

SKRIPSI

APLIKASI PEMBANGKIT JADWAL DOSEN



Adrian Reynaldi

NPM: 2013730058

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2017**

UNDERGRADUATE THESIS

APLIKASI PEMBANGKIT JADWAL DOSEN



Adrian Reynaldi

NPM: 2013730058

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2017**

ABSTRAK

Penggunaan kertas oleh manusia sudah berlangsung selama ribuan tahun. Kemudahannya untuk disimpan, dibawa dan biaya pembuatannya yang relatif rendah membuat kertas sebagai alat yang ideal untuk mencatat berbagai informasi. Namun, dengan munculnya isu *global warming* membuat penggunaan kertas semakin tidak menarik karena penebangan pohon secara massal (yang sebagian diantaranya dipakai untuk bahan pembuatan kertas) diindikasikan merupakan salah satu penyebab utama terjadinya *global warming*. Hal ini membuat berbagai kalangan mulai beralih ke media digital untuk mencatat informasi agar mengurangi penggunaan kertas. Selain itu, pengalihan dari kertas ke media digital juga berguna untuk menekan biaya operasional jangka panjang.

Fakultas Teknologi Informasi dan Sains di Universitas Katolik Parahyangan mengembangkan aplikasi BlueTape sebagai salah satu cara untuk mengurangi pekerjaan *paper-based* menjadi *paperless*. Aplikasi yang sudah dikembangkan sejak tahun 2015 ini telah berhasil membuat permintaan dan pengaturan transkrip antar mahasiswa dan tata usaha FTIS UNPAR menjadi tidak perlu lagi menggunakan kertas, melainkan cukup menggunakan BlueTape. Saat skripsi ini ditulis, aplikasi ini masih dikembangkan untuk menambah berbagai macam fitur lain. Salah satu diantaranya adalah sistem pencatatan jadwal dosen.

Pencatatan dan publikasi jadwal dosen saat skripsi ini ditulis masih menggunakan kertas. Jadwal dicetak sendiri oleh masing-masing dosen dan ditempel di depan ruangnya. Metode seperti ini selain menghaburkan kertas, juga tidak efektif karena mahasiswa perlu datang ke ruangan dosen masing-masing untuk mengetahui jadwal dosen terkait. Skripsi ini bertujuan untuk mengatasi hal ini dengan merancang Aplikasi Pembangkit Jadwal Dosen menggunakan BlueTape sebagai basis *framework*-nya. Aplikasi ini bertujuan agar dosen dapat mencatat dan menampilkan jadwalnya di BlueTape sehingga mahasiswa bisa mengetahui jadwal dosen dari manapun dan secara tidak langsung mengurangi penggunaan kertas karena dosen tidak perlu lagi mencetak jadwalnya masing-masing.

Kata-kata kunci: kertas, BlueTape, tata usaha, jadwal dosen, Aplikasi Pembangkit Jadwal Dosen, *paperless*

ABSTRACT

The use of paper has been going on for hundreds of years. It's easy storage, mobility and relatively low production cost makes it ideal device to write all kind of informations. However, the emergence of global warming issue makes paper less appealing to use due to massive tree logging (because some of it are being used to create papers) has been indicated as one of the main causes of global warming. This makes various organisations and individuals shifted to digital media for information writing as to reduce paper use. Furthermore, changing from paper to digital media also reduce long-term operational costs.

Faculty of Information Technology and Science in Parahyangan Catholic University developed BlueTape as one of the ways to reduce paper-based work into paperless one. The application that has been developed since 2015 has manage to make transcript request and management between college students and the administrator of FTIS UNPAR paperless by using BlueTape. When this thesis is still being written, this application is still being developed to add other features. One of them is a system to record lecturer's schedules.

Currently when this thesis is being written the record and publication of lecturer's schedules is still done by paper. The schedule is printed by the lecturer's themselves and then placed just outside of their own room. Beside wasting a lot of papers, this method is also ineffective because students need to go to the lecturer room to know their schedule. This thesis purpose is to solve this issue by designing *Aplikasi Pembangkit Jadwal Dosen* or Lecturer's Schedule Generator Application with BlueTape as it's framework. This application is developed to help lecturers to record and publish their schedule in BlueTape so students can access their schedule anywhere and indirectly reduce paper use because the lecturers doesn't need to print their own schedule anymore.

Keywords: paper, BlueTape, administrator, lecturer's schedule, *Aplikasi Pembangkit Jadwal Dosen*, paperless

DAFTAR ISI

DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	3
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 BlueTape	5
2.2 CodeIgniter	5
2.2.1 <i>System Requirements</i> Server Untuk Menjalankan CodeIgniter	6
2.2.2 Cara Instalasi CodeIgniter	6
2.2.3 Cross-site Request Forgery (CSRF)	7
2.2.4 Model-View-Controller	7
2.2.5 Controller	7
2.2.6 View	9
2.2.7 Model	10
2.2.8 Kelas Migrasi	11
2.2.9 Flow Chart Codeigniter	12
2.3 Cross Site Request Forgery	13
2.4 Zurb Foundation	14
2.4.1 Top Bar	14
2.4.2 Reveal Modal	14
2.4.3 Variabel Sass <i>Reveal Modal</i>	15
2.4.4 Scrolling Table	15
2.5 Google OAuth 2.0	18
2.6 PHPExcel	19
3 ANALISIS	25
3.1 Analisis Sistem Kini	25
3.1.1 <i>Transcript Request / Manage</i>	26
3.1.2 Perubahan Kuliah	28
3.1.3 Pengguna Aplikasi	29
3.1.4 Diagram Use Case Sistem Kini	30
3.2 Analisis Sistem Usulan: Aplikasi Pembangkit Jadwal Dosen	33
3.2.1 Diagram Use Case Sistem Usulan	34

3.2.2	Pengguna Aplikasi	36
4	PERANCANGAN DAN IMPLEMENTASI	39
4.1	Perancangan Basis Data	39
4.1.1	Diagram Hubungan Entitas	39
4.1.2	Perancangan Tabel	40
4.1.3	Perancangan Rinci	40
4.2	Perancangan Antarmuka	44
4.2.1	Perancangan Antarmuka Entri Jadwal Dosen	44
4.2.2	Perancangan Antarmuka Edit Jadwal Dosen	46
4.2.3	Perancangan Antarmuka Lihat Jadwal Dosen	48
5	IMPLEMENTASI DAN PENGUJIAN	49
5.1	Implementasi	49
5.1.1	Implementasi Basis Data	49
5.1.2	Implementasi Kelas	50
5.2	Pengujian	50
5.2.1	Pengujian Fungsional	50
5.3	Penugjian Ekperimental	53
5.3.1	Pengujian Eksperimental Pembuatan File Spreadsheet Menggunakan Tipe Extensi File .xlsx	53
5.3.2	Pengujian Eksperimental Penambahan Tombol <i>Delete All</i>	56
5.3.3	Pengujian Eksperimental Penambahan Informasi Waktu <i>Update Terakhir</i> Jadwal	59
5.3.4	Pengujian Eksperimental <i>Conflict Handler</i>	61
6	KESIMPULAN DAN SARAN	67
6.1	Kesimpulan	67
6.2	Saran	67
	DAFTAR REFERENSI	69
A	KODE PROGRAM IMPLEMENTASI MODUL ENTRI JADWAL DOSEN	71
B	KODE PROGRAM IMPLEMENTASI MODUL LIHAT JADWAL DOSEN	77
C	KODE PROGRAM MODEL JADWALDOSEN	83
D	KODE PROGRAM UNTUK LOGIN	85
E	KODE PROGRAM UNTUK MIGRASI	87

DAFTAR GAMBAR

1.1	Template jadwal dosen	1
2.1	Flow Chart Codeigniter	12
2.2	Skenario OAuth menggunakan token akses	18
2.3	Arsitektur PHPExcel	19
3.1	Halaman Utama BlueTape Untuk Login dan Membuka Petunjuk Penggunaan . . .	26
3.2	Tampilan Cetak Transkrip	27
3.3	Tampilan Manajemen Transkrip	27
3.4	Tampilan Menu Perubahan Kuliah	28
3.5	Tampilan Menu Manajemen Perubahan Kuliah	29
3.6	Diagram Use Case Sistem Kini	30
3.7	Diagram Use Case Sistem Usulan	34
4.1	Diagram Hubungan Entitas	39
4.2	Perancangan Antarmuka Entri Jadwal Dosen	45
4.3	Contoh Jadwal	46
4.4	Perancangan Antarmuka Edit Jadwal Dosen	47
4.5	Perancangan Antarmuka Lihat Jadwal Dosen	48
5.1	Tampilan eror saat membuka file bertipe .xlsx	54
5.2	Keterangan Error File .xlsx	54
5.3	Error Sudah Tidak Muncul Ketika File Dibuka	56
5.4	Tombol <i>Delete All</i> di Entri Jadwal Dosen	58
5.5	Tampilan Konfirmasi	58
5.6	Semua Jadwal Dihapus	59
5.7	Informasi Waktu <i>Update</i> Terakhir Jadwal di Entri Jadwal Dosen	60
5.8	Informasi Waktu <i>Update</i> Terakhir Jadwal di Lihat Jadwal Dosen	60
5.9	Memasukan Jadwal ke Waktu yang Sudah Ada Jadwal	62
5.10	Tampilan Error Gagal Masuk	62
5.11	Jadwal yang Sudah Ada	63
5.12	Mengubah Jadwal "1"	63
5.13	Tampilan Error Gagal <i>Update</i>	64
5.14	Mengubah Jadwal "2"	64
5.15	Tampilan Error Gagal <i>Update</i>	65

DAFTAR TABEL

2.1	Daftar Preferensi di Kelas Migrasi	11
2.2	Daftar Variabel Sass Untuk <i>Reveal Modal</i>	15
2.3	Daftar Variabel Sass Untuk Kelas Tabel	17
4.1	Perancangan Tabel <code>jadwal_dosen</code>	40
4.2	Perincian method <code>insert</code>	41
4.3	Perincian method <code>edit</code>	41
4.4	Perincian method <code>delete</code>	42
4.5	Perincian method <code>index</code>	42
4.6	Perincian method <code>ekspor</code>	43
4.7	Perincian method <code>add_jadwal</code>	43
4.8	Perincian method <code>update_jadwal</code>	44
4.9	Perincian method <code>delete_jadwal</code>	44
4.10	Perincian method <code>get_jadwal</code>	44
5.1	Pengujian Login	50
5.2	Pengujian Fungsional Entri Jadwal Dosen	51
5.3	Pengujian Fungsional Edit Jadwal Dosen	51
5.4	Pengujian Fungsional Hapus Jadwal Dosen	52
5.5	Pengujian Fungsional Lihat Jadwal Dosen	52
5.6	Pengujian Fungsional Ekspor ke XLS	53
5.7	Perancangan method <code>deleteAll</code>	57
5.8	Perancangan method <code>deleteByUsername</code>	57
5.9	Perancangan field tambahan di <code>jadwal_dosen</code>	59

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Aplikasi *Blue Tape* adalah aplikasi sederhana yang memiliki tujuan utama untuk mengubah berbagai pekerjaan *paper-based* di FTIS UNPAR menjadi *paperless*. Selain itu aplikasi ini memiliki beberapa kegunaan lainnya seperti mengautentikasi mahasiswa dan staf UNPAR via OAuth 2.0 ke Google (layanan OAuth ke Google ini juga dapat digunakan untuk menentukan hak akses yang bisa dilihat dari email pengguna) dan *Pilot Project* untuk permohonan transkrip ke Tata Usaha . Aplikasi ini merupakan aplikasi berbasis web dengan memanfaatkan *Codeigniter* dan *Zurb Foundation*. Selain itu aplikasi *Blue Tape* ini didesain sebagai *framework* agar dapat ditambahkan layanan-layanan baru. Untuk menambahkan layanan baru sudah tersedia menu khusus, developer cukup menambahkan layanan baru dalam bentuk modul. Untuk saat ini *Blue Tape* baru memiliki layanan untuk *Transcript Request / Manage* yang memiliki fungsi untuk melakukan permohonan serta pencetakan transkrip mahasiswa.

Pada saat ini untuk menginformasikan jadwalnya masing-masing, dosen harus mencetak *hardcopy*-nya dengan template seperti pada Gambar 1.1. di bawah ini.

JADWAL AKTIVITAS DOSEN					
Dosen:					
	Senin	Selasa	Rabu	Kamis	Jumat
7 - 8					
8 - 9					
9 - 10					
10 - 11					
11 - 12					
12 - 13					
13 - 14					
14 - 15					
15 - 16					
16 - 17					
Keterangan:					
		Waktu Konsultasi			
		Jika Dijadwalkan			

Gambar 1.1: Template jadwal dosen

Jadwal tersebut akan ditempel pada dinding ruangan masing-masing dosen. Sedangkan bila menggunakan *Blue Tape* maka dosen tidak perlu lagi mencetak jadwalnya tersebut karena mahasiswa dapat melihat jadwal setiap dosen di dalam aplikasi ini. Maka dari itu aplikasi ini membuat

1 pencatatan jadwal dosen menjadi *papeless*.

2 Pada Skripsi ini akan ditambahkan dua modul yaitu modul entri jadwal untuk dosen informatika
3 dan modul lihat jadwal dosen untuk mahasiswa ke dalam aplikasi Blue Tape. Modul-modul tersebut
4 berfungsi untuk melakukan hal-hal yang berhubungan dengan pembangkitan jadwal dosen. Modul
5 dosen memiliki beberapa fungsi diantaranya: input jadwal mingguan dosen(jadwal dapat berupa
6 jadwal konsultasi, jadwal konsultasi tentatif ataupun jadwal rutin), mencatat *update* terakhir jadwal
7 dosen dan mengekspor jadwal dosen ke XLS. Modul Umum sendiri memiliki fungsi untuk melihat
8 jadwal seluruh dosen dan mengekspor jadwal dosen ke XLS.

9 1.2 Rumusan Masalah

10 Rumusan masalah yang akan dibahas dalam penelitian ini :

- 11 • Bagaimana mengintegrasikan autentikasi mahasiswa maupun staf UNPAR yang mengakses
12 *Blue Tape*?
- 13 • Bagaimana cara mencatat, *update* dan melihat jadwal dosen di *Blue Tape*?
- 14 • Bagaimana mengekspor jadwal dosen ke XLS sesuai template yang saat ini berlaku?

15 1.3 Tujuan

16 Tujuan yang ingin dicapai dalam penelitian ini :

- 17 • Mengimplementasikan autentikasi pengguna yang mengakses *Blue Tape*
- 18 • Membuat modul entri jadwal dosen dan modul lihat jadwal dosen yang berfungsi untuk
19 menginput jadwal mingguan, *update* dan melihat jadwal dosen
- 20 • Mengimplementasikan kode-kode yang diperlukan untuk memasukkan data-data yang ada di
21 dalam PHP ke dalam *file* XLS.

22 1.4 Batasan Masalah

23 Dalam penelitian ini ditetapkan batasan-batasan masalah sebagai berikut:

24 -tidak memeriksa hasil akhir XLS atau tidak -user experience tidak dibuat serumit atau selengkap
25 Google Calendar

- 26 • Aplikasi ini tidak memeriksa hasil akhir bertipe file .xls, .xlsx atau tipe file lainnya.
- 27 • Aplikasi ini tidak memiliki fitur serumit atau selengkap Google Calendar. Aplikasi ini hanya
28 didesign khusus untuk memasukan, menampilkan dan mengeskor jadwal dosen.

1.5 Metodologi

Metode penelitian yang akan digunakan dalam skripsi ini adalah:

1. Studi literatur mengenai:

- bahasa pemrograman PHP
- *framework* Codeigniter
- modul *Zurb Foundation*, PHPExcel dan *regular expression*
- Prosedur pembangkitan jadwal dosen

2. Analisis kebutuhan aplikasi dengan mengenali metode pencatatan jadwal dosen saat ini dan mengimplementasikannya ke dalam modul tersebut

3. Membangun modul aplikasi yang sesuai dengan kebutuhan dosen dan mahasiswa dalam pembangkitan jadwal dosen agar aplikasi yang dibuat dapat membantu kedua pihak dalam mengakses informasi-informasi yang berkaitan dengan jadwal dosen . Pembuatan modul aplikasi ini dibagi menjadi empat tahap :

- Analisis kebutuhan modul
- Perancangan modul
- Implementasi
- Pengujian modul

1.6 Sistematika Pembahasan

Untuk penulisan skripsi ini akan dibagi dalam enam bagian sebagai berikut :

Bab 1 Pendahuluan berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian dan sistematika penulisan.

Bab 2 Landasan Teori berisi dasar-dasar teori yang akan digunakan dalam pembuatan aplikasi pembangkit jadwal dosen. Dasar-dasar teori yang akan digunakan diantaranya adalah bahasa pemrograman PHP, framework Codeigniter, Google OAuth, CSRF, Zurb Foundation dan PHPExcel.

Bab 3 Analisis berisi analisis kebutuhan data, analisis sistem yang sudah ada sekarang dan analisis sistem usulan.

Bab 4 Perancangan berisi perancangan basis data, perancangan kelas-kelas program dan perancangan antarmuka program.

Bab 5 Implementasi dan Pengujian menjelaskan mengenai lingkungan yang digunakan dalam proses implementasi dan pengujian Aplikasi Pembangkit Jadwal Dosen. Bab ini juga mencatat hasil pengujian fungsional dan hasil pengujian eksperimental program.

Bab 6 Kesimpulan dan Saran berisi kesimpulan dari hasil pembuatan Aplikasi Pembangkit Jadwal Dosen ini dan kegunaannya, serta memberikan saran-saran untuk kelanjutan pengembangan aplikasi ini.

BAB 2

LANDASAN TEORI

2.1 BlueTape

Aplikasi *Blue Tape* adalah perangkat lunak *open source* sederhana yang memiliki tujuan utama untuk mengubah berbagai pekerjaan *paper-based* di FTIS UNPAR menjadi *paperless*. Selain itu perangkat lunak ini memiliki beberapa kegunaan lainnya seperti mengautentikasi mahasiswa dan staf UNPAR via OAuth 2.0 ke Google (layanan OAuth ke Google ini juga dapat digunakan untuk menentukan hak akses yang bisa dilihat dari email pengguna) dan *Pilot Project* untuk permohonan transkrip ke Tata Usaha . Aplikasi ini merupakan aplikasi berbasis web dengan memanfaatkan *Codeigniter* dan *Zurb Foundation*.

Perangkat lunak *Blue Tape* ini didesain sebagai *framework* yang terdiri dari beberapa layanan yang dipisahkan ke dalam modul-modul. Pemisahan layanan ke dalam modul-modul dibuat dengan tujuan agar pemeliharaan perangkat lunak lebih mudah dan juga mempermudah cara untuk menambahkan layanan baru ke dalam BlueTape. Sudah ada layanan yang aktif di BlueTape saat ini yaitu *Transcript Request / Manage* yang memiliki fungsi untuk melakukan permohonan serta pencetakan transkrip mahasiswa.

2.2 CodeIgniter

CodeIgniter adalah *framework* pengembangan aplikasi untuk *developer* yang membangun situs web menggunakan PHP. Tujuannya adalah untuk memungkinkan Anda mengembangkan proyek lebih cepat, daripada bila *developer* menulis kode dari awal, dengan menyediakan banyak kumpulan *library* untuk tugas-tugas yang sering dibutuhkan dan juga menyediakan tampilan sederhana serta struktur logika untuk mengakses *library-library* tersebut. CodeIgniter memungkinkan *developer* untuk fokus secara kreatif pada proyek *developer* dengan cara meminimalkan jumlah kode yang dibutuhkan untuk setiap tugas yang diberikan. [1]

CodeIgniter dirancang untuk memenuhi kebutuhan :

- *Framework* dengan tapak keberadaan yang kecil
- performa yang baik
- kompatibilitas akun *hosting* yang luas yang dapat berjalan di berbagai versi dan konfigurasi PHP

- *Framework* yang hampir tidak membutuhkan konfigurasi
- *Framework* yang tidak membutuhkan *command line*
- *Framework* yang tidak mengikuti aturan pengkodean yang ketat
- membutuhkan solusi yang sederhana
- dokumentasi yang menyeluruh

2.2.1 *System Requirements* Server Untuk Menjalankan CodeIgniter

Server disarankan sudah menggunakan PHP versi 5.6 atau versi-versi setelahnya. CodeIgniter dapat berjalan pada PHP versi lama, namun ada kemungkinan muncul masalah-masalah yang berkaitan dengan performa dan keamanan. [1]

Database yang didukung oleh CodeIgniter adalah sebagai berikut :

- MySQL (5.1+) melalui *mysql*, *mysqli* dan *pdo* drivers
- Oracle melalui *oci8* dan *pdo* drivers
- PostgreSQL melalui *postgre* dan *pdo* drivers
- MS SQL melalui *mssql*, *sqlsrv* (versi 2005 dan setelahnya) dan *pdo* drivers
- SQLite melalui *sqlite* (versi 2), *sqlite3* (versi 3) dan *pdo* drivers
- CUBRID melalui *cubrid* dan *pdo* drivers
- Interbase/Firebird melalui *ibase* dan *pdo* drivers
- ODBC melalui *odbc* dan *pdo* drivers

2.2.2 Cara Instalasi CodeIgniter

Instalasi CodeIgniter dilakukan dalam 4 langkah :

- *Unzip package* CodeIgniter tersebut [1]
- Unggah folder CodeIgniter dan data-datanya ke dalam server. Pada umumnya *index.php* berada pada *root* server.[1]
- Buka *application/config/config.php* menggunakan program pengolah teks dan tentukan URL-nya. Jika akan digunakan enkripsi, tentukan kunci enkripsinya.[1]
- Jika *database* akan digunakan, buka *application/config/database.php* menggunakan program pengolah teks dan atur *database* anda.[1]

2.2.3 Cross-site Request Forgery (CSRF)

Perlindungan terhadap CSRF dapat dilakukan dengan cara mengubah nilai variabel berikut menjadi TRUE pada file **application/config/config.php** [1]

```
$config['csrf_protection'] = TRUE;
```

Jika *form helper* digunakan di view aplikasi, maka field csrf akan secara otomatis dimasukkan ke dalam form tersebut. Kode dibawah ini akan secara otomatis dimasukkan oleh CodeIgniter ketika `form_open()` dipanggil.

```
<input type="hidden" name="<?=$csrf['name'];?>" value="<?=$csrf['hash'
↪ '];?>" />
```

Token dapat diatur agar selalu meregenerasi setiap terjadi *submission* atau juga dapat diatur agar selalu tetap sama sepanjang masa hidup *cookie* CSRF-nya. Regenerasi token akan membuat keamanan sistem lebih kuat, namun dapat menyebabkan masalah navigasi web (seperti navigasi *back/forward* halaman ke halaman, membuka tab baru, dan lain-lain). Regenerasi token dapat diatur dengan mengubah nilai variabel : [1]

```
$config['csrf_regenerate'] = TRUE;
```

2.2.4 Model-View-Controller

CodeIgniter didasari pola pengembangan *Model-View-Controller* atau MVC. MVC memisahkan logika aplikasi dengan tampilannya. [1]

- **Model** merepresentasikan struktur data. Pada umumnya kelas-kelas model menampung fungsi-fungsi untuk mengambil, memperbarui atau memasukan data ke dalam basis data.
- **View** menampilkan informasi ke pengguna.
- **Controller** berfungsi sebagai perantara antara model dan view.

2.2.5 Controller

Sebuah *controller* adalah kelas yang dinamakan demikian agar dapat diasosiasikan dengan URI. sebagai contoh URI "example.com/index.php/blog/" , CodeIgniter akan mencari *controller* bernama Blog.php dan menjalankannya. Nama *controller* harus diawali dengan huruf kapital. Selain itu *controller* juga harus *extend* kelas "CI_Controller". [1]

Contoh yang benar :

```
<?php
class Blog extends CI_Controller {

}
}
```

Contoh yang salah :

```
<?php
class blog extends CI_Controller {

}
```

2.2.5.1 Method

Untuk menjalankan suatu method, maka developer perlu menuliskannya pada segmen kedua URI. Contoh "example.com/index.php/blog/comments" maka akan dijalankan method comments() pada controller blog.php. Method yang akan dijalankan bila bagian kedua URI kosong adalah method index(). Jika URI mengandung lebih dari dua segment, segment-segment tersebut akan dimasukkan ke dalam method sebagai parameter. [1]

2.2.5.2 Default Controller

CodeIgniter dapat diperintahkan untuk menjalankan *default controller* jika tidak terdapat URI, pada umumnya terjadi ketika hanya terdapat permintaan menggunakan URL dasar *website*. Penentuan *default controller* terdapat pada file "application/config/routes.php" dan set variabel . Nama *controller* tersebut adalah 'blog', maka ketika index.php dijalankan tanpa menspesifikasikan URI akan dijalankan *controller* 'blog'. [1]

2.2.5.3 Memproses Output

CodeIgniter memiliki kelas *output* yang mengurus pengiriman data ke *web browser* secara otomatis. Untuk kasus-kasus saat pengguna ingin mengubah cara pengiriman data tersebut, CodeIgniter menyediakan caranya dengan menambahkan method bernama "__output()" ke *controller* terkait. Jika *controller* memiliki method bernama "__output()" maka *controller* tersebut akan selalu dipanggil oleh kelas "output". [1]

Contoh penggunaan method "__output()" :

```
public function _output(\&$output)
{
    echo $output;
}
```

2.2.5.4 Private Method

Method-method dengan tipe *private* tidak dapat diakses oleh publik. Method ini hanya dapat diakses oleh method lain dalam *controller* yang sama. Selain itu method ini juga tidak akan dapat diakses melalui URL. [1]

Contoh penulisan *private method*:

```
private function _utility()
{
    // kode program
}
```

```
}

```

Method di atas tidak dapat diakses dengan cara pemanggilan method pada umumnya seperti :

```
example.com/index.php/blog/_utility/

```

2.2.5.5 Mengorganisir Controller-controller ke Dalam Sub Direktori

Di CodeIgniter pengguna dapat mengorganisir *controller-controller* ke dalam sub direktori. Untuk melakukannya cukup dengan membuat sub direktori di dalam direktori *application/controllers/* dan simpan kelas-kelas *controller* ke dalamnya. Ketika menggunakan fitur ini, pengguna harus menspesifikasikan foldernya ke dalam URI.^[1]

Contoh ada controller yang berlokasi di:

```
application/controllers/products/Shoes.php

```

Untuk memanggil controller tersebut URI harus menuliskan nama sub direktorinya :

```
example.com/index.php/products/shoes/show/123

```

2.2.6 View

View adalah sebuah halaman web, bagian-bagian halaman (seperti *header*, *footer*, *sidebar*, dan-lain-lain) atau bagian dari *view* lainnya. *View* tidak pernah dipanggil secara langsung, melainkan harus melalui controller karena dalam framework MVC controller berfungsi sebagai pengatur.^[1]

Controller dapat memanggil view menggunakan potongan kode seperti di bawah ini :

```
$this->load->view('name');
```

Jika 1 halaman terdiri dari beberapa view, CodeIgniter juga menyediakan cara untuk memanggil banyak view sekaligus:

```
class Page

function INDEX()
    $data['page_title'] = 'Your title'
    load->view('header')
    load->view('menu')
    load->view('content', $data)
    load->view('footer')
```

Bila ada data lain yang akan dikirimkan ke view dari controller, gunakan potongan kode seperti di bawah ini :

```
$this->load->view('name', $data);
```

2.2.7 Model

Model adalah kelas PHP yang dirancang untuk bekerja dengan informasi-informasi di *database*. Sebagai contoh, misalkan CodeIgniter digunakan untuk mengatur sebuah blog maka model mengandung fungsi-fungsi seperti *select*, *insert*, *update* data-data blog. [1]

2.2.7.1 Anatomi dari Model

Model disimpan di direktori **application/models**. Model dapat disimpan dalam sub-direktori jika dibutuhkan. Bentuk dasar kode pada kelas model adalah seperti di bawah ini: [1]

```
class Model_name extends CI_Model {  
  
    public function __construct()  
    {  
        parent::__construct();  
        //constructor code  
    }  
}
```

Model_name adalah nama dari kelasnya. Huruf pertama dari nama kelas harus huruf kapital dengan huruf-huruf setelahnya ditulis menggunakan huruf kecil. Kelas model harus dipastikan meng-*extend* kelas Model dasar. Nama file harus sama dengan nama kelasnya. [1]

2.2.7.2 Load Model

Pada umumnya model akan di-*load* di dalam method-method *controller*. Untuk *load* model diperlukan method berikut:

```
$this->model->model('model_name');
```

Jika model terdapat pada sub-direktori, perlu dimasukan *path*-nya ke dalam direktori 'models'. Misalnya model terdapat di *application/models/blog/Queries.php* maka untuk *load* model gunakan :

```
$this->load->model('blog/queries');
```

Bila ada sebuah model yang akan digunakan secara global pada aplikasi pengguna, CodeIgniter bisa diperintahkan untuk *auto-load* model tersebut ketika instalasi sistem. Untuk melakukan hal ini, tambahkan model ke dalam *autoload array* yang berlokasi di "*application/config/autoload.php*". [1]

2.2.7.3 Menghubungkan Model Dengan Database

Ketika Model dipanggil, model tidak akan terhubung ke database secara otomatis. Ada dua cara untuk menghubungkan model dengan *database* : [1]

1. Menghubungkan secara otomatis ketika instalasi sistem dilakukan menggunakan fitur auto-load seperti disebutkan di bagian *Load Model* di atas.

2. Menghubungkan *database* secara manual. Untuk menghubungkan *database* dengan model, pengguna perlu untuk memanggil *database*-nya di *controller* atau di *model* itu sendiri dengan menuliskan potongan kode: [1]

```
$this->load->database();
```

Parameter-parameter yang tersedia untuk potongan kode di atas adalah:

- Nama dari database yang dituju berupa array atau *Data Source Name*(DSN) string.
- *Boolean* untuk menentukan id koneksi perlu dikembalikan ke model atau tidak.
- *Boolean* untuk menentukan perlunya dibuat *Query Builder* atau tidak. Jika tidak ditentukan, nilai dari parameter ini adalah TRUE.

2.2.8 Kelas Migrasi

Kelas ini digunakan untuk memudahkan pengubahan *database* secara terstruktur dan terorganisir dengan baik. Dengan cara ini maka pengguna tidak perlu memberi tahu pengguna lainnya yang sedang mengembangkan aplikasi yang sama untuk melakukan perubahan *database* secara manual, pengguna lain cukup menjalankan kelas migrasi ini. Kelas ini akan secara otomatis membuat tabel bernama "*migration*" di *database* yang mencatat migrasi-migrasi mana saja yang sudah dijalankan, yang perlu dilakukan secara manual oleh pengguna adalah untuk memperbarui *file* aplikasinya dan memanggil *\$this->migration->current()* untuk menentukan migrasi mana yang perlu dijalankan. Untuk mengatur preferensi dari kelas migrasi terdapat di **application/config/migration.php**. [1]

Preferensi	Default	Opsi	Deskripsi
migration_enabled	FALSE	TRUE / FALSE	Menentukan migrasi dijalankan atau tidak.
migration_path	APPPATH.migrations/'	None	Alamat ke folder migrasi.
migration_version	0	None	Versi migrasi yang digunakan.
migration_table	migrations	None	Nama tabel migrasi di <i>database</i> .
migration_auto_latest	FALSE	TRUE / FALSE	Menentukan dijalanannya migrasi otomatis atau tidak.
migration_type	'timestamp'	'timestamp' / 'sequential'	Tipe penomoran kelas migrasi yang digunakan.

Tabel 2.1: Daftar Preferensi di Kelas Migrasi

2.2.8.1 Nama Kelas Migrasi

Setiap migrasi dijalankan sesuai urutan numerik baik dengan urutan menaik atau menurun tergantung dari metode yang digunakan. Ada dua cara penomoran yang bisa dilakukan yaitu :

- **Sekuensial:** setiap migrasi dinomori secara sekuensial, dimulai dengan angka **001**. Setiap nomor harus terdiri dari 3 digit dan tidak ada jeda di angka tersebut. Cara penomoran ini adalah satu-satunya cara yang digunakan sebelum versi CodeIgniter 3.0 . [1]
- **Timestamp:** setiap migrasi menggunakan *timestamp* ketika kelas migrasinya dibuat. Format penomrannya adalah YYYYMMDDHHIISS.
 - YYYY : tahun terdiri dari 4 digit
 - MM : bulan terdiri dari 2 digit
 - DD : hari terdiri dari 2 digit
 - HH : jam terdiri dari 2 digit
 - II : menit terdiri dari 2 digit
 - SS : detik terdiri dari 2 digit

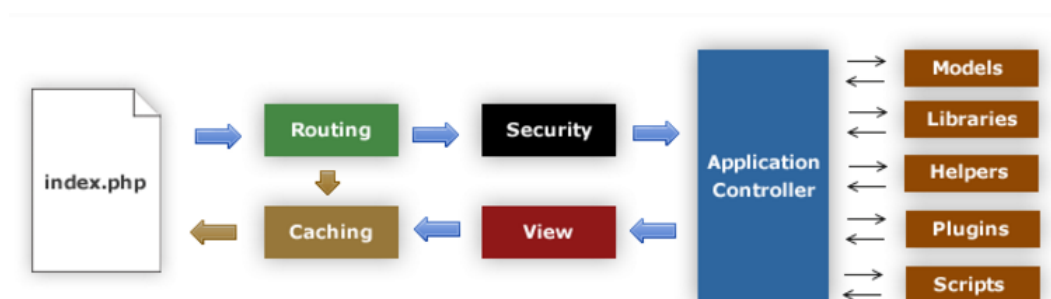
Cara ini digunakan untuk menghindari konflik penomoran ketika bekerja dalam tim. Cara penomoran ini merupakan cara yang lebih banyak digunakan sejak versi CodeIgniter 3.0 dan setelahnya. [1]

Urutan penomran dapat diatur menggunakan `$config['migration_type']` di file `application/config/migration.php` . Apapun cara yang dipilih, pisahkan nama file dan penomoran kelasnya menggunakan tanda garis bawah. Sebagai contoh: [1]

- 001_add_blog.php (penomoran sekuensial)
- 20121031100537_add_blog.php (penomoran dengan *timestamp*)

2.2.9 Flow Chart Codeignter

Gambar 2.1 mengilustrasikan aliran data dalam sistem :



Gambar 2.1: Flow Chart Codeignter

1. `index.php` berfungsi sebagai controller depan. Menginisialisasi sumber daya yang dibutuhkan untuk menjalankan CodeIgniter

2. Router memeriksa permintaan HTTP untuk menentukan apa yang akan dilakukan pada permintaan tersebut.
3. Jika ada cache file, maka akan dikirim langsung ke browser. Melewati cara eksekusi sistem yang normal.
4. Security. Sebelum controller aplikasi dimuat, permintaan HTTP dan data-data pengguna yang telah diserahkan disaring untuk keamanan.
5. Controller memuat model, pustaka inti (*core libraries*), pembantu dan sumber daya lain yang dibutuhkan untuk memproses permintaan khusus.
6. Kemudian tampilan akhir dibuat dan dikirim ke web browser untuk dilihat. Jika caching diaktifkan, maka tampilan dimasukkan ke dalam cache terlebih dahulu sehingga pada permintaan selanjutnya tampilan tersebut dapat diakses lebih cepat.

2.3 Cross Site Request Forgery

"Cross Site Request Forgery" atau "Session Riding" adalah teknik penyerangan yang mengeksploitasi otentikasi implisit. Penyerangan ini dilakukan dengan cara menyebabkan browser korban membuat *http requests* tersembunyi ke sumber daya - sumber daya terlarang. Pada kasus ketika *request* untuk sumber daya tersebut berhasil, akan menyebabkan browser atau aplikasi web terkait untuk melakukan tindakan-tindakan penyerangan tersebut secara terus menerus. Penyerangan ini bertujuan untuk berbagai hal seperti mengubah *field-field* di dalam *database*, mengirim email atau mengubah bagian-bagian aplikasi. Semua aksi tersebut dilakukan menggunakan token otentikasi milik korban. [2]

Pada umumnya *web browser* memiliki kebijakan untuk memperbolehkan *website-website* untuk mengirim *HTTP request* ke alamat jaringan manapun. Akibat dari kebijakan ini, maka penyerang dapat mengendalikan konten-konten yang dianggap oleh *browser* bukan di bawah kendali pengguna. Konten yang dapat dikendalikan oleh penyerang diantaranya : [2]

- **Konektivitas Jaringan.** Misalnya bila korban menggunakan *firewall*, maka penyerang dapat mempengaruhi browser pada mesin korban untuk mengirim *network request* ke mesin-mesin lain yang menggunakan firewall juga yang tidak dapat diakses secara langsung melalui mesin milik penyerang.
- **Membaca Status Browser.** *Request-request* yang dikirimkan ke jaringan melalui *browser* pada umumnya berisi status browser seperti cookies, sertifikat klien atau data otentikasi sederhana.
- **Menulis Status Browser.** Penyerang membuat *browser* untuk menulis *network request*. *Browser* juga akan bereaksi pada respon dari website yang dituju. Hal ini mengakibatkan *browser* memodifikasi beberapa bagian status browser.

Serangan CSRF tidak terbatas pada satu *request* palsu saja. Alur kerja yang membutuhkan serangkaian *request* http juga rentan terhadap serangan ini selama kondisi-kondisi yang diperlukan terpenuhi. Misalnya bila setiap konten dan *identifier* dalam setiap langkah pada alur kerja *web form* diketahui dan alur kerja dari *website* tersebut tidak memiliki mekanisme lain untuk melacak cara kerja langkah per-langkah *web form*-nya, melainkan hanya menggunakan *identifier* dari *session* saja. Jika kondisi tersebut dipenuhi, maka penyerang dapat membuat serangkaian *iframe-iframe* tersembunyi yang mengandung *web form* berbahaya. *Form-form* tersebut lalu dikirimkan secara sekuensial melalui JavaScript menggunakan *event onload* dari *iframe* untuk membuat server yang menerimanya mengira bahwa *form-form* tersebut diisi oleh user dengan benar. [3]

2.4 Zurb Foundation

Foundation adalah kumpulan pola desain HTML, CSS dan Javascript yang dapat digunakan untuk membuat website. Hal tersebut untuk membantu *developer* agar tidak perlu menulis kode yang sama berulang kali. Selain membantu menghemat waktu, Foundation juga membantu *developer* untuk menulis kode dengan lebih baik. Foundation dapat bekerja pada berbagai media seperti komputer desktop, laptop, tablet, dan telepon genggam.[4]

Komponen-komponen dalam Foundation sendiri ada beberapa macam diantaranya sebagai berikut:

- Grid untuk mempermudah pembagian halaman.
- Desain tombol yang bermacam-macam. Desain tombol ini dapat diubah-ubah dengan cara menambahkan kelas.
- Navigasi untuk mempermudah pengunjung aplikasi dalam menggunakan aplikasinya.
- Plugins JavaScript untuk mempermudah *developer* dalam membuat tampilan aplikasinya.

2.4.1 Top Bar

Top bar adalah *wrapper* sederhana untuk komponen-komponen menu website. Top bar dapat memiliki 2 bagian yaitu bagian kiri (**.top-bar-left**) dan bagian kanan (**.top-bar-right**). Pada layar yang kecil bagian top bar ini bisa menjadi di atas atau di bawah sisi lainnya. [4]

2.4.2 Reveal Modal

Pada dasarnya modal hanyalah sebuah kontainer kosong, jadi pengguna dapat memasukan konten apapun ke dalamnya. Untuk membuat modal, buat kelas **.reveal** dan atribut **data-reveal** lalu beri id unik ke kontainernya tersebut seperti contoh di bawah ini : [4]

```
<div class="reveal" id="exampleModal1" data-reveal>
    ...
</div>
```

Untuk membuka modal, tambahkan atribut **data-open** ke elemen apapun. Nilai dari **data-open** adalah id dari modalnya.

```

1 <p><button class="button" data-open="exampleModal1">Click me for a modal</button
2
3   ↪ ></p>
4

```

Jika tidak diberi aturan tambahan, pada dasarnya modal akan ditutup jika pengguna menekan area di luar dari modal atau ketika tombol ESC ditekan. Untuk memberi tombol "close" di menu modal tambahkan atribut **data-close** ke elemen yang berisi pemicu modal. Contoh dapat dilihat pada potongan kode di bawah. [\[4\]](#)

```

9
10 <button class="close-button" data-close aria-label="Close modal" type="button">
11   <span aria-hidden="true">&times;</span>
12 </button>
13

```

2.4.3 Variabel Sass *Reveal Modal*

Berikut adalah variabel-variabel yang dapat digunakan untuk mengkostumisasi modal :

Nama	Tipe	Nilai <i>Default</i>	Deskripsi
\$reveal-background	Color	\$white	Warna <i>default</i> latar belakang modal
\$reveal-width	Number	600px	Lebar <i>default</i> modal, tanpa memakai kelas apapun
\$reveal-max-width	Number	\$global-width	Lebar maximum modal
\$reveal-padding	Number	\$global-padding	<i>Default</i> padding di dalam modal
\$reveal-border	Number	\$global-radius	Nilai <i>default</i> radius untuk sebuah modal
\$reveal-zindex	Number	1005	nilai z-index untuk modal.
\$reveal-overlay-background	Color	rgba(\$black,0,45)	Warna latar belakang penutup modal

Tabel 2.2: Daftar Variabel Sass Untuk *Reveal Modal*

2.4.4 Scrolling Table

Scrolling table digunakan bila banyak data yang ada pada tabel. Dengan menggunakan ini maka isi tabel dapat digeser secara horizontal. Untuk menggunakan tabel jenis ini, deklarasikan kelas **table-scroll** seperti contoh di bawah:

```

21
22 <div class="table-scroll">
23   <table></table>

```

</div>

2.4.4.1 Variabel Sass Untuk Tabel

Berikut adalah daftar variabel Sass yang dapat dikustomisasi untuk mengubah tampilan komponen-komponen dari tabel

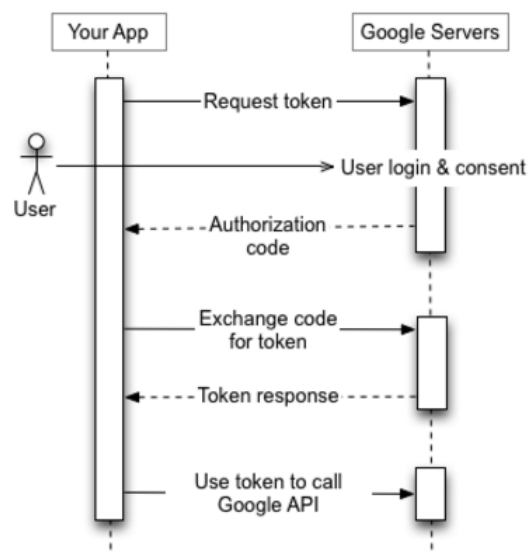
Nama	Tipe	Nilai <i>Default</i>	Deskripsi
<code>\$table-background</code>	Color	<code>\$white</code>	Warna <i>default</i> tabel
<code>\$table-color-scale</code>	Number	5%	Skala warna gelap pada tabel bergaris
<code>\$table-border</code>	List	1px solid smart-scale(<code>\$table-background</code> , <code>\$table-color-scale</code>)	Tipe batas tabel yang digunakan
<code>\$table-padding</code>	Number	rem-calc(8 10 10)	<i>Default</i> padding di tabel
<code>\$table-hover-scale</code>	Number	2%	Skala warna gelap baris tabel ketika ditunjuk kursor
<code>\$table-row-hover</code>	List	darken(<code>\$table-background</code> , <code>\$table-hover-scale</code>)	Warna baris pada tabel ketika ditunjuk oleh kursor.
<code>\$table-row-stripe-hover</code>	List	darken(<code>\$table-background</code> , <code>\$table-color-scale</code> + <code>\$table-hover-scale</code>)	Warna baris gelap pada tabel bergaris ketika ditunjuk kursor
<code>\$table-is-striped</code>	Boolean	TRUE	Jika bernilai TRUE maka tabel menggunakan tipe tabel bergaris
<code>\$table-striped-background</code>	Color	smart-scale(<code>\$table-background</code> , <code>\$table-color-scale</code>)	Warna baris gelap pada tabel bergaris
<code>\$table-stripe</code>	Keyword	even	Nilai untuk memunculkan baris gelap pada tabel, kecuali pada bagian <i>header</i> . Jika bernilai <i>even</i> , maka baris genap akan memiliki warna latar. Jika bernilai <i>odd</i> maka baris ganjil yang akan memiliki warna latar
<code>\$table-head-background</code>	Color	smart-scale(<code>\$table-background</code> , <code>\$table-color-scale</code> / 2)	Warna latar belakang <i>header</i> tabel
<code>\$table-head-row-hover</code>	List	darken(<code>\$table-head-background</code> , <code>\$table-hover-scale</code>)	Warna <i>header</i> ketika ditunjuk kursor
<code>\$table-foot-background</code>	Color	smart-scale(<code>\$table-background</code> , <code>\$table-color-scale</code>)	Warna latar belakang <i>footer</i>
<code>\$table-foot-row-hover</code>	List	darken(<code>\$table-foot-background</code> , <code>\$table-hover-scale</code>)	Warna latar belakang <i>footer</i> ketika ditunjuk kursor
<code>\$table-head-font-color</code>	Color	<code>\$body-font-color</code>	Warna teks yang berada di <i>header</i>
<code>\$table-foot-font-color</code>	Color	<code>\$body-font-color</code>	Warna teks yang berada di <i>footer</i>
<code>\$show-header-for-stacked</code>	Boolean	false	Nilai untuk menentukan penggunaan header ketika menggunakan <i>stacked table</i>
<code>\$table-stack-breakpoint</code>	Breakpoint	medium	Batas pemicu perubahan jenis tabel dari tipe <i>mobile</i> menjadi <i>desktop</i> atau sebaliknya

Tabel 2.3: Daftar Variabel Sass Untuk Kelas Tabel

2.5 Google OAuth 2.0

Google OAuth 2.0 merupakan salah satu protokol dari Google Sign-in. Google OAuth 2.0 digunakan oleh Google API untuk otorisasi dan autentikasi. Secara garis besar, cara pemakaian Google OAuth 2.0 adalah sebagai berikut :

1. Dapatkan OAuth 2.0 credential dari konsol Google API. OAuth 2.0 credential seperti client ID dan client secret yang diketahui oleh Google dan aplikasi pengguna, dapat didapatkan di halaman <https://console.developers.google.com/>.
2. Dapatkan token akses dari Google Authorization Server. Sebelum aplikasi dapat mengakses data pribadi menggunakan Google API, aplikasi tersebut harus mendapat token akses yang memberikan akses ke API. Satu token akses dapat memberikan berbagai macam akses ke banyak API. Variable parameter "*scope*" mengendalikan kumpulan-kumpulan sumber daya dan operasi yang telah diperbolehkan untuk diakses oleh token akses. Selama masa permintaan token akses, aplikasi mengirimkan satu atau lebih nilai ke dalam parameter "*scope*". Ada beberapa cara untuk melakukan permintaan, tergantung dari tipe aplikasi yang sedang dibuat. Sebagai contoh aplikasi JavaScript dapat meminta token akses menggunakan *redirect* dari *browser* yang mengarah ke Google, sementara aplikasi lain yang terinstall di dalam perangkat yang tidak memiliki browser menggunakan *web service* untuk melakukan permintaan. Beberapa permintaan membutuhkan tahap autentikasi yang meminta pengguna untuk masuk ke akun Google mereka. Setelah masuk ke dalam akun, pengguna akan diminta jika mereka bersedia untuk memberikan izin ke aplikasi yang sedang melakukan permintaan tersebut.



Gambar 2.2: Skenario OAuth menggunakan token akses

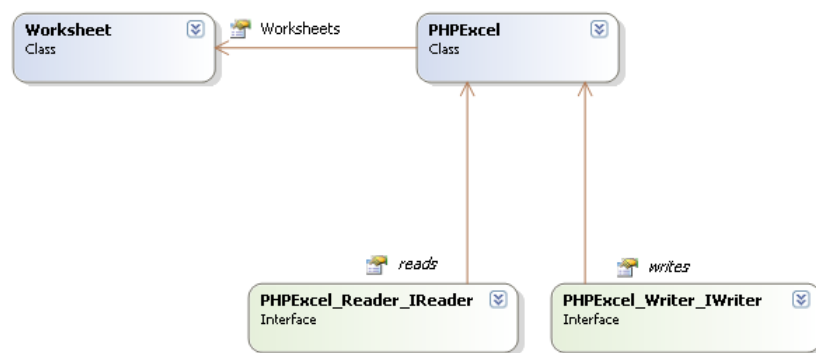
3. Mengirim akses token ke API. Setelah aplikasi mendapatkan akses token, aplikasi tersebut akan mengirim token ke Google API dalam bentuk HTTP *authorization header*. Jika memungkinkan,

aplikasi dapat mengirim token-token sebagai parameter *URI query-string*. Pengiriman dalam bentuk parameter URI tidak disarankan karena parameter URI dapat tersimpan dalam *log* yang tidak aman. Token akses hanya berlaku untuk kumpulan operasi dan sumber daya yang dideskripsikan dalam parameter *scope* di permintaan token.

4. Jika dibutuhkan, token akses dapat di-*refresh* karena token akses memiliki masa berlaku terbatas. Jika aplikasi membutuhkan akses ke Google API lebih dari masa berlaku satu buah token, aplikasi dapat mendapatkan token *refresh*. Token *refresh* memungkinkan aplikasi untuk mendapatkan token akses baru.

2.6 PHPExcel

PHPExcel adalah suatu proyek yang menyediakan berbagai kelas-kelas untuk pemrograman bahasa PHP yang memungkinkan *developer* untuk menulis dan membaca dari berbagai macam bentuk *spreadsheet* seperti Excel (BIFF) .xls, Excel 2007 (OfficeOpenXML) .xlsx, CSV, Libre/OpenOffice Calc .ods, Gnumeric, PDF, HTML, dan lain-lain. Proyek ini dibangun sesuai standar Microsoft OpenXML dan PHP.[5]



Gambar 2.3: Arsitektur PHPExcel

Untuk menjalankan PHPExcel, diperlukan :

- PHP versi 5.2.0 keatas
- PHP extension `php_zip` diaktifkan
- PHP extension `php_xml` diaktifkan
- PHP extension `php_gd2` diaktifkan

2.6.0.1 Membuat *Spreadsheet*

Untuk membuat *spreadsheet* pengguna perlu menggunakan kelas PHPExcel. Kelas PHPExcel ini merupakan bagian inti dari PHPExcel, kelas ini merepresentasikan *workbook* yang akan dibuat. Pada umumnya ada dua cara untuk membuat *workbook* sebagai berikut :[5]

1. Memuat *workbook* dari file *spreadsheet* yang sudah ada.

Cara paling mudah untuk memuat sebuah *workbook* adalah dengan memerintahkan *PHPExcel*

IO Factory untuk mengidentifikasi file *workbook*-nya lalu memuatnya dengan cara memanggil *static method* `load()` dari kelas "*PHPExcel_IOFactory*". Contoh cara pemuatan *workbook* di dalam kode aplikasi adalah sebagai berikut:

```
$inputFileName = './sampleData/example1.xls';

/** Load $inputFileName ke dalam obyek PHPExcel */
$objPHPExcel = PHPExcel_IOFactory::load($inputFileName);
```

Method `load()` bekerja dengan cara mengidentifikasi terlebih dahulu tipe filenya, lalu menginstansiasi sebuah *loader* untuk tipe file terkait. *Loader* ini lalu digunakan untuk memuat *file workbook*-nya dan menyimpan data-data lalu membentuknya ke dalam obyek PHPExcel. [5]

Pada awalnya method `load()` akan memuat *loader* yang dibutuhkan sesuai dari ekstensi *file workbook*, namun method ini akan memeriksa secara mendalam file tersebut sebelum memulai pemuatan *workbook*-nya ke dalam obyek PHPExcel. Sebagai contoh bila sebuah file merupakan file CSV namun memiliki ekstensi `.xls`, maka file tersebut akan menolak *loader* Excel5 yang biasanya digunakan untuk file `.xls`. Bila hal tersebut terjadi, maka method `load()` akan terus mencoba memuat *workbook* tersebut menggunakan *loader-loader* lainnya sampai ditemukan *loader* yang sesuai untuk file tersebut. *Loader* yang sudah cocok tersebut kemudian akan digunakan untuk membaca *file*-nya. [5]

Format file *spreadsheet* yang didukung oleh PHPExcel adalah sebagai berikut:

- BIFF (Excel5)
- SpreadsheetML (Excel2003XML)
- OfficeOpenXML (Excel2007)
- Open Document Format (OOCalc)
- Multiplan SYLK
- Gnumeric
- CSV
- HTML

2. Membuat *workbook* baru secara manual.

Untuk membuat *workbook* baru, instansiasi obyek PHPExcel di dalam kode aplikasi. [5]

```
$objPHPExcel = new PHPExcel();
```

Workbook baru yang dibuat akan memiliki satu buah *worksheet*.

2.6.0.2 Mengakses Sel

Untuk memasukan atau mengubah nilai dalam suatu sel berdasarkan koordinatnya dapat dilakukan dengan menggunakan method `setCellValue()`. [5]

Contoh :

```
// Memasukan nilai string ke sel A1
$objPHPExcel->getActiveSheet()->setCellValue('A1','PHPExcel');

// Memasukan nilai numerik ke sel A2
$objPHPExcel->getActiveSheet()->setCellValue('A2',12345.6789);

// Memasukan nilai boolean ke sel A3
$objPHPExcel->getActiveSheet()->setCellValue('A3',TRUE);

// Memasukan suatu formula ke sel A4
$objPHPExcel->getActiveSheet()->setCellValue(
    'A4',
    '=IF(A3, CONCATENATE(A1," ", A2), CONCATENATE(A2," ", A1))'
);
```

Tujuh tipe data yang didukung MS Excel :

- string
- bilangan
- boolean
- null
- formula
- eror
- *inline string*

Ketika method *setCellValue()* atau *setValue()* dipanggil, PHPEXcel akan memakai tipe data yang sesuai untuk tipe data null, boolean, float atau integer yang didapat dari PHP. PHPEXcel juga bisa mengubah string yang dikirimkan dari PHP ke Excel menjadi tipe data yang lebih sesuai. Sebagai contoh string yang terdiri dari angka-angka saja akan diubah menjadi tipe data numerik, atau string yang memiliki awalan tanda sama dengan "=" akan dianggap sebagai formula.^[5]

Konversi-konversi tipe data tersebut ditangani oleh sebuah "*value binder*" yang dapat diubah-ubah sesuai keinginan pengguna bila pengguna ingin mengatur cara-cara konversinya. PHPEXcel standar juga menyediakan "*advanced value binder*" yang menangani konversi-konversi yang lebih kompleks seperti mengonversi string menjadi bilangan pecahan seperti "3/4" menjadi nilai numerik (untuk kasus ini menjadi 0,75). Fitur ini berguna ketika memuat data dari csv atau memasukan nilai ke dalam sel dari *database*.^[5]

Beberapa format yang ditangani oleh *advanced value binder* adalah sebagai berikut:

- TRUE atau FALSE dikonversi menjadi boolean

- String yang berisi nilai numerik akan diubah menjadi bilangan
- Pecahan akan diubah menjadi bilangan
- Persentase akan diubah menjadi bilangan yang dibagi 100
- Tanggal dan waktu akan diubah menjadi nilai *timestamp* di Excel
- Ketika string mengandung karakter yang memerintahkan pembautan baris baru (n), maka sel akan diatur menjadi menggunakan style "wrap"

2.6.0.3 Memasukan Tanggal dan atau Waktu ke Dalam Sel

Nilai tanggal dan waktu disimpan dalam rupa *timestamp* (bilangan desimal biasa) di dalam Excel. Bilangan tersebut kemudian dibungkus oleh bilangan lain yang menentukan format penulisan tanggalnya. Jadi untuk memasukan tanggal ke dalam sel, perlu dihitung *timestamp* yang benar, dan menentukan bilangan pembungkusnya.[5]

```
// Mendapatkan tanggal dan waktu saat ini
$dateTimeNow = time();
$excelDateValue = PHPExcel_Shared_Date::PHPToExcel( $dateTimeNow );
// Memasukan tanggal dan waktu ke dalam sel A6
$objPHPExcel->getActiveSheet()->setCellValue(
    'A6',
    $excelDateValue
);
// Mengatur format bilangan pembungkus sehingga timestamp dalam Excel bisa
    ↳ ditampilkan dalam format yang dapat dibaca oleh manusia
$objPHPExcel->getActiveSheet()->getStyle('A6')
    ->getNumberFormat()
    ->setFormatCode(
        PHPExcel_Style_NumberFormat::FORMAT_DATE_DATETIME
    );
```

2.6.0.4 Memasukan Data Numerik yang Diawali Angka Nol

Pada umumnya PHPExcel secara otomatis akan mendeteksi tipe dari nilai yang dimasukan dan mengubahnya menjadi tipe data numerik di Excel. Tipe konversi ini ditangani oleh *value binder*. Karena bilangan tidak memiliki awal angka nol, maka jika ada nilai numerik yang memiliki awalan angka nol (misalnya nomor telpon), maka nilai tersebut akan kehilangan angka-angka nol yang berada di depan.[5]

Untuk mencegah konversi demikian, ada beberapa cara untuk melakukannya diantaranya sebagai berikut:

1. Menentukan secara manual di dalam kode agar tipe datanya tidak dikonversi ke dalam bilangan.

```
1 // Memasukan nilai numerik ke dalam sel A8, tetapi memerintahkan
2 ↪ PHPExcel untuk menyimpan nilai tersebut dalam tipe data String
3 $objPHPExcel->getActiveSheet()->setCellValueExplicit(
4     'A8',
5     "01513789642",
6     PHPExcel_Cell_DataType::TYPE_STRING
7 );
```

2. Cara lainnya adalah dengan menentukan format bilangan di dalam worksheet agar menampilkan bilangan dengan awalan nol.

```
10 // Memasukan nilai numerik ke dalam sel A9
11 $objPHPExcel->getActiveSheet()->setCellValue('A9', 1513789642);
12 // Menentukan format sel untuk menampilkan nilai dalam selnya berupa
13 ↪ bilangan 11 digit dengan awalan nol
14 $objPHPExcel->getActiveSheet()->getStyle('A9')
15     ->getNumberFormat()
16     ->setFormatCode(
17         '000000000000'
18     );
```

3. Cara kedua juga dapat digunakan untuk memisahkan digit-digit ke dalam grup agar lebih mudah dibaca.

```
23 // Memasukan nilai numerik ke dalam sel A10
24 $objPHPExcel->getActiveSheet()->setCellValue('A10', 1513789642);
25 //Mengatur format sel untuk menampilkan nilai dalam selnya berupa
26 ↪ bilangan 11 digit dengan awalan nol
27 $objPHPExcel->getActiveSheet()->getStyle('A10')
28     ->getNumberFormat()
29     ->setFormatCode(
30         '0000-000-0000'
31     );
```

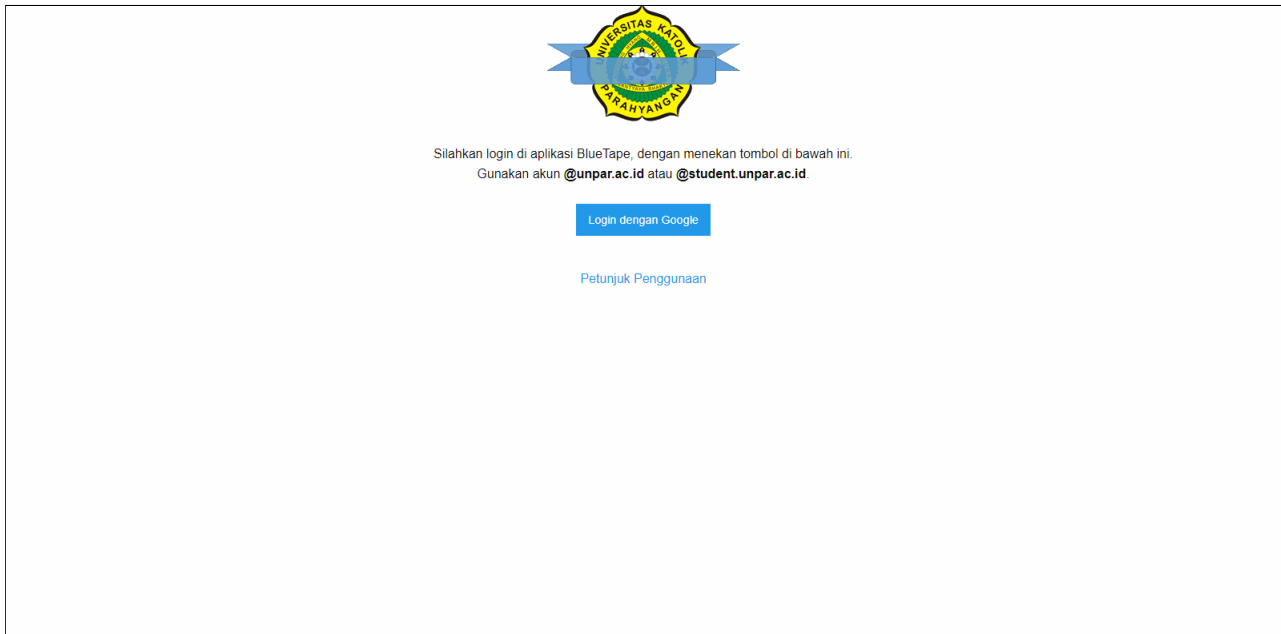

BAB 3

ANALISIS

Pengumpulan data dalam skripsi ini dilakukan dengan cara studi pustaka untuk mempelajari cara pengembangan perangkat lunak menggunakan *Framework* BlueTape yang berbasis CodeIgniter. Selain itu juga dipelajari *library-library* pembantunya diantara lain : PHPExcel , Google OAuth dan ZurbFoundation. Tujuan studi pustaka ini untuk memahami secara rinci cara-cara untuk menambahkan layanan berbentuk modul ke dalam BlueTape dan membangun layanan tersebut menggunakan *library-library* yang disebutkan sebelum ini.

3.1 Analisis Sistem Kini

Aplikasi *Blue Tape* adalah perangkat lunak *open source* sederhana yang memiliki tujuan utama untuk mengubah berbagai pekerjaan *paper-based* di FTIS UNPAR menjadi *paperless*. Selain itu perangkat lunak ini memiliki beberapa kegunaan lainnya seperti mengautentikasi mahasiswa dan staf UNPAR via OAuth 2.0 ke Google (layanan OAuth ke Google ini juga dapat digunakan untuk menentukan hak akses yang bisa dilihat dari email pengguna) dan *Pilot Project* untuk permohonan transkrip ke Tata Usaha . Aplikasi ini merupakan aplikasi berbasis web dengan memanfaatkan *Codeigniter* dan *Zurb Foundation*.



Gambar 3.1: Halaman Utama BlueTape Untuk Login dan Membuka Petunjuk Penggunaan

1 Perangkat lunak *Blue Tape* ini didesain sebagai *framework* yang terdiri dari beberapa layanan
2 yang dipisahkan ke dalam modul-modul. Pemisahan layanan ke dalam modul-modul dibuat
3 dengan tujuan agar pemeliharaan perangkat lunak lebih mudah dan juga mempermudah cara untuk
4 menambahkan layanan baru ke dalam BlueTape. Sudah ada layanan yang aktif di BlueTape saat
5 ini yaitu *Transcript Request / Manage* yang memiliki fungsi untuk melakukan permohonan serta
6 pencetakan transkrip mahasiswa.

7 3.1.1 *Transcript Request / Manage*

8 Modul *Transcript Request/Manage* merupakan salah satu dari dua modul yang sudah aktif di
9 BlueTape ketika skripsi ini ditulis. Modul ini memiliki fungsi utama sebagai alat bagi mahasiswa
10 untuk memohon pencetakan transkrip ke Tata Usaha dengan tampilan seperti di bawah ini :

Permohonan Baru

Yang memohon: NPM: Nama:

Tipe Transkrip: Keperluan:

[Kirim Permohonan](#)

Histori Permohonan

ID	Status	Tanggal Permohonan	Tipe Transkrip	Tanggal Jawab/Cetak	Keterangan	Aksi
----	--------	--------------------	----------------	---------------------	------------	------

Gambar 3.2: Tampilan Cetak Transkrip

Permintaan Transkrip

Cari NPM: [Cari](#)

ID	Status	Tanggal Permohonan	Tipe Transkrip	NPM	Aksi
#1	MENUNGGU	Senin, 16 Oktober 2017	DPS_ID	-	View Edit Print Delete

Gambar 3.3: Tampilan Manajemen Transkrip

- 1 Untuk memohon pencetakan transkrip pada halaman tersebut mahasiswa diperlukan untuk
- 2 melakukan :
- 3 1. Memilih salah satu dari 3 tipe transkrip yang ada. Ada 3 tipe transkrip: Daftar Perkembangan
- 4 Studi (DPS) Bahasa Inggris, DPS Bahasa Indonesia dan Lembar Hasil Studi Semester Akhir.
- 5 2. Lalu mahasiswa mengisi keterangan keperluan pencetakan transkrip di *field* keperluan.
- 6 3. Setelah kedua hal tersebut dipilih dan diisi , tekan tombol "Kirim Permohonan" untuk
- 7 memohon pencetakan transkrip ke Tata Usaha.

1 Tata usaha memiliki beberapa opsi untuk merespon permintaan yang sudah dikirim mahasiswa tadi
2 :

- 3 1. Melihat detail permintaan transkrip
- 4 2. Menolak permintaan transkrip
- 5 3. Mencetak detail permintaan transkrip
- 6 4. Menghapus permintaan transkrip

7 3.1.2 Perubahan Kuliah

8 Perubahan kuliah adalah modul kedua yang sudah aktif di BlueTape. Modul ini berfungsi sebagai
9 alat bagi mahasiswa untuk mengirimkan permintaan perubahan kuliah dari mahasiswa ke karyawan
10 tata usaha. Selain itu karyawan tata usaha juga dapat menggunakan modul ini untuk mengatur
11 permintaan-permintaan yang sudah dikirim dari mahasiswa tadi.

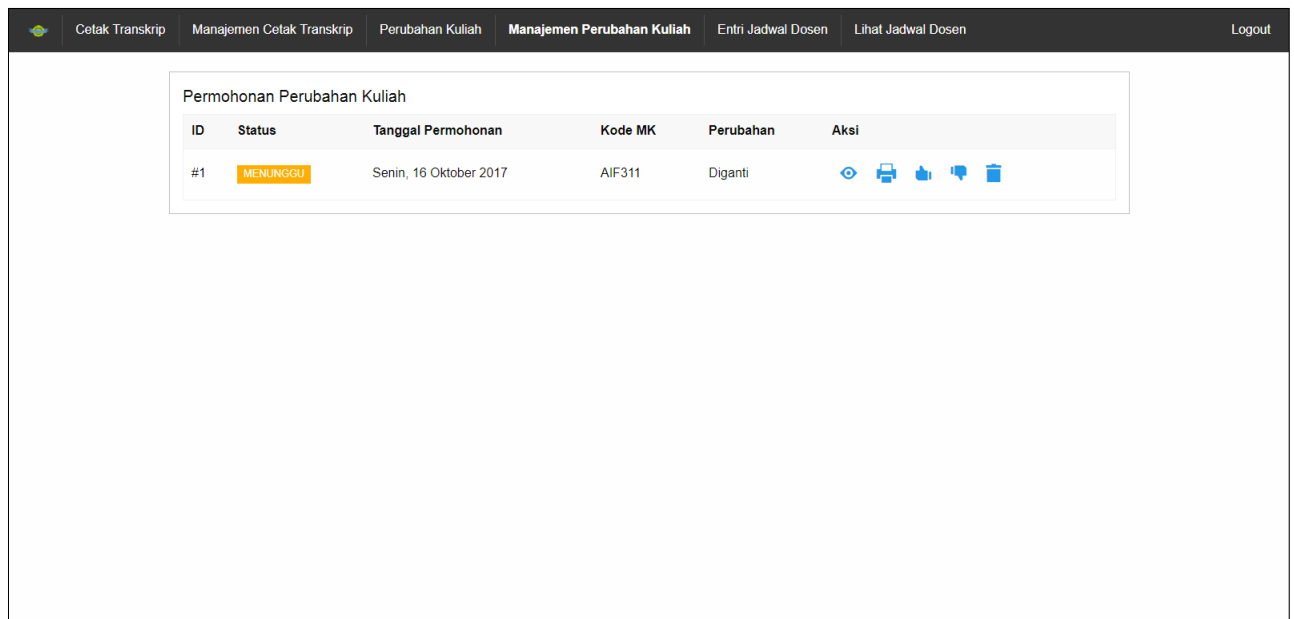
The screenshot displays the 'Perubahan Kuliah' (Change Class) menu. At the top, there is a navigation bar with links: 'Cetak Transkrip', 'Manajemen Cetak Transkrip', 'Perubahan Kuliah' (active), 'Manajemen Perubahan Kuliah', 'Entri Jadwal Dosen', 'Lihat Jadwal Dosen', and 'Logout'.

The main content area is divided into two sections:

- Permohonan Baru (New Request):** This section contains a form for submitting a new request. It includes fields for 'Pemohon' (Requester) with the email 'adrianreynaldi@yahoo.com', 'Nama' (Name) 'Adrian Reynaldi', 'Kode MK' (Course Code), 'Nama Mata Kuliah' (Course Name), 'Kelas' (Class), and 'Jenis Perubahan' (Type of Change) set to 'Diganti' (Changed). There are also fields for 'Dari Hari & Jam' (From Day & Time), 'Dari Ruang' (From Room), 'Keterangan Tambahan' (Additional Remarks), 'Menjadi Hari & Jam' (To Day & Time), and 'Menjadi Ruang' (To Room). At the bottom of this section are two buttons: 'Kirim Permohonan' (Send Request) and 'Tambah Pertemuan Ekstra' (Add Extra Meeting).
- Histori Permohonan (Request History):** This section contains a table showing the history of requests.

ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Tanggal Jawab	Keterangan	Aksi
#1	TUNGGU	Senin, 16 Oktober 2017	AIF311	Diganti			Detail

Gambar 3.4: Tampilan Menu Perubahan Kuliah



Gambar 3.5: Tampilan Menu Manajemen Perubahan Kuliah

Untuk membuat permintaan perubahan kuliah, mahasiswa perlu mengisi setiap *field* yang ada di menu. Setelah semua *field* diisi, mahasiswa bisa memilih untuk mengirim permohonan atau tambah pertemuan.

- Tombol "Kirim Permohonan" akan memeriksa apakah data yang sudah dimasukan sudah benar, bila sudah maka data permohonan akan dikirim ke server dan apabila ada yang masih salah maka sistem akan menandai bagian yang salah.
- Tombol "Tambah Pertemuan Ekstra" akan memunculkan 2 *field* baru yaitu *field* "Menjadi Hari dan jam" dan *field* "Menjadi Ruang". Bila mahasiswa ingin mengirimkan permintaan penambahan pertemuan, tetap harus menekan tombol "Kirim Permohonan" setelah kedua *field* yang baru tersebut juga diisi.

Di sisi tata usaha , modul ini memiliki fungsi untuk :

- Menolak permintaan perubahan kuliah
- Menyetujui permintaan perubahan kuliah
- Mencetak detail perubahan kuliah
- Melihat detail perubahan kuliah
- Menghapus permintaan perubahan kuliah

3.1.3 Pengguna Aplikasi

Pada bagian ini akan dijelaskan pengelompokan tipe-tipe pengguna aplikasi BlueTape.

3.1.3.1 Mahasiswa FTIS

Mahasiswa FTIS adalah semua mahasiswa Fakultas Teknik Informasi dan Sains. Saat ini golongan Mahasiswa FTIS baru memiliki satu akses yaitu untuk memnita transkrip.

3.1.3.2 Tata Usaha UNPAR

Tata Usaha UNPAR adalah golongan pengguna yang bekerja sebagai staff tata usaha di Universitas Katolik Parahyangan. Di BlueTape, Tata Usaha UNPAR memiliki akses fitur-fitur sebagai berikut :

- Mengatur permintaan transkrip dari mahasiswa
- Mengatur permintaan perubahan kuliah

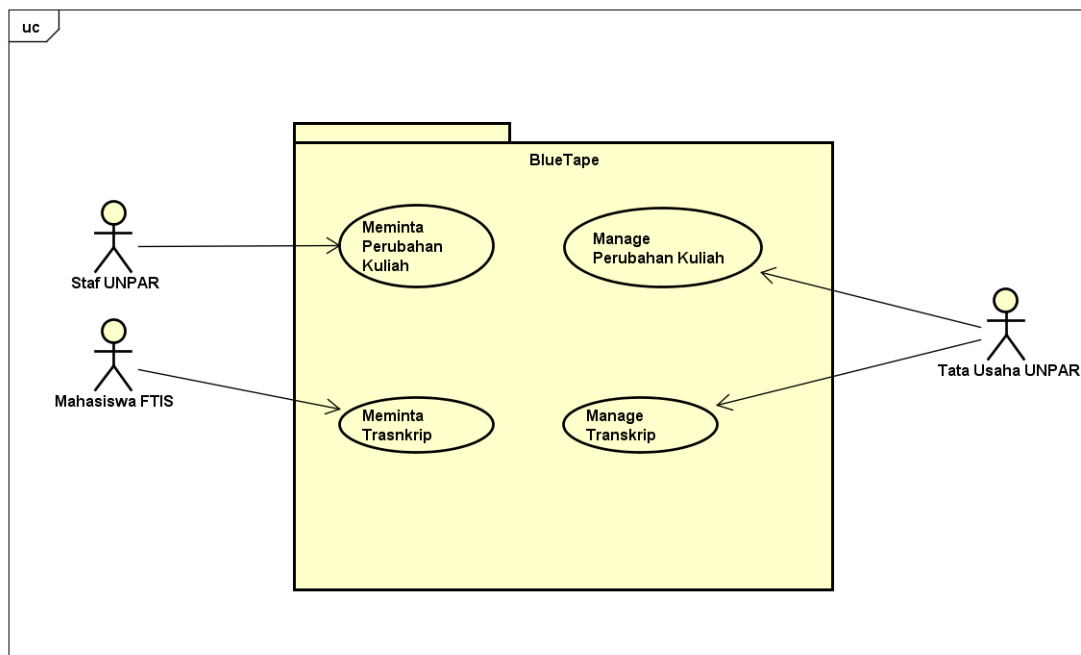
3.1.3.3 Staf UNPAR

Staf UNPAR adalah para pekerja dan karyawan di Universitas Katolik Parahyangan. Untuk saat ini golongan staf UNPAR hanya memiliki akses fitur untuk meminta perubahan kuliah.

3.1.3.4 Mahasiswa Informatika

Mahasiswa Informatika adalah pengguna yang memiliki kepentingan untuk mencetak transkrip nilai. Di Bluetape saat ini, Mahasiswa Informatika baru memiliki satu fitur yaitu untuk meminta transkrip nilai.

3.1.4 Diagram Use Case Sistem Kini



Gambar 3.6: Diagram Use Case Sistem Kini

Berikut ini adalah penjelasan skenario dari diagram *use case* dari Gambar 3.6 atas:

1. Skenario Meminta Perubahan Kuliah

- Aktor : Staf UNPAR, Tata Usaha UNPAR

- Skenario Normal

1. Staf UNPAR mengisi data-data yang diperlukan untuk melakukan perubahan kuliah
2. Sistem mengirim data-data tersebut ke server
3. Tata Usaha menerima data yang dikirim oleh Staf UNPAR untuk diproses

- Skenario *Exception*

1. Staf UNPAR memasukan data secara tidak lengkap
2. Sistem menampilkan pesan eror karena data tidak lengkap

2. Skenario Melihat Detail Perubahan Kuliah

- Aktor : Tata Usaha UNPAR

- Skenario Normal

1. Aktor memilih data perubahan kuliah
2. Aktor memilih menu lihat detail
3. Sistem menampilkan semua data-data permintaan mahasiswa dengan npm terkait

3. Skenario Menolak Perubahan Kuliah

- Aktor : Staf UNPAR, Tata Usaha UNPAR

- Skenario Normal

1. Tata Usaha UNPAR memilih data perubahan kuliah
2. Tata Usaha UNPAR memilih menu tolak perubahan kuliah
3. Tata Usaha UNPAR mengisi data alasan penolakan
4. Sistem mengirim pesan berisi alasan penolakan ke staf UNPAR
5. Sistem menghapus permintaan perubahan kuliah dari *database*

4. Skenario Menyetujui Perubahan Kuliah

- Aktor : Staf UNPAR, Tata Usaha UNPAR

- Skenario Normal

1. Tata Usaha UNPAR memilih data perubahan kuliah
2. Tata Usaha UNPAR memilih menu tolak perubahan kuliah
3. Tata Usaha UNPAR mengisi keterangan penyetujuan
4. Sistem mengirim pesan berisi keterangan penyetujuan ke staf UNPAR
5. Sistem menghapus permintaan perubahan kuliah dari *database*

5. Skenario Menghapus Perubahan Kuliah

- Aktor : Tata Usaha UNPAR

- Skenario Normal

1. Tata Usaha UNPAR memilih data perubahan kuliah
2. Tata Usaha UNPAR memilih menu hapus
3. Sistem menghapus permintaan perubahan kuliah dari *database*

6. Skenario Cetak Detail Perubahan Kuliah

- Aktor : Tata Usaha UNPAR
- Skenario Normal
 1. Tata Usaha UNPAR memilih data perubahan kuliah
 2. Tata Usaha UNPAR memilih menu cetak
 3. Sistem mencetak detail permintaan perubahan kuliah

7. Skenario Meminta Transkrip

- Aktor : Mahasiswa FTIS
- Skenario Normal
 1. Mahasiswa FTIS mengisi data keterangan dan tipe transkrip
 2. Sistem menyimpan data-data tersebut ke server

8. Skenario Lihat Detail Permintaan Transkrip

- Aktor : Tata Usaha UNPAR
- Skenario Normal
 1. Tata Usaha UNPAR memilih data permintaan
 2. Tata Usaha UNPAR memilih menu lihat detail
 3. Sistem menampilkan detail data permintaan

9. Tolak Permintaan Transkrip

- Aktor : Tata Usaha UNPAR
- Skenario Normal
 1. Tata Usaha UNPAR memilih data permintaan
 2. Tata Usaha UNPAR memilih menu tolak permintaan
 3. Sistem menampilkan menu keterangan penolakan
 4. Tata Usaha UNPAR mengisi data keterangan penolakan
 5. Sistem mengirim pesan berisi keterangan penolakan ke Mahasiswa FTIS.

10. Hapus Permintaan Transkrip

- Aktor : Tata Usaha UNPAR
- Skenario Normal
 1. Tata Usaha UNPAR memilih data permintaan
 2. Tata Usaha UNPAR memilih menu hapus
 3. Sistem menampilkan pesan konfirmasi
 4. Tata Usaha UNPAR mengkonfirmasi penghapusan
 5. Sistem menghapus data permintaan dari *database*

3.2 Analisis Sistem Usulan: Aplikasi Pembangkit Jadwal Dosen

Aplikasi Pembangkit Jadwal Dosen adalah aplikasi tambahan untuk BlueTape yang memiliki dua modul yaitu:

1. Modul Entri Jadwal Dosen yang bisa melakukan:

- Input jadwal mingguan
- Mencatat update terakhir
- Ekspor jadwal ke XLS

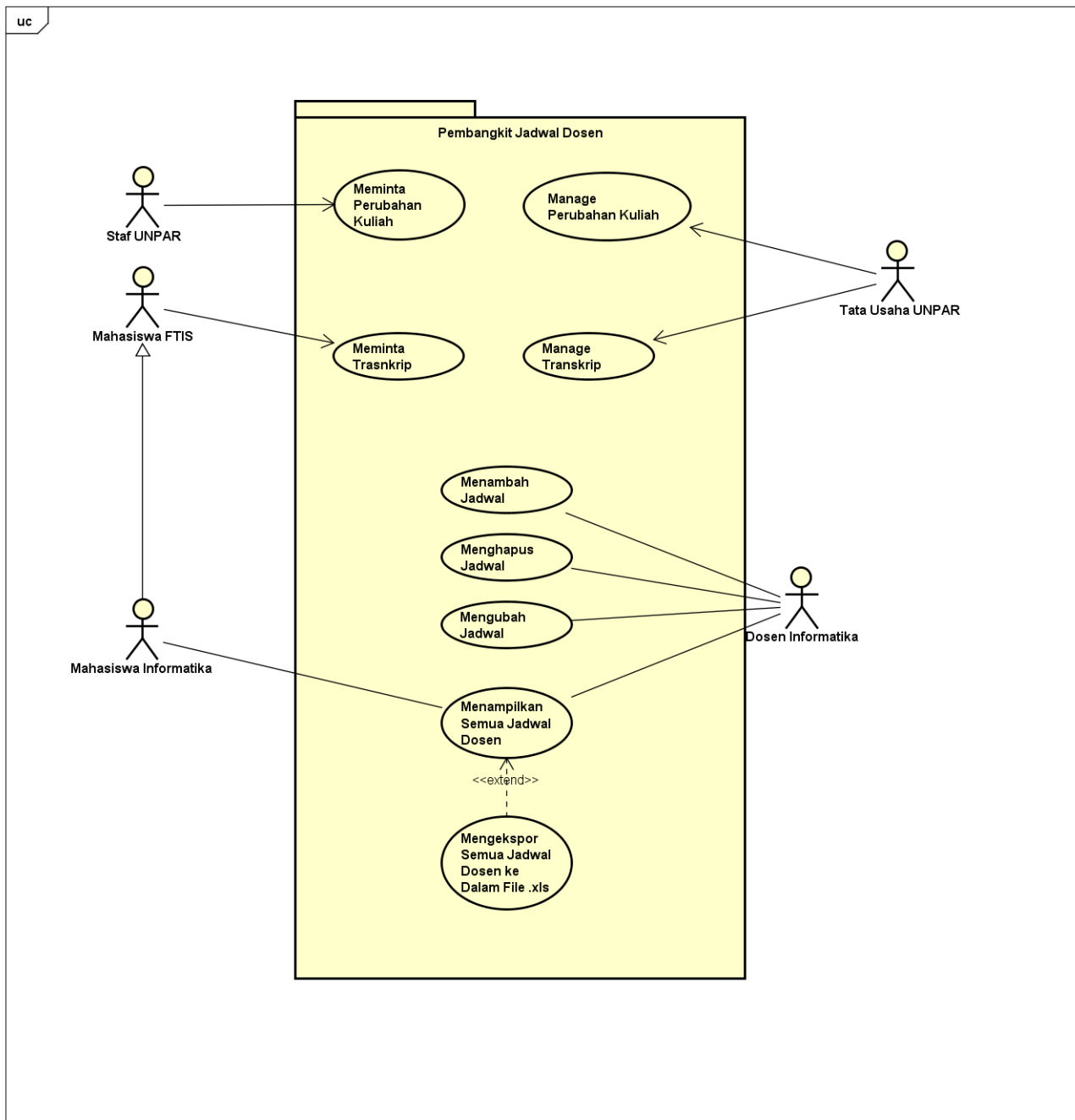
2. Modul Lihat Jadwal Dosen yang bisa melakukan:

- Melihat jadwal-jadwal semua dosen
- Ekspor jadwal ke XLS

Untuk mengimplementasikan fitur-fitur tersebut maka digunakan teknologi:

- PHP & CodeIgniter
- Zurb Foundation
- Framework BlueTape
- Google OAuth
- PHPExcel

3.2.1 Diagram Use Case Sistem Usulan



Gambar 3.7: Diagram Use Case Sistem Usulan

Tidak ada perubahan pada diagram *usecase* dan skenario dari sistem ini (bisa dilihat pada Gambar 3.6) hanya ditambahkan bagian untuk Aplikasi Pembangkit Jadwal Dosen. Berikut ini adalah penjelasan skenario yang ditambahkan dari diagram *usecase* pada Gambar 3.7:

1. Skenario Menambah Jadwal

- Aktor : Dosen Informatika
- Skenario Normal

1. Dosen Informatika mengisi data jadwal
2. Sistem menyimpan data jadwal tersebut ke dalam *database*
2. Skenario Mengubah Jadwal
 - Aktor : Dosen Informatika
 - Skenario Normal
 1. Dosen Informatika memilih jadwal yang akan diubah
 2. Sistem menampilkan menu perubahan berisi data jadwal yang dipilih tadi
 3. Dosen Informatika mengubah data-data yang ingin diubah
 4. Sistem menyimpan hasil perubahan ke dalam database
3. Skenario Menghapus Jadwal
 - Aktor : Dosen Informatika
 - Skenario Normal
 1. Dosen Informatika memilih jadwal yang akan dihapus
 2. Sistem menampilkan menu perubahan berisi data jadwal yang dipilih
 3. Dosen Informatika menekan tombol hapus
 4. Sistem menghapus data jadwal tersebut yang ada di database
4. Skenario Menampilkan Semua Jadwal Dosen
 - Aktor : Mahasiswa Informatika , Dosen Informatika
 - Skenario Normal
 1. Aktor memilih menu lihat jadwal dosen
 2. Sistem memuat semua data jadwal-jadwal dari *database*
 3. Sistem mengelompokan setiap jadwal berdasarkan pemiliknya
 4. Sistem membuat tab-tab yang merepresentasikan setiap dosen yang sudah menyimpan jadwal ke dalam database
 5. Sistem memasukan jadwal-jadwal ke dalam tab-tab sesuai nama pemiliknya.
5. Skenario Mengekspor Semua Jadwal Dosen ke Dalam File .xls
 - Aktor : Mahasiswa Informatika , Dosen Informatika
 - Skenario Normal
 1. Aktor memilih menu lihat jadwal dosen
 2. Sistem memuat semua data jadwal-jadwal dari *database*
 3. Sistem mengelompokan setiap jadwal berdasarkan pemiliknya
 4. Sistem membuat tab-tab yang merepresentasikan setiap dosen yang sudah menyimpan jadwal ke dalam database
 5. Sistem memasukan jadwal-jadwal ke dalam tab-tab sesuai nama pemiliknya.
 6. Aktor menekan tombol ekspor

7. Sistem mengkonversi semua data jadwal dari bentuk php ke dalam bentuk *spreadsheet* .xls

8. Sistem menampilkan menu pemilihan lokasi penyimpanan *file* .xls

9. Aktor menekan tombol simpan

10. Sistem menyimpan file .xls tersebut di lokasi yang sudah dipilih oleh aktor.

- Skenario Exception

1. Aktor memilih menu lihat jadwal dosen

2. Sistem memuat semua data jadwal-jadwal dari *database*

3. Sistem tidak menerima data apapun dari *database*

4. Sistem men-*disable* tombol ekspor

3.2.2 Pengguna Aplikasi

Pada bagian ini akan dijelaskan pengelompokan tipe-tipe pengguna aplikasi BlueTape dengan sistem usulan. Perubahan terjadi pada kelompok "Mahasiswa Informatika" yang mendapat fitur tambahan. Selain itu, ada penambahan kelompok pengguna yaitu "Dosen Informatika". Perubahan-perubahan akan dijelaskan pada bagian masing-masing.

3.2.2.1 Mahasiswa FTIS

Mahasiswa FTIS adalah semua mahasiswa Fakultas Teknik Informasi dan Sains. Saat ini golongan Mahasiswa FTIS baru memiliki satu akses yaitu untuk memnita transkrip.

3.2.2.2 Tata Usaha UNPAR

Tata Usaha UNPAR adalah golongan pengguna yang bekerja sebagai staff tata usaha di Universitas Katolik Parahyangan. Di BlueTape, Tata Usaha UNPAR memiliki akses fitur-fitur sebagai berikut :

- Mengatur permintaan transkrip dari mahasiswa
- Mengatur permintaan perubahan kuliah

3.2.2.3 Staf UNPAR

Staf UNPAR adalah para pekerja dan karyawan di Universitas Katolik Parahyangan. Untuk saat ini golongan staf UNPAR hanya memiliki akses fitur untuk meminta perubahan kuliah.

3.2.2.4 Mahasiswa Informatika

Mahasiswa Informatika adalah pengguna yang memiliki kepentingan untuk melihat jadwal-jadwal semua dosen Informatika. Dengan mengetahui jadwal dosen, maka mahasiswa dapat mengatur jadwal bimbingan atau konsultasi lainnya dengan dosen terkait secara lebih teratur.

- Melihat semua jadwal dosen Informatika
- Mengekspor semua jadwal dosen Informatika ke dalam tipe file .xls
- Meminta transkrip nilai

3.2.2.5 Dosen Informatika

Dosen merupakan pengguna yang dapat memasukan jadwal-jadwalnya ke dalam BlueTape agar dapat dilihat mahasiswa. Dosen memiliki akses pada fitur :

- Memasukan jadwal ke dalam BlueTape
- Mengubah jadwal yang sudah dimasukan
- Menghapus jadwal yang sudah dimasukan
- Melihat semua jadwal dosen , termasuk jadwal miliknya sendiri
- Mengespor semua jadwal dosen ke dalam tipe file .xls

BAB 4

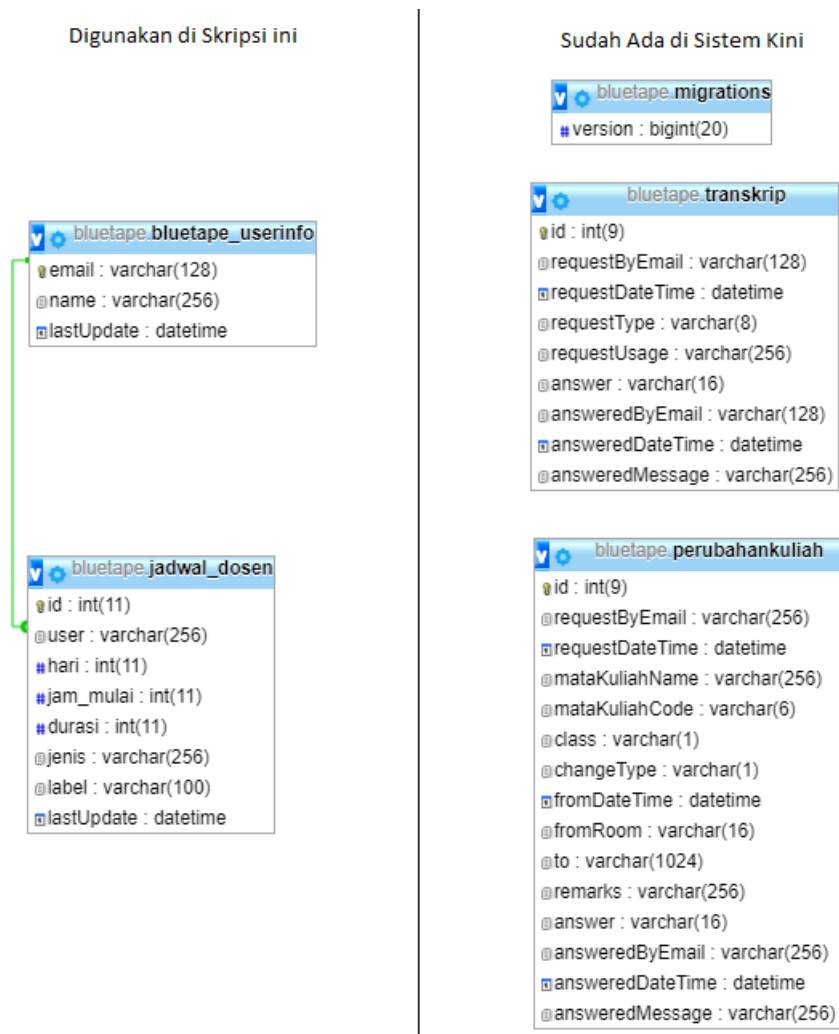
PERANCANGAN DAN IMPLEMENTASI

4.1 Perancangan Basis Data

4.1.1 Diagram Hubungan Entitas

Dari analisis yang telah dilakukan, maka dibuatlah Diagram Hubungan Entitas seperti pada gambar

4.1



Gambar 4.1: Diagram Hubungan Entitas

4.1.2 Perancangan Tabel

Atribut	Tipe Data	Ukuran	PK* / FK*	Keterangan
id	int	11	PK	id jadwal
user	varchar	256	FK	pemilik jadwal
hari	int	11	bukan keduanya	hari berlangsungnya jadwal
jam_mulai	int	11	bukan keduanya	jam berlangsungnya jadwal
durasi	int	11	bukan keduanya	lama jadwal berlangsung
jenis	varchar	256	bukan keduanya	jenis kegiatan jadwal
label	varchar	256	bukan keduanya	nama kegiatan

Tabel 4.1: Perancangan Tabel jadwal_dosen

*PK = Primary Key

Keterangan atribut:

- id:** sebagai penanda yang membedakan setiap jadwal satu sama lain. Memiliki *length default* int dari MySQL yaitu 11. Merupakan *primary key* karena id harus unik agar setiap jadwal dapat dibedakan.
- user:** email pemilik jadwal. *Foreign key* ini memiliki referensi ke atribut *email* di tabel *bluetape_userinfo*. Merupakan *foreign key* untuk mendapatkan data-data milik pengguna.
- hari:** hari berlangsungnya jadwal. Nilai 0 menandakan hari Senin, 1 hari Selasa, 2 hari Rabu, 3 hari Kamis dan 4 hari Jumat. Memiliki *length default* int dari MySQL yaitu 11.
- durasi:** lama berlangsungnya jadwal dalam satuan jam dengan durasi minimal 1 jam. Memiliki *length default* int dari MySQL yaitu 11.
- jenis:** jenis kegiatan jadwal. Menggunakan *varchar* untuk mengantisipasi penambahan fitur pengguna dapat menambahkan jenis kegiatan sendiri.
- label:** nama kegiatan jadwal. Menggunakan *varchar* karena nama label diisi oleh pengguna.

4.1.3 Perancangan Rinci

4.1.3.1 Controller EntriJadwalDosen

Berikut adalah perancangan kelas Controller EntriJadwalDosen

Nama Method	insert
Parameter Input	data_jadwal[] via POST
Parameter Output	redirect base_URL/EntriJadwalDosen
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk memasukan jadwal
Algoritma	<ul style="list-style-type: none"> • proses menerima input-input data dari user • proses mengecek apakah data-data yang dimasukan sudah valid • proses mengecek apakah jadwal yang bertabrakan dengan jadwal lain atau tidak • bila jadwal bertabrakan dengan jadwal lain maka proses menampilkan pesan "Jadwal gagal dimasukan. Sudah ada jadwal lain pada waktu ini" • bila tidak ada masalah, proses akan memasukan data ke dalam database

Tabel 4.2: Perincian method insert

Nama Method	edit
Parameter Input	id_jadwal,data_jadwal[] via POST
Parameter Output	redirect base_URL/EntriJadwalDosen
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk mengubah jadwal yang sudah dimasukan sebelumnya
Algoritma	<ul style="list-style-type: none"> • proses menerima input-input data dari user • proses mengecek apakah data-data yang dimasukan sudah valid • proses mengecek apakah waktu baru dari jadwal yang diubah bertabrakan dengan jadwal lain atau tidak • bila jadwal bertabrakan dengan jadwal lain maka proses menampilkan pesan "Pengubahan gagal. Sudah ada jadwal lain pada waktu ini" • bila tidak ada masalah, maka jadwal akan memperbarui data yang ada di database.

Tabel 4.3: Perincian method edit

Nama Method	delete
Parameter Input	id_jadwal via POST
Parameter Output	redirect base_URL/EntriJadwalDosen
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk menghapus jadwal
Algoritma	<ul style="list-style-type: none"> • proses menerima pilihan jadwal yang akan dihapus oleh user. • proses menghapus jadwal berdasarkan id_jadwal yang diterima dari jadwal yang dipilih user.

Tabel 4.4: Perincian method *delete*

1 4.1.3.2 Controller LihatJadwalDosen

2 Berikut adalah perincian rancangan kelas Controller LihatJadwalDosen

3

Nama Method	index
Parameter Input	-
Parameter Output	menampilkan semua jadwal dalam bentuk tabel
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk memperlihatkan jadwal ke user
Algoritma	<ul style="list-style-type: none"> • proses memuat semua data jadwal dari database. • proses mengelompokkan jadwal-jadwal dari database tersebut berdasarkan pemiliknya. • proses membuat tab-tab setiap tab diberi nama dosen yang sudah memasukan jadwal ke dalam sistem • proses memasukan setiap jadwal yang sudah dipisahkan berdasarkan pemilik ke dalam tab-tab sesuai dengan nama pemilik yang bersangkutan. • secara default akan ditampilkan jadwal dosen yang pertama kali dimuat oleh proses. • bila user menekan tab, proses akan menampilkan semua jadwal dosen terkait

Tabel 4.5: Perincian method index

Nama Method	ekspor
Parameter Input	-
Parameter Output	file .xls , redirect base_URL/LihatJadwalDosen
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk mengkonversi jadwal dari php ke dalam tipe file .xls
Algoritma	<ul style="list-style-type: none"> • proses memuat semua data jadwal dari database. • proses mengelompokkan jadwal-jadwal dari database tersebut berdasarkan pemiliknya. • proses membuat tab-tab di dalam spreadsheet setiap tab diberi nama dosen yang sudah memasukan jadwal ke dalam sistem • proses memasukan setiap jadwal yang sudah dipisahkan berdasarkan pemilik ke dalam tab-tab di speradsheet sesuai dengan nama pemilik yang bersangkutan. • secara default akan ditampilkan jadwal dosen yang pertama kali dimuat oleh proses. • bila user menekan tab, proses akan menampilkan semua jadwal dosen terkait

Tabel 4.6: Perincian method ekspor

1 4.1.3.3 Model JadwalDosen

2 Berikut adalah perincian rancangan kelas Model JadwalDosen

Nama Method	add_jadwal
Parameter Input	data_jadwal[] via POST
Parameter Output	data jadwal ditambahkan ke database
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk memasukan jadwal ke dalam <i>database</i>
Algoritma	<ul style="list-style-type: none"> • proses menerima data dari user • data-data dimasukan ke dalam <i>database</i>

Tabel 4.7: Perincian method add_jadwal

Nama Method	update_jadwal
Parameter Input	id_jadwal, data_jadwal[] via POST
Parameter Output	memperbarui data jadwal di database berdasarkan id
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk mengupdate data jadwal di <i>database</i> berdasarkan pilihan user
Algoritma	<ul style="list-style-type: none"> • proses menerima data dan id_jadwal • proses mengupdate data di <i>database</i> berdasarkan id_jadwal

Tabel 4.8: Perincian method update_jadwal

Nama Method	delete_jadwal
Parameter Input	id_jadwal via POST
Parameter Output	data jadwal di database dihapus berdasarkan id
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk menghapus data jadwal dari <i>database</i> berdasarkan pilihan user
Algoritma	<ul style="list-style-type: none"> • proses menerima id_jadwal • proses menghapus data jadwal di database berdasarkan id_jadwal

Tabel 4.9: Perincian method delete_jadwal

Nama Method	get_jadwal
Parameter Input	-
Parameter Output	seluruh data jadwal
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk mengambil semua data jadwal
Algoritma	<ul style="list-style-type: none"> • proses memuat semua data jadwal dari database. • proses mengirim semua data yang suda dimuat ke controller pemanggil

Tabel 4.10: Perincian method get_jadwal

1 4.2 Perancangan Antarmuka

2 Pada bagian ini akan dibahas rancangan antarmuka Aplikasi Pembangkit Jadwal Dosen.

3 4.2.1 Perancangan Antarmuka Entri Jadwal Dosen

4 Perancangan tampilan untuk modul Entri Jadwal Dosen dapat dilihat pada gambar di bawah ini

The screenshot shows a web application interface for entering lecturer schedules. At the top, there's a browser window with 'skripsi.com' in the address bar. Below the browser, there's a navigation bar with two tabs: 'Entri Jadwal Dosen' (active) and 'Lihat Jadwal Dosen'. The main content area is divided into two sections: 'Tambah Jadwal' (Add Schedule) and 'Daftar Jadwal' (Schedule List).

The 'Tambah Jadwal' section contains a form with the following fields:

- Hari**: A dropdown menu with 'Senin' selected.
- Durasi**: A dropdown menu with '1 Jam' selected.
- Label**: A text input field.
- Jam Mulai**: A dropdown menu with '7:00' selected.
- Jenis**: A dropdown menu with 'Konsultasi' selected.
- Tambah**: A button to submit the form.

The 'Daftar Jadwal' section displays a table with the following structure:

	Senin	Selasa	Rabu	Kamis	Jumat
7-8					
8-9					
9-10					
10-11	Konsultasi				
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					

Gambar 4.2: Perancangan Antarmuka Entri Jadwal Dosen

Keterangan

1. Menu Tambah Jadwal

Menu ini berisi field-field yang dapat digunakan pengguna untuk menentukan berbagai macam hal mengenai jadwal yang akan dimasukan seperti jam dimulainya jadwal, durasi dan lain-lain.

- field Hari: berisi pilihan hari-hari berlangsungnya jadwal. Ada 5 pilihan hari yaitu hari senin, selasa, rabu, kamis, dan jumat.
- field Jam Mulai: berisi pilihan jam berlangsungnya jadwal. Jam yang dapat dipilih adalah jam 8 sampai jam 16 (jam 4 sore) dan semua pilihan jam tepat pada menit 0, tidak ada pilihan menitnya.
- field Durasi: berisi pilihan lama berlangsungnya jadwal. Ada 10 pilihan mulai dari 1 jam, 2 jam, 3 jam dan seterusnya sampai 10 jam. Sama seperti field Jam Mulai di pilihan ini juga tidak ada pilihan menit.
- field Jenis: berisi pilihan jenis kegiatan jadwalnya. Ada 3 pilihan yaitu Konsultasi, Terjadwal dan Kelas.
- field Label: diisi dengan nama kegiatan jadwal. Field ini dapat dikosongkan.
- tombol Tambah: bila pengguna menekan tombol ini, maka jadwal baru akan ditambahkan sesuai dengan input yang sudah dipilih atau dimasukan oleh pengguna pada field-field di atas.

2. Menu Daftar Jadwal

Tabel pada menu ini merepresentasikan hari dan jam perkuliahan. Bila pada jam tertentu ada suatu jadwal, maka sel-sel di yang berada di antara sel jam mulai sampai sel berakhirnya jadwal akan diwarnai sesuai dengan jenis kegiatan jadwalnya. Jadwal konsultasi akan diwarnai hijau, kelas diwarnai putih dan jadwal lainnya yang terjadwal akan diwarnai kuning. Selain itu ditengah-tengah sel yang diwarnai juga akan ditampilkan nama kegiatan jadwalnya yang sudah pengguna isi pada field Label. Contoh tampilan jadwal ada pada Gambar 4.2 di bagian Daftar Jadwal. Gambar 4.3 di bawah ini menerangkannya secara lebih mendetil.

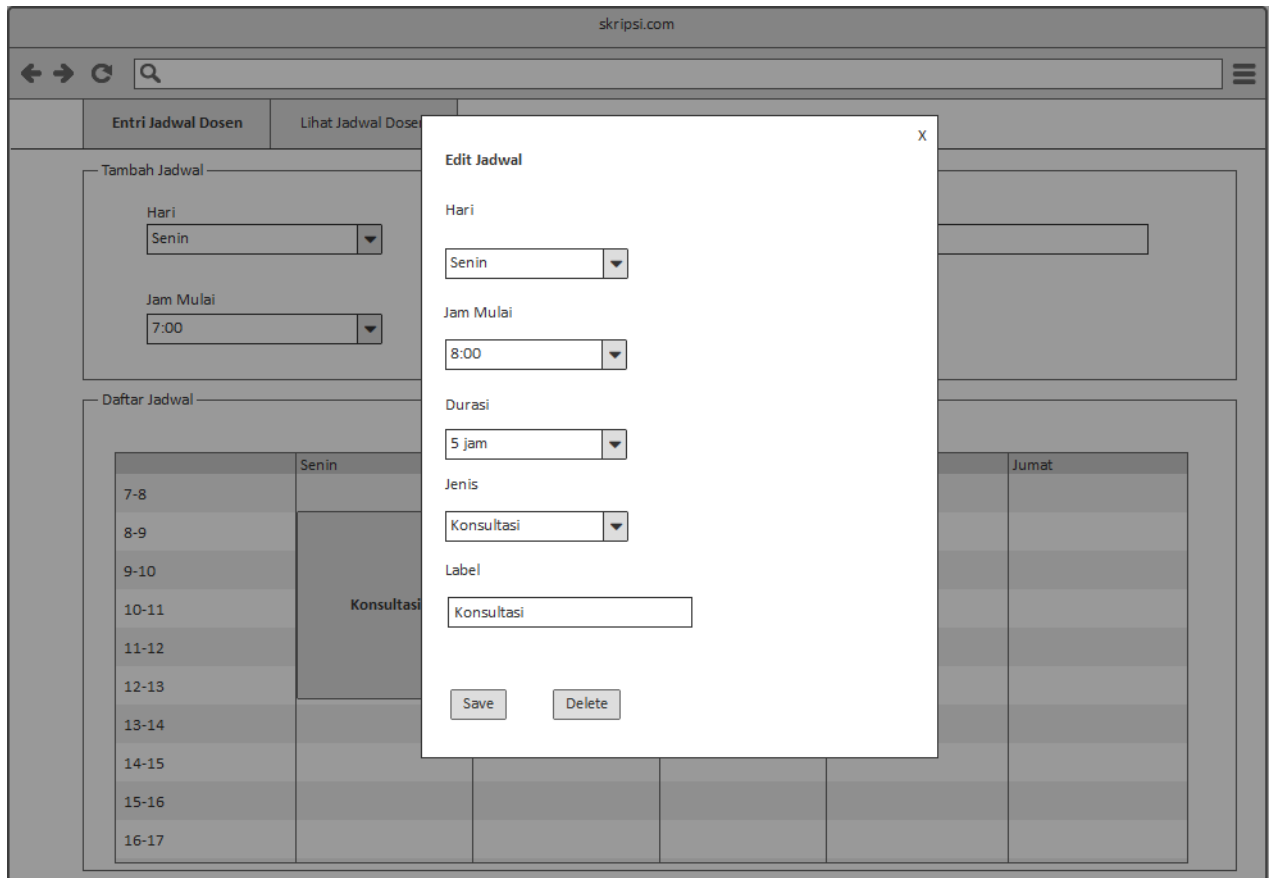
	Senin
7-8	
8-9	Konsultasi
9-10	
10-11	
11-12	
12-13	
13-14	
14-15	
15-16	
16-17	

Gambar 4.3: Contoh Jadwal

Pada contoh diatas berarti jadwal dengan nama kegiatan "Konsultasi" yang diberi warna kuning berlangsung pada hari Senin dari jam 8 pagi sampai jam 1 siang. Warna kuning ini untuk membedakan jenis kegiatan "Konsultasi" dengan jenis kegiatan lainnya misalnya kegiatan "Kelas" diberi warna putih. Jadwal yang ada pada tabel juga dapat ditekan untuk memunculkan menu edit yang akan dijelaskan lebih lanjut di Bagian 4.2.2 di bawah.

4.2.2 Perancangan Antarmuka Edit Jadwal Dosen

Perancangan antarmuka untuk *pop-up* yang berisi menu pengubahan data jadwal yang sudah dimasukan dapat dilihat pada gambar berikut



Gambar 4.4: Perancangan Antarmuka Edit Jadwal Dosen

Keterangan

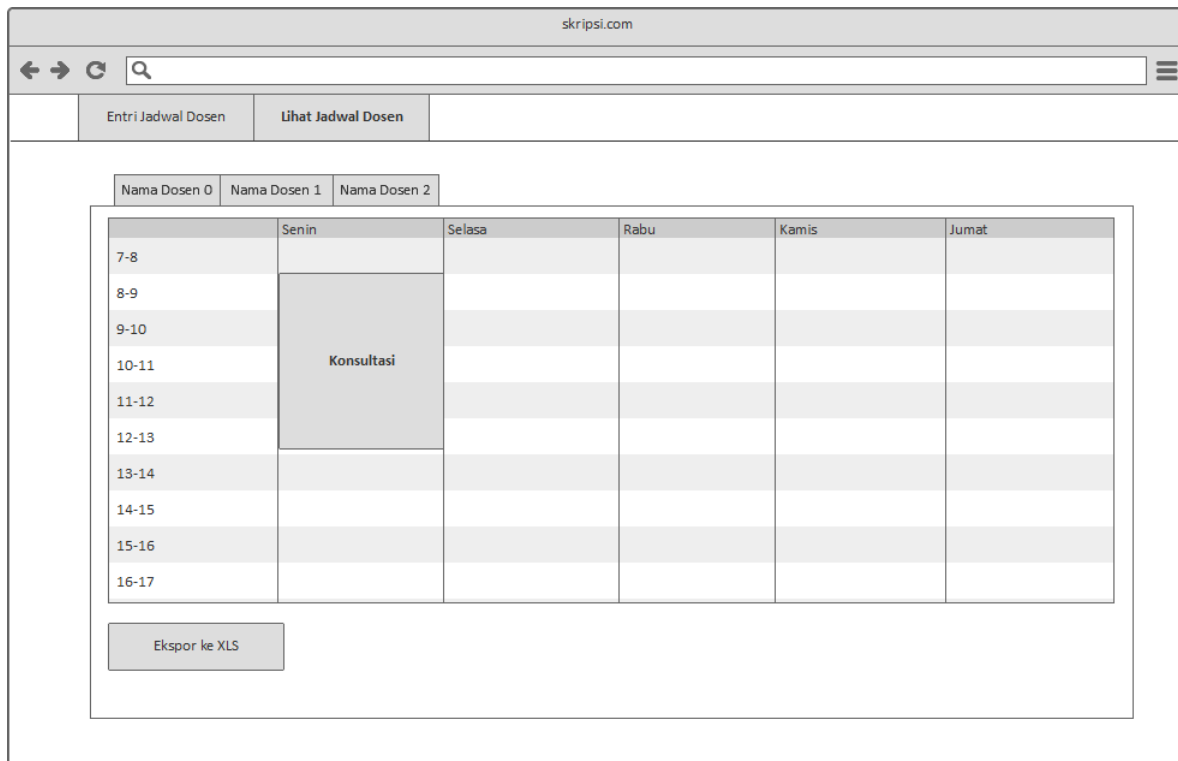
- field Hari: berisi pilihan hari-hari berlangsungnya jadwal. Ada 5 pilihan hari yaitu hari senin, selasa, rabu, Kamis, dan jumat.
- field Jam Mulai: berisi pilihan jam berlangsungnya jadwal. Jam yang dapat dipilih adalah jam 8 sampai jam 16 (jam 4 sore) dan semua pilihan jam tepat pada menit 0, tidak ada pilihan menitnya.
- field Durasi: berisi pilihan lama berlangsungnya jadwal. Ada 10 pilihan mulai dari 1 jam, 2 jam, 3 jam dan seterusnya sampai 10 jam. Sama seperti field Jam Mulai di pilihan ini juga tidak ada pilihan menit.
- field Jenis: berisi pilihan jenis kegiatan jadwalnya. Ada 3 pilihan yaitu Konsultasi, Terjadwal dan Kelas.
- field Label: diisi dengan nama kegiatan jadwal. Field ini dapat dikosongkan.
- tombol Tambah: bila pengguna menekan tombol ini, maka jadwal baru akan ditambahkan sesuai dengan input yang sudah dipilih atau dimasukan oleh pengguna pada field-field di atas.

Berdasarkan keterangan di atas menu Edit Jadwal Dosen ini sama persis dengan menu Entri Jadwal Dosen. Satu-satunya perbedaan adalah setiap *field* menampilkan keterangan detail jadwal yang

- 1 dipilih tersebut. Misalkan jadwal yang dipilih berlangsung dari jam 8 sampai 9, maka *field* jam
- 2 mulai akan menampilkan jam 8 dan *field* durasi menampilkan 1 jam.

3 4.2.3 Perancangan Antarmuka Lihat Jadwal Dosen

- 4 Perancangan antarmuka untuk modul Lihat Jadwal Dosen dapat dilihat pada gambar di bawah ini



Gambar 4.5: Perancangan Antarmuka Lihat Jadwal Dosen

5 Keterangan

- 6 • Tab yang berlabel Nama Dosen 0, Nama Dosen 1, dan Nama Dosen 2 menandakan bahwa
- 7 jadwal yang ditampilkan di tab tersebut merupakan jadwal milik dosen terkait.
- 8 • Tombol "Ekspor ke XLS" berfungsi untuk membuat file XLS yang berisi semua jadwal dosen
- 9 yang ditampilkan di halaman Lihat Jadwal Dosen. Setiap jadwal dosen akan dikelompokkan
- 10 ke dalam worksheet-worksheet.
- 11 Tampilan tabel yang berisi jadwal-jadwal ini sama persis dengan tampilan di halaman Entri Jadwal
- 12 Dosen, perbedaannya adalah jadwal-jadwal yang ada di dalam tabel ini tidak dapat ditekan untuk
- 13 menampilkan menu Edit Jadwal Dosen.

BAB 5

IMPLEMENTASI DAN PENGUJIAN

5.1 Implementasi

5.1.0.1 Lingkungan Pengembangan dan Pengujian Fungsional

Pembangunan perangkat lunak ini memakai bahasa pemrograman server PHP 5 dan MySQL 5.0 sebagai basis datanya. Pengembangan perangkat lunak ini dilakukan di komputer dengan spesifikasi perangkat keras dan perangkat lunaknya sebagai berikut:

- **Perangkat Keras**

1. Prosesor AMD A6 VISION
2. RAM 4 GB
3. 500 GB harddisk
4. Monitor
5. Keyboard
6. Mouse

- **Perangkat Lunak**

1. Sistem Operasi Windows 8
2. NetBeans IDE 8.0.2
3. Server phpMyAdmin 4.3.11
4. XAMPP Control Panel v3.2.1
5. Client: Google Chrome 61.0.3163.100 32-bit
6. Microsoft Excel 2013

5.1.1 Implementasi Basis Data

Implementasi basis data dalam Aplikasi Pembangkit Jadwal Dosen menggunakan satu tabel basis data. Tabel tersebut adalah tabel jadwal_dosen yang menyimpan informasi-informasi seperti:

1. user yang memiliki jadwal terkait
2. hari berlangsungnya jadwal

3. jam dimulainya kegiatan jadwal
4. durasi (dalam jam) lama berlangsungnya jadwal
5. nama kegiatan jadwal
6. waktu terakhir jadwal diupdate

5.1.2 Implementasi Kelas

Implementasi kelas dalam Aplikasi Pembangkit Jadwal Dosen terdiri dari kelas - kelas sebagai berikut:

1. Kelas *controller* `EntriJadwalDosen`
Merupakan kelas yang mengatur hubungan antara kelas view `EntriJadwalDosen` dan kelas model `JadwalDosen_model`.
2. Kelas *controller* `LihatJadwalDosen`
Merupakan kelas yang berfungsi menghubungkan antara kelas view `LihatJadwalDosen` dan kelas model `JadwalDosen_model`.
3. Kelas *view* `EntriJadwalDosen` yang berfungsi membuat *user interface* untuk memasukan jadwal
4. Kelas *view* `LihatJadwalDosen` yang berfungsi membuat *user interface* untuk melihat jadwal
5. Kelas model `JadwalDosen_model` yang berfungsi untuk menulis atau membaca ke basis data.

5.2 Pengujian

Pengujian aplikasi ini dilakukan dengan menggunakan metode pengujian *Black-Box Testing*. Pengujian ini difokuskan pada pengujian fungsional. Maksud dari pengujian fungsional ini adalah untuk menguji reaksi perangkat lunak terhadap aksi yang dilakukan oleh pengguna. Bila reaksi sistem tidak sesuai yang diharapkan, maka aplikasi masih memiliki kekurangan.

5.2.1 Pengujian Fungsional

5.2.1.1 Pengujian Fungsional Login

Hasil pengujian dapat dilihat pada Tabel 5.1

Aksi	Reaksi yang diharapkan	Reaksi PL
Menekan tombol login	Menampilkan menu login	Menampilkan menu login

Tabel 5.1: Pengujian Login

1 5.2.1.2 Pengujian Fungsional Entri Jadwal Dosen

2 Hasil pengujian dapat dilihat pada Tabel 5.2

Aksi	Reaksi yang diharapkan	Reaksi PL
Memasukan data-data yang diperlukan untuk input jadwal	Jadwal yang sudah dimasukan muncul pada halaman Entri Jadwal Dosen	Jadwal yang sudah dimasukan muncul pada halaman Entri Jadwal Dosen
Data jadwal yang dimasukan tidak lengkap	Jadwal dimasukan dengan data yang sesuai nilai <i>default</i> yang ditampilkan di menu	Jadwal dimasukan dengan data yang sesuai nilai <i>default</i> yang ditampilkan di menu
Waktu dimulainya jadwal baru sama dengan jadwal yang sudah pernah dimasukan sebelumnya	Menampilkan pesan kesalahan dan batal memasukan data ke dalam database	Jadwal masuk ke dalam database, namun tidak muncul di halaman Entri Jadwal Dosen maupun halaman Lihat Jadwal Dosen

Tabel 5.2: Pengujian Fungsional Entri Jadwal Dosen

3

4 5.2.1.3 Pengujian Fungsional Edit Jadwal Dosen

5 Hasil pengujian dapat dilihat pada Tabel 5.3

Aksi	Reaksi yang diharapkan	Reaksi PL
Memilih jadwal yang akan di-edit	Memunculkan modal yang berisi menu edit yang menampilkan data-data jadwal yang sedang dipilih tersebut. Data-data tersebut juga dapat diubah oleh pengguna sesuai kebutuhannya	Memunculkan modal yang berisi menu edit yang menampilkan data-data jadwal yang sedang dipilih tersebut. Data-data tersebut juga dapat diubah oleh pengguna sesuai kebutuhannya
Memilih tombol simpan	Modal ditutup , data jadwal di <i>database</i> diupdate sesuai dengan input user dan menampilkan menu Entri Jadwal Dosen	Modal ditutup , data jadwal di <i>database</i> diupdate sesuai dengan input user dan menampilkan menu Entri Jadwal
Memilih tombol keluar	Modal ditutup lalu menampilkan menu Entri Jadwal Dosen	Modal ditutup lalu menampilkan menu Entri Jadwal Dosen

Tabel 5.3: Pengujian Fungsional Edit Jadwal Dosen

6

7 5.2.1.4 Pengujian Fungsional Hapus Jadwal Dosen

8 Hasil pengujian dapat dilihat pada Tabel 5.4

Aksi	Reaksi yang diharapkan	Reaksi PL
Memilih jadwal yang akan di-edit	Memunculkan modal yang berisi menu edit yang menampilkan data-data jadwal yang sedang dipilih tersebut. Data-data tersebut juga dapat diubah oleh pengguna sesuai kebutuhannya	Memunculkan modal yang berisi menu edit yang menampilkan data-data jadwal yang sedang dipilih tersebut. Data-data tersebut juga dapat diubah oleh pengguna sesuai kebutuhannya
Memilih tombol hapus	Modal ditutup dan jadwal dihapus dari basis data, lalu sistem menampilkan menu Entri Jadwal Dosen	Modal ditutup dan jadwal dihapus dari basis data, lalu sistem menampilkan menu Entri Jadwal Dosen
Memilih tombol keluar	Modal ditutup lalu menampilkan menu Entri Jadwal Dosen	Modal ditutup lalu menampilkan menu Entri Jadwal Dosen

Tabel 5.4: Pengujian Fungsional Hapus Jadwal Dosen

1

2 5.2.1.5 Pengujian Fungsional Lihat Jadwal Dosen

3 Hasil pengujian dapat dilihat pada Tabel 5.5

Aksi	Reaksi yang diharapkan	Reaksi PL
Menekan tombol Lihat Jadwal Dosen	Sistem menampilkan semua jadwal dosen dan mengelompokkannya ke dalam tab-tab berdasarkan email setiap pemilik jadwal terkait. Kemudian setiap tab diberi label berisi nama lengkap pemilik jadwal tersebut.	Sistem menampilkan semua jadwal dosen dan mengelompokkannya ke dalam tab-tab berdasarkan email setiap pemilik jadwal terkait. Kemudian setiap tab diberi label berisi nama lengkap pemilik jadwal tersebut.

Tabel 5.5: Pengujian Fungsional Lihat Jadwal Dosen

4

5 5.2.1.6 Pengujian Fungsional Ekspor ke XLS

6 Hasil pengujian dapat dilihat pada Tabel 5.6

Aksi	Reaksi yang diharapkan	Reaksi PL
Menekan tombol ekspor ke XLS	Sistem membuat file .xlsx yang berisi seluruh jadwal dosen yang dikelompokkan ke dalam <i>worksheet-worksheet</i> berdasarkan <i>email</i> pemilik jadwal, lalu sistem menampilkan menu untuk memilih lokasi penyimpanan file .xlsx yang akan diekspor ke komputer pengguna.	Sistem membuat file .xlsx yang berisi seluruh jadwal dosen yang dikelompokkan ke dalam <i>worksheet-worksheet</i> berdasarkan <i>email</i> pemilik jadwal, lalu sistem menampilkan menu untuk memilih lokasi penyimpanan file .xlsx yang akan diekspor ke komputer pengguna.

Tabel 5.6: Pengujian Fungsional Ekspor ke XLS

1

2 5.3 Penugjian Ekperimental

3 Pengujian eksperimtal dilakukan dengan cara mengunggah kode-kode program ke bluetape.azure
 4 yang sudah digunakan oleh mahasiswa dan staf UNPAR sehingga dosen dapat memakai fasilitas
 5 baru ini untuk didapatkan. Pengujian ini dilakukan pada semester ganjil 2017/2018. Dari hasil
 6 pemakaian, didapatkan masukan-masukan dari pengguna aplkasi ini yang menjadi basis pengujian
 7 eksperimental. Masukan-masukan ini disimpan ke dalam *issue* git proyek BlueTape yang memiliki
 8 tag "Jadwal Dosen". Semua *issue* dapat dilihat di <https://github.com/ftisunpar/BlueTape>.

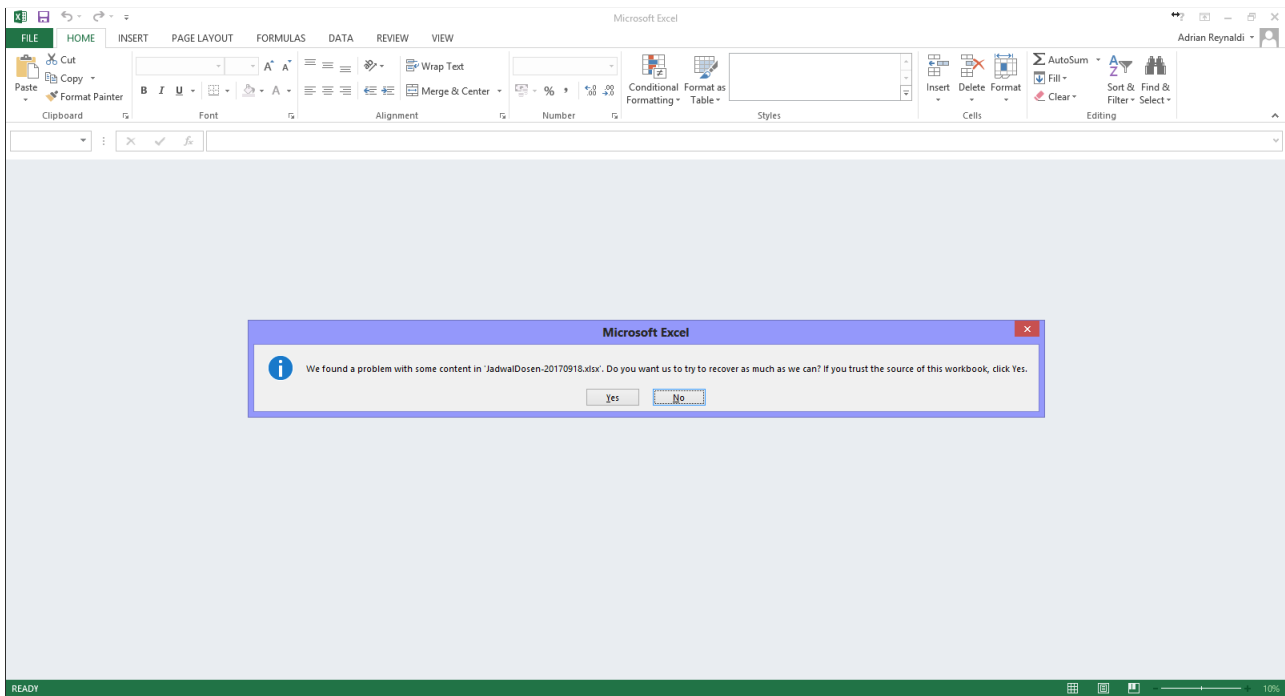
9 5.3.1 Pengujian Eksperimental Pembuatan File Spreadsheet Menggunakan Ti- 10 pe Extensi File .xlsx

11 5.3.1.1 Permasalahan

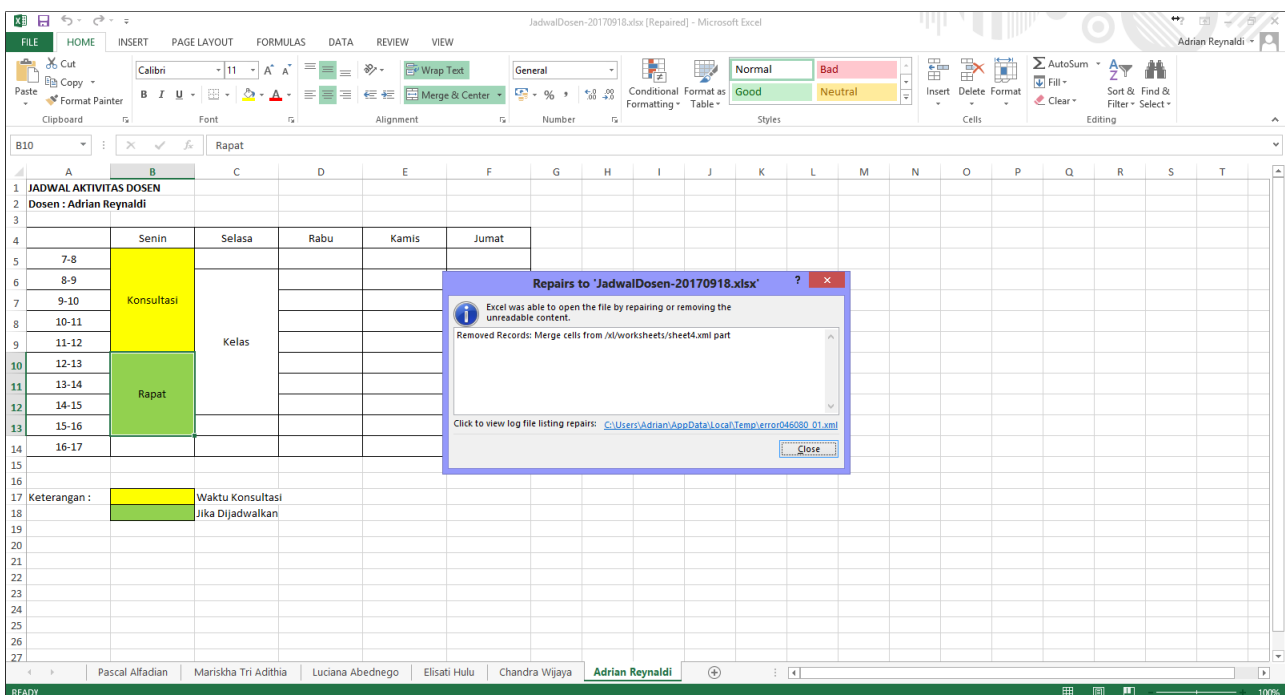
12 Pengujian ini merupakan hasil masukan dari Ibu Mariskha yang memilki masalah bahwa tab
 13 jadwal milik beliau tidak muncul setelah jadwal-jadwal diekspor ke dalam *spreadsheet*. Pengujian ini
 14 dilakukan dengan mengekspor jadwal ke dalam tipe file .xlsx yang sudah digunakan sejak Microsoft
 15 Excel 2007.

16 5.3.1.2 Analisis

17 Ketika file .xlsx tersebut dibuka menggunakan Microsoft Excel 2013, muncul eror yang dapat dilihat
 18 pada Gambar 5.1 dan Gambar 5.2.



Gambar 5.1: Tampilan error saat membuka file bertipe .xlsx



Gambar 5.2: Keterangan Error File .xlsx

Log error:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<recoveryLog xmlns="http://schemas.openxmlformats.org/spreadsheetml/2006/main">
<logFileName>error046080_01.xml</logFileName>
<summary>Errors were detected in file 'C:\Users\Adrian\Documents\Tugas\Skripsi\
  ↳ XLS Testing\JadwalDosen-20170918.xlsx'</summary>
```

```

1 <removedRecords summary="Following is a list of removed records:"><removedRecord>
2   ↳ Removed Records: Merge cells from /xl/worksheets/sheet4.xml part
3 </removedRecord></removedRecords></recoveryLog>
4

```

5.3.1.3 Hasil Pengujian dan Penyelesaian Masalah

Dilihat dari log eror pada bagian Analisis di atas, eror terjadi karena ada perintah *merge cells* yang dihapus oleh MS Excel. Hal ini terjadi karena adanya perintah dari PHPExcel versi 1.8.0 yang belum mendukung file bertipe .xlsx sehingga tab jadwal milik ibu Mariskha tidak muncul. Maka karena eror disebabkan oleh permasalahan kompatibilitas PHPExcel 1.8.0 dan tipe file .xlsx, hal ini dapat diatasi dengan cara mengubah tipe file yang diekspor dari file .xlsx menjadi tipe yang lebih lama yaitu .xls.

Potongan kode lama yang menyebabkan eror:

```

14 $filename = 'JadwalDosen-'.date("Ymd").'.xlsx'; //Nama file XLS yang akan dibuat
15 header('Content-type: application/vnd.ms-excel');
16 header('Content-Disposition: attachment;filename="' . $filename . '"');
17
18 $objWriter = PHPExcel_IOFactory::createWriter($this->excel, 'Excel2007');
19

```

Listing 5.1: Kode Lama

Potongan kode baru untuk menghilangkan eror:

```

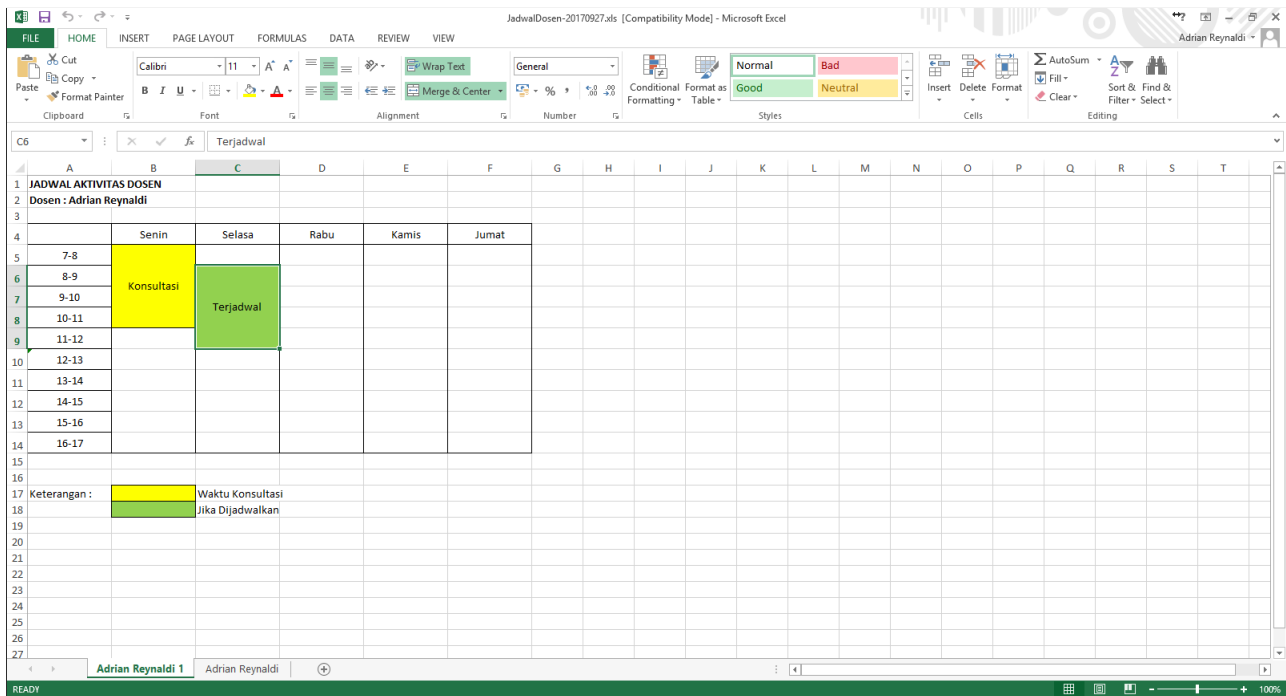
22 $filename = 'JadwalDosen-'.date("Ymd").'.xls'; //Nama file XLS yang akan dibuat
23 header('Content-type: application/vnd.ms-excel');
24 header('Content-Disposition: attachment;filename="' . $filename . '"');
25
26 $objWriter = PHPExcel_IOFactory::createWriter($this->excel, 'Excel5');
27

```

Listing 5.2: Kode Baru

Seperti bisa dilihat pada dua potongan kode program di atas, terjadi perubahan pada baris pertama kode dan baris terakhir kode. Pada baris pertama ekstensi file diubah dari .xlsx menjadi .xls. Pada baris terakhir tipe spreadsheet diubah dari 'Excel2007' menjadi 'Excel5'.

Setelah tipe file diubah menjadi .xls, eror di Microsoft Excel pun hilang.



Gambar 5.3: Eror Sudah Tidak Muncul Ketika File Dibuka

5.3.2 Pengujian Eksperimental Penambahan Tombol *Delete All*

5.3.2.1 Masalah

Ketika jadwal yang dimasukan sudah sangat banyak, maka akan sulit untuk membersihkan tabel jadwal bila pengguna harus menghapus jadwal-jadwal tersebut satu per satu. Oleh karena itu Bapak Pascal mengusulkan untuk mengimplementasikan tombol "Delete All" yang berfungsi untuk menghapus semua jadwal yang sudah dimasukan oleh pengguna.

5.3.2.2 Analisis

Karena tombol "Delete All" ini memiliki pengaruh yang besar bila tidak sengaja tertekan oleh pengguna, maka diperlukan konfirmasi sebelum perintah penghapusan jadwal dieksekusi sistem. Untuk menangani itu diperlukan cara kerja sistem sebagai berikut:

1. Ketika tombol "Delete All" ditekan akan ditampilkan *pop-up* konfirmasi penghapusan jadwal.
2. Lalu ketika tombol "ok" ditekan akan mengeksekusi perintah untuk menghapus semua jadwal milik user.
3. Jika pengguna menekan tombol "cancel", sistem akan membatalkan perintah penghapusan jadwal dan menutup *pop-up* konfirmasi.

Untuk mengimplementasikan tombol "Delete All" diperlukan method tambahan pada controller `EntriJadwalDosen` dan `JadwalDosen_model` untuk menghapus semua jadwal berdasarkan `username/email` pengguna..

Nama Method	deleteAll
Parameter Input	email via POST
Parameter Output	-
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk memanggil method pada model JadwalDosen_model untuk menghapus semua jadwal milik pengguna
Algoritma	<ul style="list-style-type: none"> • proses menerima input berupa email pengguna. • proses memanggil method deleteByUsername(email) dari kelas JadwalDosen_model dengan parameter email diisi dengan nama email pengguna.

Tabel 5.7: Perancangan method deleteAll

Nama Method	deleteByUsername
Parameter Input	email via POST
Parameter Output	-
Tabel yang berhubungan	jadwal_dosen
Deskripsi	Proses untuk menghapus semua jadwal milik pengguna di basis data
Algoritma	<ul style="list-style-type: none"> • proses menerima input berupa email dari controller EntriJadwalDosen. • proses menghapus semua jadwal pengguna dari database.

Tabel 5.8: Perancangan method deleteByUsername

1 5.3.2.3 Hasil Pengujian dan Penyelesaian Masalah

- 2 Tombol "Delete All" dibuat di halaman EntriJadwalDosen di pojok kiri bawah.

Tambah Jadwal

Hari
Senin

Durasi
1 jam

Label

Jam Mulai
7:00

Jenis
Konsultasi

Tambah

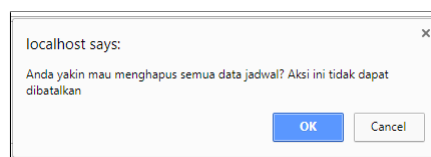
Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8					
8-9					
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					

Delete All

Gambar 5.4: Tombol *Delete All* di Entri Jadwal Dosen

- 1 Ketika tombol "Delete All" ditekan memunculkan tampilan konfirmasi

Gambar 5.5: Tombol *Delete All* di Entri Jadwal Dosen

- 2 Setelah tombol "ok" ditekan, semua jadwal dihapus

Tambah Jadwal

Hari

Senin

Durasi

1 jam

Label

Jam Mulai

7:00

Jenis

Konsultasi

Tambah

Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8					
8-9					
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					

Terakhir diupdate pada : Belum ada jadwal
Delete All

Gambar 5.6: Semua Jadwal Dihapus

Hasil pengujian berjalan lancar dan sesuai dengan hasil yang diharapkan.

5.3.3 Pengujian Eksperimental Penambahan Informasi Waktu *Update* Terakhir Jadwal

5.3.3.1 Masalah

Fitur ini merupakan usulan dari Bapak Pascal karena sering kali pengguna lupa kapan terakhir kali pengguna memasukan atau mengubah jadwal. Hal ini menyulitkan pengguna untuk mengetahui apakah jadwal yang telah ia masukan dapat dipakai atau tidak.

5.3.3.2 Analisis

Informasi waktu *update* terakhir jadwal oleh pengguna berfungsi agar pengguna dapat membedakan apakah jadwal miliknya merupakan versi lama yang sudah tidak dipakai atau merupakan versi baru. Hasil yang diharapkan adalah muncul label di halaman EntriJadwalDosen dan LihatJadwalDosen.

Untuk mengimplementasikan fitur ini perlu ditambahkan field baru di tabel jadwal_dosen:

Atribut	Tipe Data	Ukuran	PK* / FK*	Keterangan
lastUpdate	datetime	-	bukan PK/FK	tanggal terakhir jadwal di- <i>update</i>

Tabel 5.9: Perancangan field tambahan di jadwal_dosen

5.3.3.3 Hasil Pengujian dan Penyelesaian Masalah

- 2 Label yang bertuliskan informasi mengenai tanggal terakhir *update* jadwal muncul di halaman
- 3 EntriJadwalDosen

Tambah Jadwal

Hari: Durasi: Label:

Jam Mulai: Jenis:

Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8					
8-9					
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					

Terakhir diupdate pada : 30 Oktober 2017

Gambar 5.7: Informasi Waktu *Update* Terakhir Jadwal di Entri Jadwal Dosen

- 4 Label berisi tanggal terakhir update jadwal muncul di halaman LihatJadwalDosen

Adrian Reynaldi

	Senin	Selasa	Rabu	Kamis	Jumat
7-8					
8-9					
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17					

Terakhir diupdate pada : 30 Oktober 2017

Gambar 5.8: Informasi Waktu *Update* Terakhir Jadwal di Lihat Jadwal Dosen

- 5 Hasil pengujian berjalan lancar dan sesuai dengan hasil yang diharapkan.

5.3.4 Pengujian Eksperimental *Conflict Handler*

5.3.4.1 Masalah

Terdapat masalah bila ketika pengguna memasukan atau mengupdate jadwal, sistem tidak memeriksa apakah sudah ada jadwal pada waktu yang dimasukan oleh pengguna. Masalah ini merupakan masalah yang ditemukan oleh penulis sendiri.

5.3.4.2 Analisis

Untuk mencegah pengguna memasukan jadwal yang akan menimbulkan konflik dengan jadwal yang sudah ada, maka sistem perlu memeriksa:

- Memeriksa apakah sudah ada jadwal di jam mulai jadwal baru.
- Memeriksa apakah sudah ada jadwal di jam akhir jadwal baru.

Untuk mencegah pengguna juga kebingungan ketika sistem menolak memasukan jadwal karena terjadinya konflik, maka perlu adanya:

- Tampilan eror bahwa jadwal baru gagal dimasukan karena terjadi bentrok dengan jadwal lama.
- Tampilan eror bahwa edit jadwal gagal dilakukan karena terjadi bentrok dengan jadwal lain.

5.3.4.3 Hasil Pengujian dan Penyelesaian Masalah

Terdapat dua pengujian, yaitu pengujian pertama untuk memeriksa apakah konflik ketika memasukan jadwal baru sudah tertangani. Lalu pengujian kedua untuk memeriksa apakah konflik ketika mengubah jadwal yang menyebabkan bentrok dengan jadwal lain sudah ditangani.

1. Pengujian Pertama

Mencoba memasukan jadwal di hari Selasa jam 7 pagi ketika sudah ada jadwal lain di jam dan hari tersebut.

Tambah Jadwal

Hari: Selasa, Durasi: 1 jam, Label:

Jam Mulai: 7:00, Jenis: Konsultasi,

Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8	asd				
8-9		asd			
9-10					
10-11					
11-12					

Gambar 5.9: Memasukan Jadwal ke Waktu yang Sudah Ada Jadwal

- 1 Muncul tampilan eror yang memberi tahu pengguna bahwa jadwal yang ingin dimasukan
- 2 gagal masuk karena terjadi bentrok.

Jadwal gagal dimasukan karena sudah ada jadwal pada waktu tersebut, silahkan pilih waktu lain.

Tambah Jadwal

Hari: Selasa, Durasi: 1 jam, Label:

Jam Mulai: 7:00, Jenis: Konsultasi,

Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8	asd				
8-9		asd			
9-10					
10-11					
11-12					

Gambar 5.10: Tampilan Error Gagal Masuk

- 3 2. **Pengujian Kedua** Daftar jadwal yang sudah ada dapat dilihat di Gambar 5.11

Daftar Jadwal					
	Senin	Selasa	Rabu	Kamis	Jumat
7-8	asd	1			
8-9		Tengah			
9-10					
10-11					
11-12	asd	2	asd		
12-13					
13-14					
14-15					
15-16					
16-17					

Gambar 5.11: Jadwal yang Sudah Ada

- 1 Mencoba mengubah jadwal "1" menjadi berdurasi 3 jam sehingga memicu bentrok dengan
- 2 jadwal "Tengah".

Edit Jadwal

Hari

Selasa

Jam Mulai

7:00

Durasi

3 jam

Jenis

Konsultasi

Label

1

Save

Delete

Gambar 5.12: Mengubah Jadwal "1"

- 3 Muncul tampilan eror yang memberi tahu bahwa jadwal gagal diubah.

Jadwal gagal di-update karena menimbulkan konflik dengan jadwal lain.

Manajemen Cetak Transkrip	Perubahan Kuliah	Manajemen Perubahan Kuliah	Entri Jadwal Dosen	Lihat Jadwal Dosen
---------------------------	------------------	----------------------------	--------------------	--------------------

Tambah Jadwal

Hari: Durasi: Label:

Jam Mulai: Jenis:

Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8	asd	1			
8-9		Tengah			
9-10					
10-11					
11-12		2			

Gambar 5.13: Tampilan Error Gagal *Update*

- 1 Mencoba mengubah jadwal "2" menjadi mulai pada jam 10 sehingga memicu bentrok dengan
- 2 jadwal "Tengah".

Edit Jadwal

Hari:

Jam Mulai:

Durasi:

Jenis:

Label:

Gambar 5.14: Mengubah Jadwal "2"

- 3 Muncul tampilan error yang memberi tahu bahwa jadwal gagal diubah.

Jadwal gagal di-update karena menimbulkan konflik dengan jadwal lain.

Transkrip
Manajemen Cetak Transkrip
Perubahan Kuliah
Manajemen Perubahan Kuliah
Entri Jadwal Dosen
Lihat Jadwal Dosen

Tambah Jadwal

Hari

Senin

Durasi

1 jam

Label

Jam Mulai

7:00

Jenis

Konsultasi

Tambah

Daftar Jadwal

	Senin	Selasa	Rabu	Kamis	Jumat
7-8	asd	1			
8-9		Tengah			
9-10					
10-11					
11-12		2			

Gambar 5.15: Tampilan Error Gagal *Update*

- 1 Setelah kedua pengujian di atas maka dapat disimpulkan pengujian berjalan dengan lancar dan
- 2 hasil yang diharapkan tercapai.

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Penelitian yang dilakukan dalam pengembangan Aplikasi Pembangkit Jadwal Dosen berhasil memenuhi harapan dalam mencatat dan menampilkan jadwal dosen. Kesimpulan yang dapat diambil dari hasil penelitian adalah sebagai berikut:

1. Aplikasi memenuhi tujuan dalam mengotentikasi pengguna yang mengakses BlueTape. Pengguna dosen dapat mengakses modul Entri Jadwal Dosen dan Lihat Jadwal Dosen sedangkan pengguna mahasiswa hanya dapat mengakses modul Lihat Jadwal Dosen.
2. Aplikasi memenuhi tujuan menyediakan cara bagi dosen untuk memasukan jadwalnya ke dalam BlueTape dengan mengimplementasikan modul Entri Jadwal Dosen yang berisi menu untuk menambah jadwal. Selain itu, aplikasi ini juga memenuhi tujuan agar jadwal dosen dapat ditampilkan di BlueTape dengan mengimplementasikan modul Lihat Jadwal Dosen yang menampilkan setiap jadwal dosen dalam bentuk tabel-tabel.
3. Aplikasi memenuhi tujuan untuk mengekspor jadwal yang disimpan di basis data menjadi tipe file xls.

6.2 Saran

Berdasarkan hasil kesimpulan di atas, maka berikut merupakan saran-saran yang dapat diberikan untuk pengembangan selanjutnya:

1. Ditemukan bahwa format email untuk mahasiswa angkatan 2017 memiliki format yang berbeda dengan mahasiswa-mahasiswa angkatan sebelumnya. Mahasiswa angkatan 2017 memiliki format xxxx73yyyy@student.unpar.ac.id xxxx merupakan angkatan dan yyyy merupakan nomor pokoknya. Sedangkan sebelumnya email mahasiswa memiliki format 73xxyyyy@student.unpar.ac.id dengan xx adalah angkatan dan yyyy nomor pokoknya. Perlu dikaji ulang cara otentikasi dan pengelompokan pengguna mahasiswa pada penerapan Google OAuth.
2. Mengurutkan tab-tab jadwal dosen berdasarkan alphabet atau urutan lainnya agar nama dosen mudah untuk dicari.
3. Mengganti library PHPExcel dengan versi terbarunya yaitu PHPOffice untuk mendukung pembuatan file bertipe .xlsx agar ukuran file lebih kecil.

DAFTAR REFERENSI

- [1] Codeigniter 3 (2017) *Codeigniter 3.1.4 Documentation*. British Columbia Institute of Technology. 5112 Bobby Hicks Hwy, Gray, TN 37615, USA.
- [2] Barth, A., Jackson, C., dan Mitchell, J. C. (2008) Robust defenses for cross-site request forgery. *To appear at the 15th ACM Conference on Computer and Communications Security (CCS 2008)*.
- [3] Johns, M. dan Winter, J. (2006) Request rodeo: Client side protection against session riding. Bagian dari Piessens, F. (ed.), *Proceedings of the OWASP Europe 2006 Conference, refereed papers track, Report CW448*, May, pp. 5 – 17. Departement Computerwetenschappen, Katholieke Universiteit Leuven.
- [4] Foundation 6 (2017) *Foundation v6.3.1 Documentation*. Zurb Inc. 100 W Rincon Ave, Campbell, CA 95008, USA.
- [5] Version 1.8.0 (2014) *PHPExcel Formula Function Reference Developer Documentation*. PHPOffice.

LAMPIRAN A

KODE PROGRAM IMPLEMENTASI MODUL ENTRI JADWAL DOSEN

Kode Program untuk *controller* modul Entri Jadwal Dosen

Listing A.1: EntriJadwalDosen.php

```
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class EntriJadwalDosen extends CI_Controller {
6
7     public function __construct() {
8         parent::__construct();
9         try {
10             $this->Auth_model->checkModuleAllowed(get_class());
11         } catch (Exception $ex) {
12             $this->session->set_flashdata('error', $ex->getMessage());
13             header('Location:./');
14         }
15         $this->load->library('bluetape');
16         $this->load->model('JadwalDosen_model');
17         $this->load->model('Auth_model');
18         $this->load->database();
19     }
20
21     //method yang dipanggil ketika halaman EntriJadwalDosen dibuka
22     public function index() {
23         // Retrieve logged in user data
24         $userInfo = $this->Auth_model->getUserInfo();
25         $dataJadwal = $this->JadwalDosen_model->getJadwalByUsername($userInfo['email']);
26         $namaHari = $this->JadwalDosen_model->getNamaHari();
27         $namaBulan = $this->JadwalDosen_model->getNamaBulan();
28         $this->load->view('EntriJadwalDosen/main', array(
29             'currentModule' => get_class(),
30             'request_add_jadwal' => $this->session->userdata('request_add_jadwal'),
31             'dataJadwal' => $dataJadwal,
32             'namaHari' => $namaHari,
33             'namaBulan' => $namaBulan
34         ));
35     }
36
37     //fungsi untuk menambah jadwal ke dalam database dari input user
38     public function add() {
39         $userInfo = $this->Auth_model->getUserInfo();
40
41         $jam_mulai = $this->input->post('jam_mulai');
42         $durasi = $this->input->post('durasi');
43         $jam_akhir = $jam_mulai + $durasi;
44         $hari = $this->input->post('hari');
45         $bisaMasuk = TRUE;
46         for($i=7; $i<17; $i++){
47             //memeriksa apakah ada jadwal lain di antara jam mulai dan jam akhir pada jadwal yang dimasukan oleh user
48             $exist=$this->JadwalDosen_model->cekJadwalByJamMulai($i,$hari,$userInfo['email']);
49             if($exist!=null){ //ada potensi jadwal bentrok
50                 $existJamAkhir = $exist[0]->jam_mulai+$exist[0]->durasi; //mendapatkan jam akhir dari jadwal yang berpotensi bentrok
51                 if(($exist[0]->jam_mulai<=$jam_mulai && $existJamAkhir>$jam_mulai) || ($exist[0]->jam_mulai<$jam_akhir && $existJamAkhir>=$jam_akhir)){
52                     $bisaMasuk = FALSE;
53                     break;
54                 }
55             }
56         }
57         if($bisaMasuk){
58             $data = array(
59                 'user' => $userInfo['email'],
60                 'hari' => $this->input->post('hari'),
61                 'jam_mulai' => $this->input->post('jam_mulai'),
62                 'durasi' => $this->input->post('durasi'),
63                 'jenis_jadwal' => $this->input->post('jenis_jadwal'),
64                 'label_jadwal' => $this->input->post('label_jadwal')
65             );
66             $this->JadwalDosen_model->addJadwal($data);
67             header('Location:./EntriJadwalDosen');
68         }
69     }
70 }
```

```

70         $this->session->set_flashdata('error', 'Jadwal_gagal_dimasukan_karena_sudah_ada_jadwal_pada_waktu_tersebut,_silahkan_
           pilih_waktu_lain. ');
71         header('Location:./EntriJadwalDosen');
72     }
73 }
74
75 //fungsi untuk mengupdate jadwal di dalam database dari input user
76 public function update($id_jadwal) {
77     $userInfo = $this->Auth_model->getUserInfo();
78
79     $jam_mulai = $this->input->post('jam_mulai');
80     $durasi = $this->input->post('durasi');
81     $jam_akhir = $jam_mulai + $durasi;
82     $hari = $this->input->post('hari');
83     $bisaMasuk = TRUE;
84     for($i=7 ; $i<17 ; $i++){
85         //memeriksa apakah ada jadwal lain di antara jam mulai dan jam akhir pada jadwal yang dimasukan oleh user
86         $exist=$this->JadwalDosen_model->cekJadwalByJamMulai($i,$hari,$userInfo['email']);
87         if($exist!=null){ //ada potensi jadwal bentrok
88             $existJamAkhir = $exist[0]->jam_mulai+$exist[0]->durasi; //mendapatkan jam akhir dari jadwal yang berpotensi
              bentrok
89             if( ($exist[0]->jam_mulai<=$jam_mulai && $existJamAkhir>$jam_mulai) || ($exist[0]->jam_mulai<$jam_akhir &&
              $existJamAkhir>=$jam_akhir)){
90                 if(strcmp($exist[0]->id,$id_jadwal)){ //memeriksa apakah jadwal tersebut merupakan dirinya sendiri atau bukan
91                     $bisaMasuk = FALSE;
92                     break;
93                 }
94             }
95         }
96     }
97     if($bisaMasuk){
98         $data = array(
99             'hari' => $this->input->post('hari'),
100             'jam_mulai' => $this->input->post('jam_mulai'),
101             'durasi' => $this->input->post('durasi'),
102             'jenis' => $this->input->post('jenis_jadwal'),
103             'label' => $this->input->post('label_jadwal'),
104             'lastUpdate' => date('Y-m-d_H:i:s')
105         );
106         $this->JadwalDosen_model->updateJadwal($id_jadwal, $data);
107         header('Location:./EntriJadwalDosen');
108     }
109     else{
110         $this->session->set_flashdata('error', 'Jadwal_gagal_di-update_karena_menimbulkan_konflik_dengan_jadwal_lain. ');
111         header('Location:./EntriJadwalDosen');
112     }
113 }
114
115 //menghapus jadwal berdasarkan id jadwal yang dipilih user
116 public function delete($id_jadwal) {
117     $this->JadwalDosen_model->deleteJadwal($id_jadwal);
118     header('Location:./EntriJadwalDosen');
119 }
120
121 //menghapus semua jadwal milik user
122 public function deleteAll(){
123     $userInfo = $this->Auth_model->getUserInfo();
124     $this->JadwalDosen_model->deleteByUsername($userInfo['email']);
125     header('Location:./EntriJadwalDosen');
126 }
127
128 //mendapatkan data-data milik user
129 public function getDataJadwal($id_jadwal) {
130     echo $this->JadwalDosen_model->getJadwalByIdJadwal($id_jadwal);
131 }
132 }

```

Kode Program untuk *view* modul Entri Jadwal Dosen

Listing A.2: main.php

```

1 <?php
2 defined('BASEPATH') OR exit('No_direct_script_access_allowed');
3 ?><!doctype html>
4 <html class="no-js" lang="en">
5 <?php $this->load->view('templates/script_foundation'); ?>
6 <?php $this->load->view('templates/head_loggedin'); ?>
7 <?php $this->load->view('templates/flashmessage'); ?>
8 <?php $this->load->helper('url'); ?>
9 <body>
10 <?php $this->load->view('templates/topbar_loggedin'); ?>
11
12 <div class="row">
13
14     <div class="large-12_column_callout">
15         <h5>Tambah Jadwal</h5>
16         <div class="large-4_columns">
17             <form method="POST" action="/EntriJadwalDosen/add">
18                 <input type="hidden" name="<?=$_this->security->get_csrf_token_name()_?>" value="<?=$_this->security->
                  get_csrf_hash()_?>" />
19                 Hari
20                 <select name="hari">
21                     <?php
22                         $hariValue = 0;
23                     <foreach ($namaHari as $hari) {
24                         <?php
25                             <option value="<?=$_this->security->get_csrf_token_name()_?>" <?=$_this->security->
26                                 get_csrf_hash()_?>" <?=$_this->security->get_csrf_token_name()_?>" value="<?=$_this->security->

```

```

27         <?php
28             $hariValue++;
29         }
30         ?>
31     </select><br>
32     Jam Mulai
33     <select name="jam_mulai">
34         <?php for ($i = 7; $i <= 16; $i++) { ?>
35             <option value="<?php echo $i ?>"> <?php echo $i ?>:00 </option>
36         <?php } ?>
37     </select><br>
38 </div>
39 <div class="_large-4_columns">
40     Durasi
41     <select name="durasi">
42         <?php for ($i = 1; $i <= 9; $i++) { ?>
43             <option value="<?php echo $i ?>"> <?php echo $i ?> jam </option>
44         <?php } ?>
45     </select><br>
46     Jenis
47     <select name="jenis_jadwal">
48         <option value="konsultasi" style="background-color:yellow"> Konsultasi </option>
49         <option value="terjadwal" style="background-color:green;color:white"> Terjadwal</option>
50         <option value="kelas" style="background-color:white"> Kelas </option>
51     </select>
52 </div>
53 <div class="large-4_columns">
54     Label <input type="text" name="label_jadwal"><br>
55     <input type="submit" class="button" value="Tambah">
56 </form>
57 </div>
58 </div>
59 <!-- ===== Pembentukan Tabel ===== -->
60
61 <div class="large-12_column_callout">
62     <h5>Daftar Jadwal</h5>
63     <div class="table-scroll" id="jadwal_table">
64         <table border=1 style="border-color:black;_border-collapse:separate">
65             <tr>
66                 <td style='width:10%'></td>
67                 <?php
68                 for ($i = 0; $i < 5; $i++) {
69                     echo "<td style='width:18%'>_namaHari[$i]_</td>"; //Membuat Header Tabel yang berisi daftar
69                         hari
69                 }
70                 ?>
71             </tr>
72             <?php
73             //GENERATE BODY UTAMA TABEL
74             $cellRowID = 1;
75             for ($i = 7; $i < 17; $i++) {
76                 echo "<tr><td>" . $i . "-" . ($i + 1);
77                 $cellColID = 1;
78                 for ($j = 0; $j < 5; $j++) {
79                     echo"<td align='center' _id='cell' . $cellRowID . '-' . $cellColID . '>' . '</td>";
80                     $cellColID++;
81                 }
82                 $cellRowID++;
83             }
84             // MEWARNAI TABEL
85             $rowIdx = 1;
86             $spanCounter = [0, 0, 0, 0, 0, 0, 0, 0, 0];
87             $cellRowID = 0;
88             foreach ($dataJadwal as $dataHariIni) {
89                 $colIdx = $dataHariIni->hari + 1; // + 1 karena perbedaan selisih index tabel dan value hari di
90                     database
91                 $rowIdx = $dataHariIni->jam_mulai - 6; // + 1 karena perbedaan selisih index tabel dan value
92                     jam_mulai di database
93                 if ($dataHariIni->jenis == "konsultasi") {
94                     $color = "#FFFF00";
95                 } else if ($dataHariIni->jenis == "kelas") {
96                     $color = "#FFFFFF";
97                 } else {
98                     $color = "#92D14F";
99                 }
100                 ?>
101                 <script type="text/javascript">
102                     var table = document.getElementById('jadwal_table');
103                     var rows = table.getElementsByTagName('tr');
104                     var $cellLocation = "#cell<?php echo $rowIdx;_?>-<?php echo $colIdx;_?>";
105
106                     $($cellLocation).css('background-color', '<?php echo $color;_?>');
107                     $($cellLocation).attr('rowspan', '<?php echo $dataHariIni->durasi ?>');
108
109                     //menghapus cell-cell yang tergeser karena rowspan
110                     for (i = <?php echo ($rowIdx + 1); ?>; i < <?php echo ($rowIdx + $dataHariIni->durasi); ?>; i
111                         ++){
112                         $("#cell" + i + "-" +<?php echo $colIdx; ?>).remove();
113                     }
114                     $($cellLocation).html("<?php echo $dataHariIni->label_?>");
115
116                     //membuat cell-cell yang telah diwarnai jadi memunculkan pop-up untuk mengedit jadwal ketika
117                     diklik oleh mouse
118                     $(document).on("click", $cellLocation, function () {
119                         var $menuName = "#edit_menu<?php echo $dataHariIni->id_?>";
120                         $($menuName).foundation('open');
121                     });
122                 </script>

```

```

120         <?php
121         }
122         $cellRowID++;
123         $rowIdx++;
124         ?>
125     </table>
126 </div>
127 <?php
128 if($dataJadwal!=null){
129     $palingBaru=$dataJadwal[0]->lastUpdate;
130     foreach($dataJadwal as $perJadwal){
131         if($palingBaru<$perJadwal->lastUpdate){
132             $palingBaru=$perJadwal->lastUpdate;
133         }
134     }
135     $tgl = date('d',strtotime($palingBaru));
136     $tgl = $tgl.'.'. $namaBulan[date('m',strtotime($palingBaru)) - 1]; // minus 1 karena index dimulai dari 0
137     $tgl = $tgl.'.'.date('Y',strtotime($palingBaru));
138     ?>
139     Terakhir diupdate pada : <?=$tgl?><br>
140 <?php
141 }
142 else{
143     $palingBaru=FALSE;
144     ?>
145     Terakhir diupdate pada : Belum ada jadwal <br>
146 <?php
147 }
148 ?>
149 <a href="/EntriJadwalDosen/deleteAll/export/" class="alert_button" onClick="return_konfirmasi();">Delete All</a>
150 <script>
151     function konfirmasi()
152     {
153         yakin = confirm("Anda yakin mau menghapus semua data jadwal? Aksi ini tidak dapat dibatalkan");
154         if(yakin!=true)
155         {
156             return false;
157         }
158     }
159 </script>
160 </div>
161
162 <!-- ===== END Pembentukan Tabel ===== -->
163
164 <!-- ===== MENU EDIT JADWAL ===== -->
165 <?php foreach ($dataJadwal as $dataHariIni) { ?>
166     <div id="edit_menu<?php_echo_$dataHariIni->id_?>" class="reveal" data-reveal >
167         <button class="close-button" data-close aria-label="Close_modal" type="button">
168             <span aria-hidden="true">&times;</span>
169         </button>
170         <h5> Edit Jadwal </h5>
171         <form name="form<?php_echo_$dataHariIni->id_?>" method="POST" action="/EntriJadwalDosen/update/<?php_echo_$dataHariIni->id_?>">
172             <input type="hidden" name="<?=$_this->security->get_csrf_token_name()_?>" value="<?=$_this->security->get_csrf_hash()_?>" />
173             <input type="hidden" name="id_jadwal_parameter" value="<?php_echo_$dataHariIni->id_?>"> </a> <br>
174             Hari
175             <select name="hari">
176                 <?php
177                 $hariValue = 0;
178                 foreach ($namaHari as $hari) {
179                     if ($hariValue == $dataHariIni->hari) {
180                         ?>
181                         <option value="<?=$_$hariValue_?>" selected> <?=$hari ?> </option>
182                         <?php
183                     } else {
184                         ?>
185                         <option value="<?=$_$hariValue_?>" <?=$hari ?> </option>
186                         <?php
187                     }
188                     $hariValue++;
189                 }
190                 ?>
191             </select><br>
192             Jam Mulai
193             <select name="jam_mulai">
194                 <?php
195                 for ($i = 7; $i <= 16; $i++) {
196                     if ($i == $dataHariIni->jam_mulai) {
197                         ?>
198                         <option value="<?php_echo_$i_?>" selected> <?php echo $i ?>:00 </option>
199                         <?php
200                     } else {
201                         ?>
202                         <option value="<?php_echo_$i_?>" <?php echo $i ?>:00 </option>
203                         <?php
204                     }
205                 }
206                 ?>
207             </select><br>
208             Durasi
209             <select name="durasi">
210                 <?php
211                 for ($i = 1; $i <= 9; $i++) {
212                     if ($i == $dataHariIni->durasi) {
213                         ?>

```



```

214         <option value="<?php_echo_$i_?>" selected> <?php echo $i ?> jam </option>
215     <?php
216     } else {
217         ?>
218         <option value="<?php_echo_$i_?>"> <?php echo $i ?> jam </option>
219         <?php
220     }
221 }
222 ?>
223 </select><br>
224 Jenis
225 <select name="jenis_jadwal">
226     <option style="background-color:yellow" value="konsultasi" <?php if ($dataHariIni->jenis == '
227         konsultasi') echo "selected"; ?> > Konsultasi </option>
228     <option style="background-color:green" value="terjadwal" <?php if ($dataHariIni->jenis == '
229         terjadwal') echo "selected"; ?> > Terjadwal</option>
230     <option style="background-color:white" value="kelas" <?php if ($dataHariIni->jenis == 'kelas')
231         echo "selected"; ?> > Kelas </option>
232 </select>
233 Label <input type="text" name="label_jadwal" value="<?php_echo_$dataHariIni->label;_?>"><br>
234 <div class="row_large-4_column">
235     <div class="large-2_column">
236         <input type="submit" name="submitId<?php_echo_$dataHariIni->id_?>" class="button" value="Save_?>"
237     </div>
238     <div class="large-2_column">
239         <form name="formDelete<?php_echo_$dataHariIni->id_?>" method="POST" action="/EntriJadwalDosen/
240             delete/<?php_echo_$dataHariIni->id_?>"
241             <input type="hidden" name="<?=$_$this->security->get_csrf_token_name()_?>" value="<?=$_$this->
242                 security->get_csrf_hash()_?>" />
243             <input type="submit" id="deletebtn<?php_echo_$dataHariIni->id_?>" name="deletebtn<?php_echo_
244                 $dataHariIni->id_?>" class="alert_button" value="Delete">
245         </form><div>
246     </div>
247 </div>
248 </div>
249 <?php } ?>
250 </div>
<!-- =====END MENU EDIT JADWAL===== -->
<?php $this->load->view('templates/script_foundation'); ?>
</body>
</html>

```


LAMPIRAN B

KODE PROGRAM IMPLEMENTASI MODUL LIHAT JADWAL DOSEN

Kode Program untuk *controller* modul Lihat Jadwal Dosen

Listing B.1: LihatJadwalDosen.php

```

1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class LihatJadwalDosen extends CI_Controller {
6
7     private $excel;
8     public function __construct() {
9         parent::__construct();
10        try {
11            $this->Auth_model->checkModuleAllowed(get_class());
12        } catch (Exception $ex) {
13            $this->session->set_flashdata('error', $ex->getMessage());
14            header('Location:./');
15        }
16        $this->excel = new PHPEXcel();
17        $this->load->library('bluetape');
18        $this->load->library('session');
19        $this->load->model('JadwalDosen_model');
20        $this->load->database();
21    }
22
23    //method yang dipanggil ketika halaman LihatJadwalDosen dibuka
24    public function index() {
25        // Retrieve logged in user data
26        $userInfo = $this->Auth_model->getUserInfo();
27
28        $dataJadwal = $this->JadwalDosen_model->getAllJadwal();
29        $dataJadwalPerUser = array();
30        foreach ($dataJadwal as $key => $indexValue) {
31            $dataJadwalPerUser[$indexValue->user][$key] = $indexValue; // dimensi pertama indexnya adalah user
32        }
33        ksort($dataJadwalPerUser);
34        $this->session->set_userdata( 'dataJadwalPerUser', $dataJadwalPerUser );
35        $namaHari = $this->JadwalDosen_model->getNamaHari();
36        $namaBulan = $this->JadwalDosen_model->getNamaBulan();
37        $this->load->view('LihatJadwalDosen/main', array(
38            'currentModule' => get_class(),
39            'dataJadwalPerUser' => $dataJadwalPerUser,
40            'namaHari' => $namaHari,
41            'namaBulan' => $namaBulan
42        ));
43    }
44
45    //method untuk mengubah jadwal di database kedalam file .xls
46    public function export() {
47        $dataToExport = $this->session->userdata('dataJadwalPerUser');
48        // ----- TEMPLATE TABEL JADWAL DOSEN || TIDAK PERLU DIUBAH LAGI-----
49
50        $startHourRow = 5;
51        $endHourRow = $startHourRow + 10;
52        $startColoredCell = -2; // minus dua karena perbedaan penomoran baris di excel dengan jam paling pagi yang bisa dicatat di
53        jadwal
54        $titleRow = 1;
55        $nameRow = 2;
56        $dayRow = 4;
57        $keteranganRow = 17;
58        $nextTableRowsAdder = 19;
59
60        $idx = 0;
61        $sheetIdx = 0;
62        foreach ($dataToExport as $currRow) {
63            $objWorksheet = $this->excel->createSheet($sheetIdx);
64            $this->excel->setActiveSheetIndex(0);
65            foreach ($currRow as $oneData) {
66                $name = $oneData->name;
67                break;
68            };
69            $this->excel->getActiveSheet()->setTitle($name);
70
71            //Menulis header tabel

```

```

70 $titleCell = 'A' . $titleRow;
71 $this->excel->getActiveSheet()->setCellValue($titleCell, 'JADWAL_AKTIVITAS_DOSEN');
72 $this->excel->getActiveSheet()->getStyle($titleCell)->getFont()->setBold(true);
73 $this->excel->getActiveSheet()->setCellValue('A' . $nameRow, 'Dosen:_' . $name); // menulis nama dosen yang
    bersangkutan
74 $this->excel->getActiveSheet()->getStyle('A' . $nameRow)->getFont()->setBold(true);
75
76 //Menulis hari-hari dalam tabel jadwal dosen
77 $hari = $this->JadwalDosen_model->getNamaHari();
78 $dayCol = 'B';
79 foreach ($hari as $key) {
80     $dayCell = $dayCol . $dayRow;
81     $this->excel->getActiveSheet()->setCellValue($dayCell, $key);
82     $dayCol++;
83 }
84
85 //Menentukan style dari border
86 $borderStyleArray = array(
87     'borders' => array(
88         'allborders' => array(
89             'style' => PHPExcel_Style_Border::BORDER_THIN
90         )
91     )
92 );
93 $outlineStyle = array(
94     'borders' => array(
95         'outline' => array(
96             'style' => PHPExcel_Style_Border::BORDER_THIN
97         )
98     )
99 );
100
101 //Menulis bagian keterangan
102 $this->excel->getActiveSheet()->setCellValue('A' . $keteranganRow, 'Keterangan:');
103 $this->excel->getActiveSheet()->getStyle('B' . $keteranganRow)->getFill()->setFillType(PHPExcel_Style_Fill::FILL_SOLID
104     )->getStartColor()->setRGB('FFFF00');
105 $this->excel->getActiveSheet()->getStyle('B' . ($keteranganRow + 1))->getFill()->setFillType(PHPExcel_Style_Fill::
    FILL_SOLID)->getStartColor()->setRGB('92D14F');
106 $this->excel->getActiveSheet()->setCellValue('C' . $keteranganRow, 'Waktu_Konsultasi');
107 $this->excel->getActiveSheet()->setCellValue('C' . ($keteranganRow + 1), 'Jika_Dijadwalkan');
108
109 //Membuat border
110 $this->excel->getActiveSheet()->getStyle('A4:A14')->applyFromArray($borderStyleArray); //
    menambah outline pada body tabel
111 $this->excel->getActiveSheet()->getStyle('B4:F4')->applyFromArray($borderStyleArray); //
    menambah outline pada body tabel
112 $this->excel->getActiveSheet()->getStyle('A4:F14')->applyFromArray($outlineStyle); //menambah
    outline pada body tabel
113 $this->excel->getActiveSheet()->getStyle('B4:B14')->applyFromArray($outlineStyle); //menambah
    outline pada body tabel
114 $this->excel->getActiveSheet()->getStyle('C4:C14')->applyFromArray($outlineStyle); //menambah
    outline pada body tabel
115 $this->excel->getActiveSheet()->getStyle('D4:D14')->applyFromArray($outlineStyle); //menambah
    outline pada body tabel
116 $this->excel->getActiveSheet()->getStyle('E4:E14')->applyFromArray($outlineStyle); //menambah
    outline pada body tabel
117 $this->excel->getActiveSheet()->getStyle('B' . $keteranganRow)->applyFromArray($borderStyleArray); //
    menambah border pada kotak keterangan
118 $this->excel->getActiveSheet()->getStyle('B' . ($keteranganRow + 1))->applyFromArray($borderStyleArray); //
    menambah border pada kotak keterangan yang kedua
119 unset($borderStyleArray);
120
121 //Membuat semua tulisan dalam tabel menggunakan align center
122 $this->excel->getActiveSheet()->getStyle('A' . $dayRow . ':F' . ($dayRow + 10))->getAlignment()->setHorizontal(
    PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
123 $this->excel->getActiveSheet()->getStyle('A' . $dayRow . ':F' . ($dayRow + 10))->getAlignment()->setVertical(
    PHPExcel_Style_Alignment::VERTICAL_CENTER);
124 for ($col = ord('A'); $col <= ord('F'); $col++) {
125     //mengatur lebar setiap kolom $col
126     $this->excel->getActiveSheet()->getColumnDimension(chr($col))->setWidth(15);
127 }
128 for ($row = $dayRow; $row <= ($dayRow + 10); $row++) {
129     //mengatur tinggi setiap baris $row
130     $this->excel->getActiveSheet()->getRowDimension($row)->setRowHeight(20);
131 }
132 //-----END TEMPLATE-----
133
134 //menulis jam-jam dalam tabel jadwal dosen
135 $jam = 7;
136 for ($i = $startHourRow; $i < ($endHourRow); $i++) {
137     $column = 'A' . $i;
138     $this->excel->getActiveSheet()->setCellValue($column, ($jam) . '-' . ($jam + 1));
139     $jam++;
140 }
141
142 //mewarnai tabel sesuai jadwal yang sudah dimasukkan
143 foreach ($currRow as $dataHariIni) {
144     if ($dataHariIni != null) {
145         $colHari = $this->JadwalDosen_model->hariKeKolom($dataHariIni->hari); //index kolom hari
146         $row_jam_mulai = $dataHariIni->jam_mulai + $startColoredCell; //startColoredCell berfungsi agar
            nilai row_jam_mulai tepat berada pada index baris yang tepat (jam mulai dan index baris jam pada
            excelnya berselisih 2, maka startColoredCell bernilai "-2". Karena misal jam_mulai = 7, tetapi pada
            excel baris yang bertanda "jam 7" ada pada baris ke 5, bukan baris ke 7)
146         $row_jam_selesai = $row_jam_mulai + $dataHariIni->durasi;
147         if ($row_jam_selesai > 15) { //agar pewarnaan maupun outline
            jadwal tidak melebihi tabel yang sudah ditentukan di atas
148             $row_jam_selesai = 15;
149         }
150     }
151 }

```

Kode Program untuk *view* modul Lihat Jadwal Dosen

Listing B.2: main.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><!doctype html>
4 <html class="no-js" lang="en">
5     <?php $this->load->view('templates/head_loggedin'); ?>
6     <body>
7         <?php $this->load->view('templates/topbar_loggedin'); ?>
8         <?php $this->load->helper('url'); ?>
9         <?php $this->load->view('templates/script_foundation'); ?>
10        <?php // $this->load->view('templates/flashmessage'); ?>
11
12        <div class="row">
13            <div class="large-12_column">
14                <ul class="tabs" data-tabs id="tab_jadwal">
15                    <?php
16                        $idx = 0;
17                        foreach ($dataJadwalPerUser as $currRow) {
18                            if ($idx == 0) {
19                                <li class="tabs-title_is-active"><a href="#hal<?php_echo_$idx;_?" aria-selected="true"><?php foreach
20                                    ($currRow as $data) {
21                                        echo $data->name;
22                                        break;
23                                    } ?></a></li>
24                                <?php
25                                    } else {
26                                        <li class="tabs-title"><a href="#hal<?php_echo_$idx;_?"><?php foreach ($currRow as $data) {
27                                            echo $data->name;
28                                            break;
29                                        } ?></a></li>
30                                    }
31                                <?php
32                            }
33                        }
34                    }
35                    $idx++;
36                }
37                ?>
38            </ul>
39
40            <div class="tabs-content" data-tabs-content="tab_jadwal">
41                <?php
42                    $idx = 0;
43                    foreach ($dataJadwalPerUser as $currRow) {
44                        <?php
45                            <div class="table-scroll" id="jadwal_table<?php_echo_$idx;_?">
46                                <?php
47                                    if ($idx == 0) {
48                                        echo '<div class="tabs-panel_is-active" id="hal' . $idx . "'>';
49                                    }
50                                <?php

```

```

53     } else {
54         ?>
55         <div class="tabs-panel" id="hal<?php_echo_$idx;_?>">
56             <?php
57             }
58             ?>
59             <table id="tabel<?php_echo_$idx;_?>" border=1 style="border-color:black;_border-collapse:separate
60                 ">
61                 <!-- ----- GENERATE TEMPLATE TABEL -----
62                 ----- -->
63                 <tr>
64                     <td style='width:10%'>
65                         </td>
66                         <?php
67                         for ($i = 0; $i < 5; $i++) {
68                             echo "<td_style='width:18%'>" . $namaHari[$i] . "</td>";
69                         }
70                         ?>
71                     </tr>
72                     <?php
73                     $cellRowID = 1;
74                     for ($i = 7; $i < 17; $i++) {
75                         echo "<tr><td>" . $i . "-" . ($i + 1);
76                         $cellColID = 1;
77                         for ($j = 0; $j < 5; $j++) {
78                             echo "<td_align='center'_id='t" . $idx . "cell" . $cellRowID . "-" . $cellColID . "'>"
79                             . "</td>";
80                             $cellColID++;
81                         }
82                         $cellRowID++;
83                     }
84                 }
85                 // MEWARNAI TABEL
86                 $rowIdx = 1;
87                 $cellRowID = 0;
88                 // -----
89                 foreach ($scurrRow as $dataHariIni) {
90                     $colIdx = $dataHariIni->hari + 1; // + 1 karena perbedaan selisih index tabel dan value
91                     hari di database
92                     $rowIdx = $dataHariIni->jam_mulai - 6; // + 1 karena perbedaan selisih index tabel dan
93                     value jam_mulai di database
94                     if ($dataHariIni->jenis == "konsultasi") {
95                         $color = "#FEFF00";
96                     } else if ($dataHariIni->jenis == "kelas") {
97                         $color = "#FFFFFF";
98                     } else {
99                         $color = "#92D14F";
100                     }
101                     ?>
102                     <script >
103                         var table = document.getElementById('jadwal_table<?php_echo_$idx;_?>');
104                         var $cellLocation = "#t<?php_echo_$idx;_?>cell<?php_echo_$rowIdx;_?>-<?php_echo_$colIdx;_?>";
105                         $(($cellLocation)).css('background-color', '<?php_echo_$color;_?>');
106                         $(($cellLocation)).attr('rowspan', '<?php_echo $dataHariIni->durasi ?>');
107                         //menghapus cell-cell yang tergeser karena rowspan
108                         for (i = <?php_echo ($rowIdx + 1); ?>; i < <?php_echo ($rowIdx + $dataHariIni->durasi)
109                             ; ?>; i++) {
110                             $("<?php_echo_$idx;_?>cell" + i + "-" + <?php_echo $colIdx; ?>).remove();
111                         }
112                         $(($cellLocation)).html("<?php_echo_$dataHariIni->label_?>");
113
114                     </script>
115                     <?php
116                 }
117                 ?>
118             </table>
119             <?php
120             if($scurrRow!=null){
121                 $palingBaru=date('Y-m-d_H:i:s', 0); // datetime paling awal di PHP (1 Januari 1970 00:00:00)
122                 agar langsung tertimpa di foreach di bawah
123                 foreach($scurrRow as $perJadwal){
124                     if($palingBaru < $perJadwal->lastUpdate){
125                         $palingBaru = $perJadwal->lastUpdate;
126                     }
127                 }
128                 $tgl = date('d',strtotime($palingBaru));
129                 $tgl = $tgl.'_'.$namaBulan[date('m',strtotime($palingBaru)) - 1]; // minus 1 karena index dimulai
130                 dari 0
131                 $tgl = $tgl.'_' . date('Y',strtotime($palingBaru));
132                 ?>
133                 Terakhir diupdate pada : <?=$tgl?><br>
134             <?php
135             }
136             else{
137                 $palingBaru=FALSE;
138                 ?>
139                 Terakhir diupdate pada : Belum ada jadwal <br>
140             <?php
141             }
142             ?>
143             <a href="/LihatJadwalDosen/export/" class="button">Ekspor ke XLS</a>

```

```
142|                 </div>
143|                 <?php
144|                 $idx++;
145|             }
146|         ?>
147|
148|         </div>
149|     </div>
150| </div>
151|
152| <?php $this->load->view('templates/script_foundation'); ?>
153| </body>
154| </html>
```


LAMPIRAN C

KODE PROGRAM MODEL JADWALDOSEN

Kode program untuk mengimplementasikan model JadwalDosen

Listing C.1: JadwalDosen_model.php

```
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class JadwalDosen_model extends CI_Model {
6
7     const DAY_NAME = ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat'];
8     const MONTH_NAME = ['Januari', 'Februari', 'Maret', 'April', 'Mei', 'Juni', 'Juli', 'Agustus', 'September', 'Oktober', 'November', 'Desember'];
9
10    /**
11     * Mendapatkan seluruh request dari email tertentu
12     * @param type $email email yang melakukan request atau NULL untuk semua
13     * @return array hasil dari $query->result()
14     */
15    public function requestsBy($email, $rows = NULL, $start = NULL) {
16        if ($email !== NULL) {
17            $this->db->where('requestByEmail', $email);
18        }
19        if ($start !== NULL && $rows !== NULL) {
20            $this->db->limit($rows, $start);
21        }
22        $this->db->from('jadwal_dosen');
23        $this->db->order_by('requestDateTime', 'DESC');
24        $query = $this->db->get();
25        return $query->result();
26    }
27
28    public function addJadwal($data) {
29        $this->user = $data['user'];
30        $this->hari = $data['hari'];
31        $this->jam_mulai = $data['jam_mulai'];
32        $this->durasi = $data['durasi'];
33        $this->jenis = $data['jenis_jadwal'];
34        $this->label = $data['label_jadwal'];
35        $this->lastUpdate=date('Y-m-d_H:i:s');
36        $this->db->insert('jadwal_dosen', $this);
37    }
38
39    public function getAllJadwal() {
40        $query = $this->db->query('SELECT_jadwal_dosen.*,_bluetape_userinfo.name
41        FROM_jadwal_dosen
42        INNER_JOIN_bluetape_userinfo_ON_jadwal_dosen.user=bluetape_userinfo.email');
43        return $query->result();
44    }
45
46    public function getJadwalByUsername($user) {
47        $query = $this->db->get_where('jadwal_dosen', array('user' => $user));
48        return $query->result();
49    }
50
51    public function updateJadwal($id_jadwal, $data) {
52        $this->db->where('id', $id_jadwal)->update('jadwal_dosen', $data);
53    }
54
55    public function deleteJadwal($id_jadwal) {
56        $this->db->where('id', $id_jadwal)->delete('jadwal_dosen');
57    }
58
59    public function getNamaHari() {
60        return JadwalDosen_model::DAY_NAME;
61    }
62
63    public function getNamaBulan() {
64        return JadwalDosen_model::MONTH_NAME;
65    }
66
67    public function kolomKeHari($namaHari) {
68        return strpos("BCDEF", $namaHari);
69    }
70
71    public function hariKeKolom($col) {
72        return substr("BCDEF", $col, 1);
73    }
```

```
74 | }
75 |
76 | public function deleteByUsername($username){
77 |     $this->db->where('user',$username)->delete('jadwal_dosen');
78 | }
79 |
80 | public function cekJadwalByJamMulai($jam_mulai,$hari,$user){
81 |     $query = $this->db->get_where('jadwal_dosen', array('jam_mulai' => $jam_mulai, 'hari' =>$hari, 'user' =>$user ));
82 |     return $query->result();
83 |
84 | }
85 | }
```

LAMPIRAN D

KODE PROGRAM UNTUK LOGIN

Kode program untuk mengimplementasikan Google OAuth

Listing D.1: auth-dev.php

```
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 $config['domain'] = 'http://localhost';
6 $config['google-clientid'] = 'FILLME';
7 $config['google-clientsecret'] = 'FILLME';
8 $config['google-redirecturi'] = $config['domain'] . '/auth/oauth2callback';
9
10 $config['email-config'] = Array(
11     'protocol' => 'smtp',
12     'smtp_host' => 'ssl://smtp.googlemail.com',
13     'smtp_port' => 465,
14     'smtp_user' => 'xxx',
15     'smtp_pass' => 'xxx',
16     'mailtype' => 'html',
17     'charset' => 'iso-8859-1'
18 );
```

Kode program untuk mengatur hak akses pengguna.

Listing D.2: modules.php

```
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 $config['module-names'] = array(
6     'TranskripRequest' => 'Cetak_Transkrip',
7     'TranskripManage' => 'Manajemen_Cetak_Transkrip',
8     'PerubahanKuliahRequest' => 'Perubahan_Kuliah',
9     'PerubahanKuliahManage' => 'Manajemen_Perubahan_Kuliah',
10    'EntriJadwalDosen' => 'Entri_Jadwal_Dosen',
11    'LihatJadwalDosen' => 'Lihat_Jadwal_Dosen'
12 );
13
14 $config['modules'] = array(
15     'TranskripRequest' => array('root', 'mahasiswa.ftis'),
16     'TranskripManage' => array('root', 'tu.ftis'),
17     'PerubahanKuliahRequest' => array('root', 'staf.unpar'),
18     'PerubahanKuliahManage' => array('root', 'tu.ftis'),
19     'EntriJadwalDosen' => array('root', 'dosen.informatika', 'adrian_skripsi'),
20     'LihatJadwalDosen' => array('root', 'mahasiswa.informatika', 'dosen.informatika', 'adrian_skripsi')
21 );
22
23 $config['roles'] = array(
24     'root' => array('pascal@unpar.ac.id', 'shao.wei@unpar.ac.id', 'adrianreynaldi@yahoo.com'),
25     'adrian_skripsi' => array('adrianreynaldi@yahoo.com'),
26     'tu.ftis' => array('shao.wei@unpar.ac.id', 'pranyoto@unpar.ac.id', 'walip@unpar.ac.id', 'dwina@unpar.ac.id'),
27     'mahasiswa.ftis' => '(7[123]\\d{5})|(20[1-9][0-9]7[123][0-9]{4})@student\\.unpar\\.ac\\.id',
28     'staf.unpar' => '.+@unpar\\.ac\\.id',
29     'dosen.informatika' => array('cheni@unpar.ac.id', 'mariskha@unpar.ac.id', 'nico@unpar.ac.id', 'anung@unpar.ac.id', 'moertini@unpar.ac.id', 'aditya-bagoes@unpar.ac.id', 'chandraw@unpar.ac.id', 'elisatih@unpar.ac.id', 'gkarya@unpar.ac.id', 'husnulhakim@unpar.ac.id', 'joanna@unpar.ac.id', 'lionov@unpar.ac.id', 'luciana@unpar.ac.id', 'claudio-fransiscus@unpar.ac.id', 'pascal@unpar.ac.id', 'rosad5@unpar.ac.id', 'vania-natali@unpar.ac.id', 'reynaldi95@gmail.com'),
30     'mahasiswa.informatika' => '73\\d{5}@student\\.unpar\\.ac\\.id'
31 );
32 );
```


LAMPIRAN E

KODE PROGRAM UNTUK MIGRASI

Kode program untuk pembuatan tabel sql *jadwal_dosen*.

Listing E.1: 20170508062800__EntriJadwalDosen_Jadwal.php

```
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Migration_EntriJadwalDosen_Jadwal extends CI_Migration {
6
7     public function up() {
8         $fields = array(
9             'id' => array(
10                 'type' => 'int',
11                 'auto_increment' => TRUE
12             ),
13             'user' => array( // Seharusnya mereferensi ke kolom email pada table bluetape_userinfo
14                 'type' => 'VARCHAR',
15                 'constraint' => '256'
16             ),
17             'hari' => array(
18                 'type' => 'int'
19             ),
20             'jam_mulai' => array(
21                 'type' => 'int'
22             ),
23             'durasi' => array(
24                 'type' => 'int'
25             ),
26             'jenis' => array(
27                 'type' => 'VARCHAR',
28                 'constraint' => '256'
29             ),
30             'label' => array(
31                 'type' => 'VARCHAR',
32                 'constraint' => '100'
33             ),
34         );
35         $this->dbforge->add_field($fields);
36         $this->dbforge->add_key('id', TRUE);
37         $this->dbforge->create_table('Jadwal_dosen');
38
39     }
40
41     public function down() {
42
43     }
44
45 }
46 }
```

Kode program untuk menambahkan field lastUpdate di tabel sql *jadwal_dosen*.

Listing E.2: 20171024144500__EntriJadwalDosen_Jadwal_addLastUpdate.php

```
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Migration_EntriJadwalDosen_Jadwal_addLastUpdate extends CI_Migration {
6
7     public function up() {
8         $newFields = array(
9             'lastUpdate' => array('type' => 'DATETIME',
10                 'default' => date('Y-m-d_H:i:s')
11             )
12         );
13         $this->dbforge->add_column('Jadwal_dosen', $newFields);
14         // Artinya: ALTER TABLE Jadwal_dosen ADD lastUpdate DATETIME
15     }
16
17     public function down() {
18
19     }
20
21 }
```