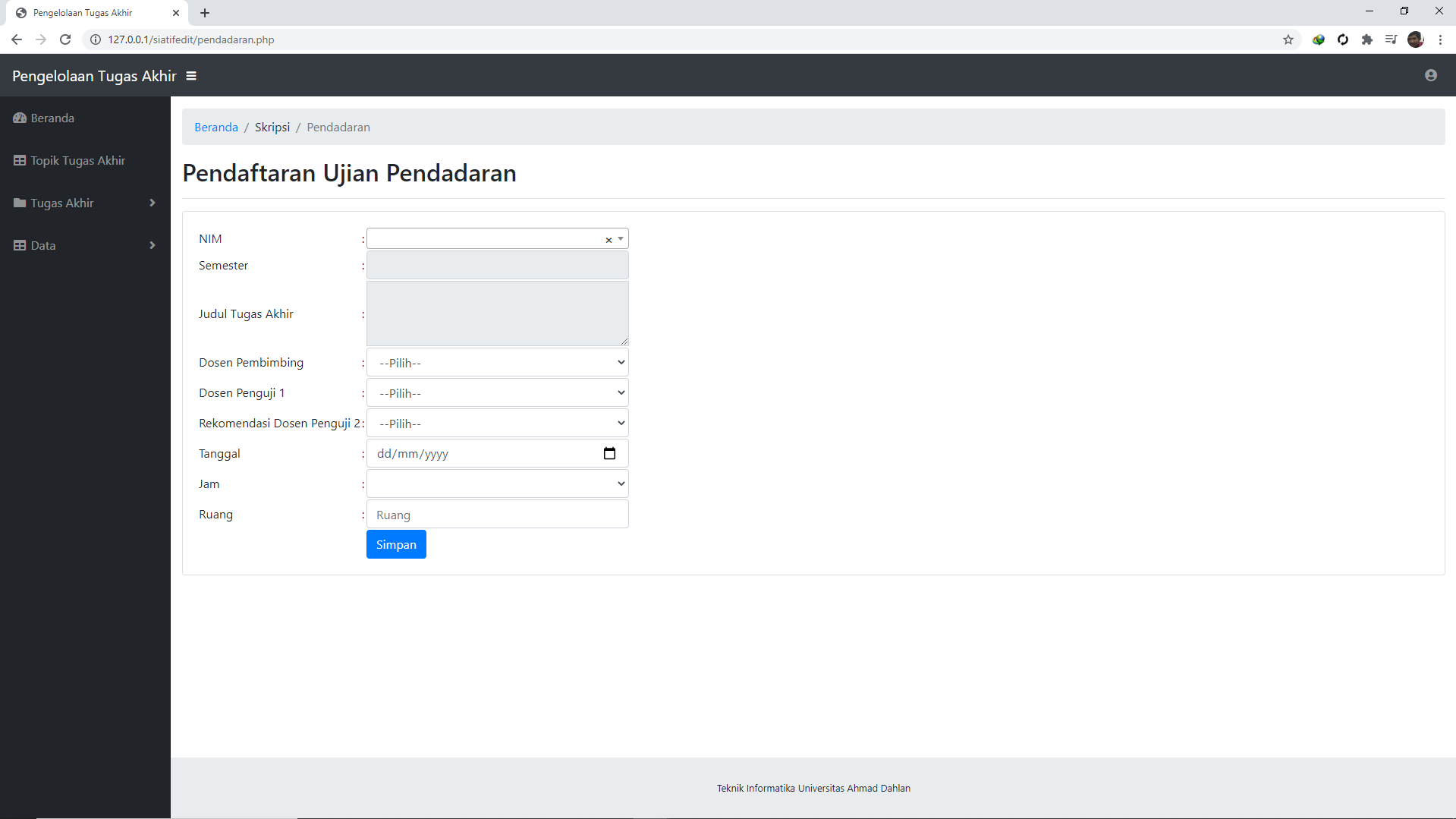
Gambar 4.6. : Halaman Login Pengguna

Halaman login menyediakan dua form *input* untuk memasukkan *username* dan *password*. Form *input* tersebut akan mengirim dan memeriksa *username* dan *password* didalam database. *Query* yang digunakan untuk proses login dapat dilihat pada *Script* 4.1. yaitu :

|  |
| --- |
| "SELECT \* FROM admin WHERE username='$user' AND password='$pass'" |

Listing 4.1. : Query login pengguna

1. Kelola Data Pendadaran



Gambar 4. 7. : Halaman Pendaftaran Pendadaran

Fungsi pada halaman pendaftaran pendadaran dapat dilihat seperti berikut :

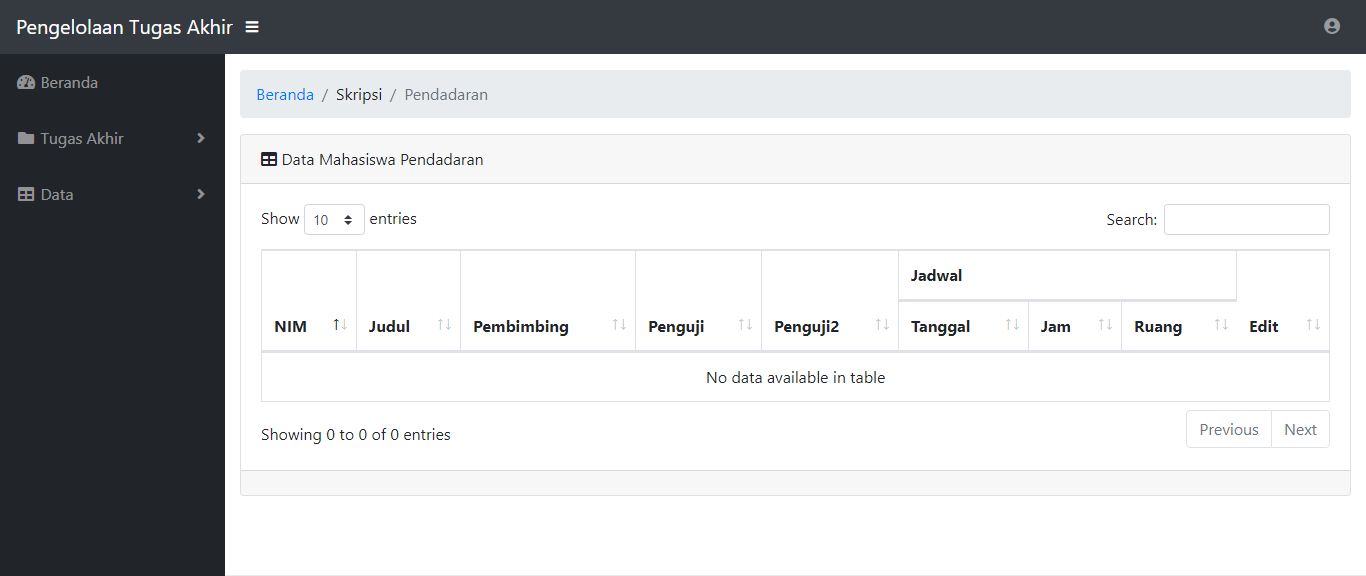
|  |
| --- |
| $.ajax({  type: "GET",  url: "l\_jadwaltomatrik.php",  data: {dosen1:val\_seleksi1,dosen2:val\_seleksi2,  dosen3:val\_seleksi3,tanggal:val\_tanggal},  cache: false,  success: function(data){  var tes = JSON.parse(data);  var $el = $('#jam');  var prevValue = $el.val();  $el.empty();  $.each(tes, function(key, value) {  var hari\_tampil = (value.hari==0) ? "senin" :  (value.hari==1) ? "selasa" : (value.hari==2) ? "rabu" :  (value.hari==3) ? "kamis" : (value.hari==4) ? "jumat" :  (value.hari==5) ? "sabtu" : "something wrong";  $el.append($('<option></option>').attr  ('value', value.hari\_waktu\_awal+' - '+value.hari\_waktu\_akhir)  .text(hari\_tampil+" "+value.hari\_waktu\_awal+'-'+value.hari\_waktu\_akhir));                  if (value === prevValue){                      $el.val(value);                  }              });   } }); |

Listing 4.2 : Script rekomendasi pendadaran

|  |
| --- |
| “UPDATE skripsi SET penguji1='$pe1' ,penguji2='$penguji2' WHERE nim='$nim'" |
| "INSERT INTO pendadaran (nim,tanggal,jam,ruang) VALUES  ('$nim','$tanggal','$jadwal','$ruang')"; |

Listing 4.3 : Query pendaftaran pendadaran

Fitur pendaftaran pendadaran dimulai dari *Script 4.2*. dengan mengambil hasil rekomendasi jam pendadaran dari *l\_jadwaltomatrik.php*. Setelah itu, diteruskan dengan mengisi data sesuai form pendaftaran dan akan disimpan melalui *Script 4.3.* dimana ada dua query disana. Query *update* untuk mengubah data penguji 2 dengan *value* sesuai isi form dan Query *Insert* untuk memasukkan jadwal pendadaran kedalam *database*.



Gambar 4.8. : Halaman Data Pendadaran

Halaman data pendadaran menampilkan data nim, judul, pembimbing, penguji 1 , penguji 2, dan jadwal pendadaran. Kolom-kolom tersebut didapatkan dari tabel dengan menggunakan query yaitu :

|  |
| --- |
| "SELECT skripsi.\* ,dosen.\* FROM skripsi, dosen WHERE dosen.nidn=skripsi.pembimbing AND skripsi.status='Skripsi' " |

Listing 4. 4: Query data pendadaran

## Pengujian Fungsional

Pengujian sistem menggunakan metode *black box*. Pengujian dilakukan untuk mengetahui *output* dari setiap *input*. Jika seluruh *output* dan fasilitas program berjalan dan tidak terjadi kesalahan maka program dianggap baik.

Tabel 4.2. :Daftar Tabel Pertanyaan Black Box Test

|  |  |  |  |
| --- | --- | --- | --- |
| No | Action | Expected Result(Output) | Actual Result |
| 1 | Mengelola data dosen | Dapat mengubah jadwal dosen | Lulus |
| 2 | Rekomendasi waktu pendadaran | Tampil hasil rekomendasi waktu pendadaran | Lulus |
| 3 | Rekomendasi ruangan pendadaran | Tampil hasil rekomendasi ruangan pendadaran | Lulus |
| 4 | Menyimpan data pendadaran | Tampil data pendadaran | Lulus |
| 5 | Mengubah jadwal pendadaran | Dapat merubah jadwal pendadaran di database | Lulus |
| 6 | Memilih tanggal merah sebagai waktu pendadaran | Tidak dapat memilih tanggal merah | Lulus |
| 7 | Merekomendasikan jam mulai pendadaran di jam 08.00 – 15.00 | Tampil rekomendasi waktu mulai pendadaran dari jam 08.00 – 15.00 | Lulus |
| 8 | Merekomendasikan 7 hari selepas pendaftaran pendadaran sebagai hari jadwal ujian pendadaran | Tampil tanggal dengan tambahan 7 hari setelah tanggal pendaftaran ujian | Lulus |
| 9 | Mengirimkan jadwal pendadaran ke email tim penguji dan mahasiswa | Email diterima oleh penguji dan mahasiswa yang terlibat | Lulus |
| 10 | Menjadwalkan ulang pendadaran | Dapat menjadwalkan ulang pendadaran melalui email | Lulus |
| 11 | Mengubah jadwal ujian 2 hari sebelum pendadaran | Tidak dapat merubah jadwal ketika jadwal pendadaran kurang dari 2 hari lagi | Lulus |

BAB V  
**KESIMPULAN DAN SARAN**

## Kesimpulan

Berdasarkan hasil dan pembahasan dari penelitian ini, maka dapat disimpulkan bahwa :

1. Telah dikembangkan aplikasi pengelolaan tugas akhir di program studi teknik informatika Universitas Ahmad Dahlan. Aplikasi ini telah mampu merekomendasikan jadwal pendadaran secara otomatis.
2. Telah dikembangkan aplikasi pengelolaan tugas akhir yang dapat mengirimkan jadwal pendadaran melalui email dosen dan mahasiswa yang terlibat. Jadwal yang paling cepat adalah 7 hari selepas pendaftaran ujian pendadaran. Di dalam email terdapat tautan yang mengarahkan dosen ke menu penjadwalan ulang akan tetapi fitur tersebut hanya bisa digunakan maksimal 3 hari sebelum pendadaran.
3. Hasil pengujian unit dan sistem menunjukkan semua fungsi sudah diuji dan sesuai dengan *requirement.*

## Saran

Adapun saran dari penelitian adalah sebagai berikut:

1. Penambahan fitur penjadwalan seminar proposal secara otomatis. Penjadwalan otomatis tersebut dapat dikembangkan salah satunya dengan metode matriks.
2. Penambahan fitur agar dapat sinkron dengan google calendar. Sehingga jadwal yang dibuat bisa lebih akurat lagi bukan hanya dari jadwal dosen mengajar tetapi juga dari agenda dosen diluar jam mengajar.

# DAFTAR PUSTAKA

Laksono, A., Utami, M., & Sugiarti, Y. (2018). Sistem Penjadwalan Kuliah Menggunakan Metode Algoritma Genetika (Studi Kasus: Fakultas Kedokteran Dan Kesehatan Universitas Muhammadiyah Jakarta). *Studia Informatika: Jurnal Sistem Informasi*, *9*(2), 177–188. https://doi.org/10.15408/sijsi.v9i2.7647

Nugraha, D. W., Dodu, A. Y. E., & Paloloang, M. F. B. (2019). Sistem Penjadwalan Perkuliahan Menggunakan Algoritma Particle Swarm Optimization Pada Jurusan Teknologi Informasi Fakultas Teknik Universitas Tadulako. *ScientiCO : Computer Science and Informatics Journal*, *2*(1), 1. https://doi.org/10.22487/j26204118.2019.v2.i1.13067

Nugroho, B. (2016). Sistem Informasi Pengelolaan Administrasi. *Tugas Akhir Di Jurusan Teknik Informatika*, *9*(1), 1–10.

Nurdewanto, B. (2016). Sistem Penjadwalan Rapat Pimpinan Universitas Dengan Metode Matriks. *Jurnal Teknologi Dan Manajemen Informatika*, *2*(1). https://doi.org/10.26905/jtmi.v2i1.614

Saifullah, S., & Hermawan, A. (2016). Pengembangan Sistem Penjadwalan Kuliah Menggunakan Algoritma Steepest Ascent Hill Climbing. *Jurnal Sistem Komputer*, *6*(2), 57–62.

Sugiarto, E., Winarno, S., & Fahmi, A. (2015). Penjadwalan Perkuliahan Otomatis Berbasis Fuzzy Logic Dan Genetic Algorithm. *Techno.COM*, *14*(4), 315–328. Retrieved from https://pdfs.semanticscholar.org/2642/3aac64989d49844af01364e684b3f7f3d3b2.pdf