

# ANALISA DAN IMPLEMENTASI PERBAIKAN PERANGKAT LUNAK BLUETAPE

STEPHEN HADI—2017730016

## 1 Data Skripsi

Pembimbing utama/tunggal: **Pascal Alfadian**

Pembimbing pendamping: -

Kode Topik : **PAN4901**

Topik ini sudah dikerjakan selama : **1 semester**

Pengambilan pertama kali topik ini pada : **Semester 49 - Ganjil 20/21**

Pengambilan pertama kali topik ini di kuliah : **Skripsi 1**

Tipe Laporan : **B** - Dokumen untuk reviewer pada presentasi dan **review Skripsi 1**

## 2 Latar Belakang

Berkembangnya FTIS UNPAR disertai dengan tersedianya semakin banyak matakuliah, dari perkembangan tersebut muncul permasalahan baru di bagian bidang administrasi. Jika dosen ingin meniadakan perkuliahan atau mengganti perkuliahan akan tidak efisien jika melakukan panggilan atau *email* ke pihak tata usaha. Hal tersebut akan memberatkan pihak tata usaha. Mahasiswa yang ingin melakukan pengajuan transkrip akan membuang waktu dan tenaga, karena saat mahasiswa ingin melakukan pengajuan transkrip maka mahasiswa harus datang ke tata usaha melakukan pengajuan dan menunggu beberapa hari untuk mendapatkan hasil transkrip tersebut. Mahasiswa yang tidak memiliki perkuliahan pada hari tersebut harus datang hanya untuk melakukan pengajuan. Hal ini selain membuang waktu juga membuang biaya transportasi.

*Bluetape* adalah aplikasi web yang dibuat oleh dosen dan mahasiswa informatika. Aplikasi ini dibuat menggunakan *Hypertext Preprocessor* atau lebih dikenal dengan *PHP*. *Database management system* atau DBMS yang digunakan adalah *MYSQL*. *Bluetape* menggunakan *framework Bootstrap* dan *Codeigniter*. *Bluetape* berguna untuk membantu kegiatan administrasi FTIS UNPAR. Aplikasi ini dapat melakukan *transkrip request/manage* dan *request perubahan kuliah/manage*. Sehingga jika dosen ingin meniadakan/mengganti perkuliahan dapat dilakukan dengan mudah tanpa harus membuat email ataupun melakukan panggilan. Mahasiswa dapat melakukan permintaan transkrip nilai tanpa tatap muka sehingga mahasiswa hanya perlu datang ke UNPAR saat ingin mengambil hasil dari transkrip tersebut, selain itu mahasiswa juga dapat melihat jadwal dosen. Dengan adanya sistem otomasi pada kegiatan administrasi tentunya pekerjaan tata usaha menjadi lebih ringan.

Aplikasi *Bluetape* digunakan oleh mahasiswa FTIS, dosen, dan tata usaha. Dari pengguna-pengguna tersebut tentu akan ada hal yang disukai dan hal yang tidak disukai. Seperti ada fitur yang bermasalah, ada fitur yang kurang atau hanya saran untuk fitur kedepannya yang akan memudahkan kegiatan administrasi dalam FTIS UNPAR. Dengan adanya masukan-masukan dari pengguna maka aplikasi *Bluetape* dapat ditingkatkan penggunaannya dan memperbaiki kelemahan dari *Bluetape*.

Pada topik skripsi ini akan dilakukan analisis untuk perbaikan dan penambahan fitur untuk aplikasi *Bluetape*, penambahan fitur akan disurvei kepada semua pengguna *Bluetape*. Pengguna yang akan disurvei tidak semua. Hanya perwakilan-perwakilan dari pihak dosen, tata usaha, dan mahasiswa. Selanjutnya akan dianalisa bersama pembimbing untuk menentukan fitur yang akan dirancang dan diimplementasi. Survei tidak terbatas hanya pada fitur tambahan, fitur-fitur yang tidak menghasilkan sesuai kegunaannya akan diperbaiki juga.

Sebelum melakukan perancangan dan implementasi diperlukan untuk *setup* aplikasi *Bluetape* pada komputer sendiri dan ada persyaratan yang harus dipenuhi yaitu mendaftarkan diri ke *google OAuth 2.0* diakrenakan *login* pada *Bluetape* menggunakan akun *gmail* UNPAR. Selanjutnya juga akan dipelajari *framework CodeIgniter* dan *Bootstrap* yang menjadi dasar dari aplikasi ini.

### 3 Rumusan Masalah

Rumusan masalah yang telah diidentifikasi sebagai berikut:

1. Apa sajakah kebutuhan yang diinginkan oleh pengguna bluetape?
2. Bagaimana menganalisa, merancang, dan mengimplementasi *feedback-feedback* dari pengguna?
3. Bagaimana melakukan pengujian setelah mengimplementasi *feedback-feedback* dari pengguna?

### 4 Tujuan

Tujuan pembuatan dan penulisan skripsi ini adalah sebagai berikut:

1. Mendapatkan *feedback-feedback* dari pengguna terkait kebutuhan yang diinginkan.
2. Menganalisa, merancang, dan mengimplementasi *feedback-feedback* dari pengguna
3. Melakukan pengujian setelah mengimplementasi semua *feedback* yang memungkinkan untuk diimplementasi

### 5 Detail Perkembangan Pengerjaan Skripsi

Detail bagian pekerjaan skripsi sesuai dengan rencan kerja/laporan perkembangan terakhir :

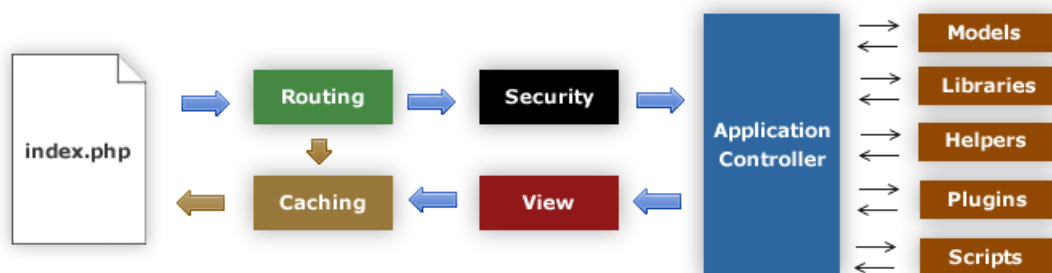
1. Memahami dan mengerti cara kerja *framework code igniter, bootstrap, phpspreadheet*.

**Status :** Ada sejak rencana kerja skripsi, kecuali *phpspreadsheet*.

**Hasil :** Cara kerja *code igniter, bootstrap*, dan *phpspreadsheet* telah dipelajari, Hasil dari studi secara garis besar:

(a) **Code Igniter**

*CodeIgniter*<sup>1</sup> adalah *framework* untuk pembuatan *website* yang menggunakan *PHP*. *CodeIgniter* mempermudah *developer* untuk meminimalisir penggunaan kode untuk mengakses suatu fungsi.



Gambar 1: Flowchart MVC

<sup>1</sup>Version 3.1.11 (2019) *CodeIgniter* 3. British Columbia Institute of Technology. 3700 Willingdon Ave

*CodeIgniter* menerapkan arsitektur *MVC* yang dapat dilihat pada gambar 1, file `index.php` berfungsi mengatur routing dan mengarahkan ke *application controller* yang berada di direktori *controller* dan melalui *controller* akan dipanggil *models*, *libraries*, *helpers*, *etc* yang dibutuhkan dengan perintah `$this->load`. *MVC* secara umum akan mengolah data pada *models* dan hasil yang sudah siap akan dikirim ke *view* melalui *controller*. Fitur tambahan dari arsitektur *CodeIgniter* adalah saat *router* memeriksa *HTTP request*, jika *cache* tersedia maka akan dikirimkan *cache* tersebut dan jika tidak ada *cache* maka *security* akan memeriksa dan melakukan filter terhadap *HTTP request* seperti pada gambar 1.

### **Controller**

*Controller* adalah pusat dari aplikasi, *controller* menangani apa yang harus dilakukan dari *HTTP request*. Dalam *CodeIgniter* untuk menginisiasi *controller* cukup menulis nama kelas diikuti dengan `extends CI_Controller`.

### **Model**

*Model* berfungsi sebagai *logic* dari aplikasi. *Model* pada *CodeIgniter* bersifat opsional, tetapi disediakan untuk *developer* yang ingin menggunakan *MVC*.

### **View**

*View* tidak pernah dipanggil secara langsung, *view* harus dipanggil melalui *controller*. *View* pada *CodeIgniter* ditaruh pada direktori *view*. Pemanggilan *view* menggunakan method `$this->load->view('nama_view')`.

### **Kelas pada code igniter**

- **CI\_Loader**

*CI\_Loader* adalah kelas yang berfungsi untuk *load elements*. *Elements* dapat berupa *libraries*, *view files*, *drivers*, *helpers*, *models*. Kelas ini sudah diinisialisasi secara otomatis oleh *code igniter*.

- **CI\_Config**

*CI\_Config* adalah kelas yang digunakan untuk mengambil konfigurasi yang dibuat. *Method* yang digunakan: `item`. *Method item* digunakan untuk mengambil variabel `$config`.

- **CI\_Email**

*CI\_Email* adalah kelas yang berfungsi untuk mempermudah pengiriman *email* yang disediakan oleh *codeigniter*. Kelas ini memiliki *method*:

- `from`

*Method* ini digunakan untuk memasang alamat *e-mail* pengirim dan nama pengirim.

- `to`

*Method* ini digunakan untuk menentukan tujuan.

- `subject`

*Method* ini digunakan untuk memasang subjek pesan.

- `message`

*Method* ini digunakan untuk memasang pesan yang ingin dikirim

- `send`

*Method* ini digunakan untuk mengirim pesan.

- **CI\_Input**

*CI\_Input* adalah kelas yang digunakan untuk menyaring dan membersihkan data masukan untuk meningkatkan keamanan. *Method* yang digunakan:

- `post`

*Method* ini berfungsi membaca masukan dengan tipe *post*.

- `server`

*Method* ini berguna untuk mengambil *server data* seperti `$_SERVER`.

- **CI\_Session**

*Session* digunakan untuk mengurus *user's state* dan melihat aktivitas *user* saat menelusuri situs. Kelas ini sama seperti `$_SESSION` *superglobal*. *Method* yang digunakan:

- `set_flashdata`  
*Method* ini berguna untuk menyimpan data ke `$_SESSION` *superglobal* dan menandai sebagai *flashdata*. *Flashdata* adalah *session data* yang hanya tersedia untuk permintaan berikutnya dan setelah itu akan dibuang.
- `flashdata`  
*Method* ini mengambil nilai dari spesifik `$_SESSION` yang telah ditandai sebagai *flashdata*.
- `set_userdata`  
*Method* ini berguna untuk menyimpan data ke `$_SESSION` *superglobal*.
- `userdata`  
*Method* ini berguna untuk mengambil *value* untuk `$_SESSION` *item* atau *array* semua data user
- `unset_userdata`  
*Method* ini berfungsi untuk membuang data dari `$_SESSION` *superglobal* dengan *key* masukan.

(b) **Phpspreadsheet**

PhpSpreadsheet adalah *library* yang ditulis dengan bahasa PHP berguna untuk membaca dan menulis file dengan jenis spreadsheet seperti Excel dan LibreOffice Calc<sup>2</sup>. Kelas pada *phpspreadsheet*:

- **Spreadsheet**

*Constructor* kelas *Spreadsheet* tidak menerima parameter apapun, dan *return value* adalah *mixed*. *Method* yang digunakan pada kelas *Spreadsheet*:

- `createSheet`  
*Method* ini berfungsi untuk membuat *sheet*.
- `setActiveSheetIndex`  
 Memilih *sheet* yang ingin dijadikan *active* berdasarkan *index*.
- `getActiveSheet`  
 Mengembalikan kelas *Worksheet* yang telah dibuat menjadi *sheet* aktif pada `setActiveSheetIndex`.

- **Worksheet**

Kelas *Worksheet* adalah kelas yang mengatur nilai dari *cell*, *cell style*, judul *sheet* dll. *Method* yang digunakan pada kelas *Worksheet*:

- `getStyle`  
 Mengembalikan kelas *Style*.
- `setCellValue`  
 mengubah suatu nilai pada *cell* tertentu.
- `mergeCells`  
 Melakukan *merge* pada *cell*.
- `setTitle`  
 Berfungsi untuk memberi judul pada *sheet*.
- `getRowDimension`  
 Mengambil dimensi baris pada baris tertentu.
- `getColumnDimension`  
 Mengambil dimensi kolom pada kolom tertentu.

---

<sup>2</sup>Version 1.14.1 (2019) *PhpSpreadsheet*. PHPOffice. The Internet.

- **Style**

Kelas *Style* adalah kelas yang mengatur *style* dari suatu *cell* seperti *alignment*, *fill*, *font* dll. *Method* yang tersedia pada kelas *Style*:

- **getFill**  
Mengembalikan kelas *Fill*. untuk mengubah *fill* pada suatu *cell* dapat memanggil **setFillType()** pada kelas *Fill*.
- **getAlignment**  
Mengembalikan kelas *Alignment*, untuk mengubah *alignment* pada suatu *cell* dapat memanggil **setHorizontal()** atau **setVertical()** pada kelas *Alignment*.
- **getFont**  
Mengembalikan kelas *Font*, untuk mengubah penebalan kata dapat menggunakan **setBold()** dengan masukan *boolean*.

- **Xls**

Kelas *Xls* memiliki 2 tipe yaitu *writer* dan *reader*. Pada penelitian kali ini tipe *Xls* yang digunakan hanya tipe *writer*. *Method* yang digunakan dari kelas *Xls*: **save**. *Method* ini berfungsi untuk membuat file *spreadsheet* tersimpan dalam format *xls*.

(c) **Bootstrap**

*Bootstrap* adalah *framework* paling terkenal untuk membuat *site* yang *mobile-first* dan *responsive*<sup>3</sup>. Komponen-komponen *bootstrap*:

- *Container*

Kelas *container* adalah kelas yang dibutuhkan jika ingin menggunakan *bootstrap* grid. Kelas *container* adalah kelas *responsive* dan *fixed-width*.

- *Grid*

*Grid* pada *bootstrap* digunakan untuk mengatur *layout* dan *alignment*. *Row* adalah kelas *wrapper* untuk kolom. Kelas *col* akan secara otomatis membagi lebar horizontal dari layout menjadi sama besar.

- *Table*

Kelas *table* adalah kelas yang membantu pembuatan *table*. Dengan komponen pendukung yang dapat membuat *table* memiliki *border* ataupun *table* dengan *style striped*.

- *Pagination*

*Pagination* dibuat menggunakan HTML *list* yang berada didalam **<nav>** untuk mengidentifikasi *pagination* sebagai bagian navigasi kepada pengguna. *Pagination* juga menyediakan komponen pendukung seperti menandai *pagination* yang aktif, membuat *link* tidak dapat di klik.

- *Modal*

*Modal* dibuat dengan *html*, *css*, *javascript*. *Modal* akan mengubah *scroll* pada **<body>** sehingga sasaran *scroll* hanya pada *modal*. *Modal* akan tertutup jika pengguna mengklik diluar *modal*.

- *Form*

*Bootstrap* menyediakan kelas yang dapat digunakan untuk menampilkan *form* yang konsisten pada *browser* ataupun *smartphone*.

- *Nav*

*Nav* adalah komponen yang mengatur navigasi dengan memasukkan kelas *nav*. Kelas *nav* dibuat dengan *flexbox* dan menjadi dasar untuk membuat berbagai macam navigasi. Kelas *nav* dapat ditaruh pada tag **<nav>** atau **<ul>**.

---

<sup>3</sup>Version 4.5 (2020) Bootstrap. Bootstrap. San Francisco, United States.

## 2. Melakukan survei ke mahasiswa, tata usaha, dan dosen untuk memahami kebutuhan pengguna

**Status :** Ada sejak rencana kerja skripsi.

**Hasil :** Survei telah dilakukan menggunakan *google form*. Form dibagikan kepada dosen dan tata usaha melalui *mailing list* dan dibagikan ke mahasiswa melalui grup informatika angkatan 2017. Survei telah mendapatkan 14 responden dalam jangka waktu satu minggu.

## 3. Menganalisa fitur kebutuhan bersama pembimbing

**Status :** Ada sejak rencana kerja skripsi.

**Hasil :** Setelah terkumpul jawaban responden, selanjutnya dibuat *issue* pada *github* sejumlah 22 *issue* yang dapat diakses pada <https://github.com/stephenhadi/BlueTape/issues>. Analisis bersama pembimbing dapat dilihat pada gambar 2 .

No	Deskripsi	Issue Number	Status
1	Fitur "Expore ke XLS" pada menu entri jadwal dosen menghasilkan file <i>corrupt</i>	#1	Akan diimplementasi
2	<i>Update google api / phpspreadsheet</i>	#2	Akan diimplementasi
3	Fitur <i>chart</i> pada manajemen cetak transkrip	#3	Akan diimplementasi
4	Fitur <i>chart</i> pada manajemen perubahan kuliah	#4	Akan diimplementasi
5	Mahasiswa dengan NPM baru tidak dapat login dan lihat jadwal dosen	#5	Akan diimplementasi
6	Kolom pada entri jadwal dosen dan lihat jadwal dosen tidak seragam	#6	Akan diimplementasi
7	Fungsi Tab pada lihat jadwal dosen tidak berfungsi	#7	Akan diimplementasi
8	Pengelompokkan rekap perubahan jadwal pada mata kuliah yang sama	#8	Ditunda
9	Menyediakan forum Q&A pada <i>bluetape</i>	#9	Tidak diimplementasi
10	<i>Scheduling</i> matakuliah pada <i>googlemeet/zoom</i>	#10	Tidak diimplementasi
11	Mengubah atau membatalkan permohonan	#11	Akan diimplementasi
12	Menambahkan jam kuliah selesai di perubahan kelas	#12	Akan diimplementasi
13	Memperbaiki form dan link yang tidak aktif	#13	Tidak diimplementasi
14	Mengintegrasi <i>bluetape</i> dengan SSO UNPAR	#14	Tidak diimplementasi
15	Menambah list permintaan pada <i>bluetape</i>	#15	Tidak diimplementasi
16	Pengajuan surat keterangan aktif kuliah secara online	#16	Tidak diimplementasi
17	Notifikasi email untuk mahasiswa jika permintaan sudah diselesaikan	#17	Akan diimplementasi
18	Dapat melihat profil mahasiswa	#18	Tidak diimplementasi
19	Halaman histori dan request transkrip terpisah	#19	Akan diimplementasi
20	Fitur bahasa indonesia dan inggris	#20	Tidak diimplementasi
21	Pagination tidak terstyle dengan baik	#22	Akan diimplementasi
22	Format Datetimepicker tidak konsisten	#23	Akan diimplementasi

Gambar 2: Analisis kebutuhan pengguna perangkat *bluetape*

### Masukan yang akan diimplementasi adalah:

- Fitur "Expore ke XLS" pada menu entri jadwal dosen menghasilkan file *corrupt*
- Update google api / phpspreadsheet*
- Fitur *chart* pada manajemen cetak transkrip
- Fitur *chart* pada manajemen perubahan kuliah
- Mahasiswa dengan NPM baru tidak dapat login dan lihat jadwal dosen
- Kolom pada entri jadwal dosen dan lihat jadwal dosen tidak seragam
- Fungsi Tab pada lihat jadwal dosen tidak berfungsi
- Mengubah atau membatalkan permohonan
- Menambahkan jam kuliah selesai di perubahan kelas

- (j) Notifikasi email untuk mahasiswa jika permintaan sudah diselesaikan
- (k) Halaman histori dan request transkrip terpisah
- (l) Pagination tidak terstyle dengan baik
- (m) Format Datetimepicker tidak konsisten

**Masukan yang ditunda adalah:**

Pengelompokkan rekap perubahan jadwal pada mata kuliah yang sama.

Masukan ini ditunda dan tidak akan dikerjakan kecuali jika ada waktu sebelum *deadline* Skripsi 2.

#### 4. Merancang dan mengimplementasi hasil survei

**Status :** Ada sejak rencana kerja skripsi.

**Hasil :** Hasil survei yang telah dianalisis akan dikerjakan telah diimplementasi dengan mengikuti aturan yang telah ditetapkan bersama pembimbing. Hasil implementasi lengkap dapat dilihat pada <https://github.com/stephenhadi/BlueTape/pulls>.

- Fitur "Expor ke XLS" pada menu entri jadwal dosen menghasilkan file *corrupt*. Error ini tidak dapat direproduksi dan file yang diunduh tidak *corrupt*.

	A	B	C	D	E	F	G	H
1	<b>JADWAL AKTIVITAS DOSEN</b>							
2	<b>Dosen : step step</b>							
3								
4		Senin	Selasa	Rabu	Kamis	Jumat		
5	7-8			Praktikum		Pemrograman		
6	8-9			Pemrograman		Komputer		
7	9-10	Pemrograman	(dengan	Berorientasi		(dengan		
8	10-11	Berorientasi	nerianian)	Praktikum	Layanan	nerianian)		
9	11-12	Objek(A)		Berorientasi	Berbasis Web			
10	12-13	Pemrograman		Persiapan				
11	13-14	Berorientasi	Rapat	Materi Minggu	(dengan			
12	14-15	Objek(B)	Jurusan(jika	Berikutnya	nerianian)			
13	15-16	Praktikum	ada)					
14	16-17	Pemrograman						
15		Komputer (Oleh						
16								
17	Keterangan :		Waktu Konsultasi					
18			Jika Dijadwalkan					
19								
20								

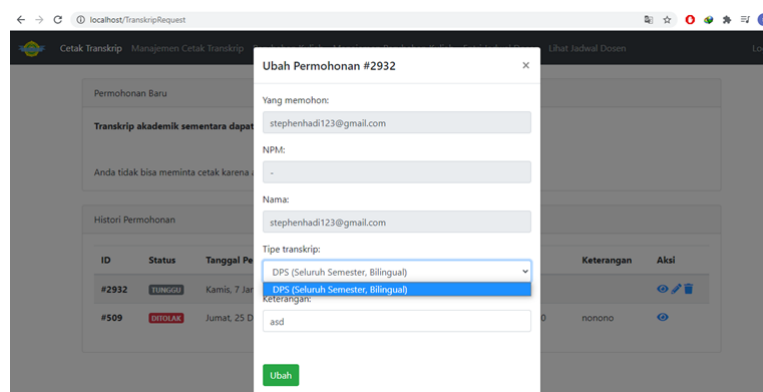
Gambar 3: File Xls yang diunduh

- *Update google api / phpspreadsheet*. Update telah dilakukan dengan merubah `composer.json`.
- Fitur *chart* pada manajemen cetak transkrip. Fitur ini sedang tahap perbaikan dikarenakan kesalahan mengerti maksud dari pengguna.
- Fitur *chart* pada manajemen perubahan kuliah. Fitur ini sedang tahap perbaikan dikarenakan kesalahan mengerti maksud dari pengguna.
- Mahasiswa dengan NPM baru tidak dapat login dan lihat jadwal dosen. Mahasiswa dengan NPM format baru dapat melihat halaman lihat jadwal dosen. Lihat gambar 4.

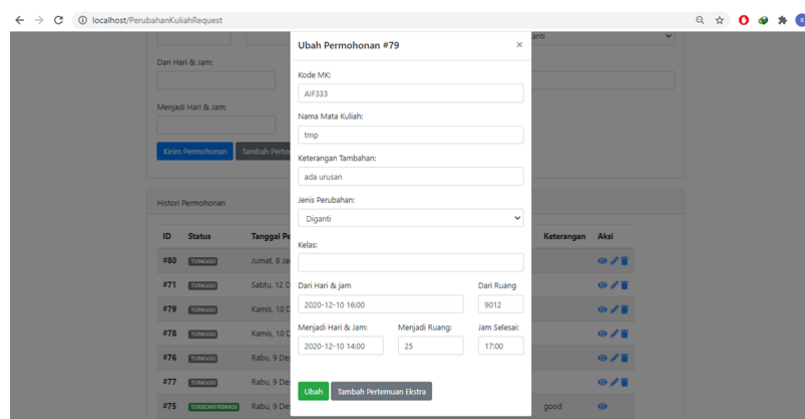
	Senin	Selasa	Rab	Kam	Jumat
7-8	Pengantar Penambangan Data Mining dengan Python				
8-9					
9-10					
10-11					
11-12					
12-13				Praktikum Pemrograman Berorientasi Objek(B)	
13-14					
14-15					
15-16					

Gambar 4: Tampilan lihat jadwal dosen pada NPM 2017730016.

- Kolom pada entri jadwal dosen dan lihat jadwal dosen tidak seragam. Masukan ini telah diperbaiki, Kolom senin sampai jumat memiliki lebar yang seragam seperti pada gambar 4.
- Fungsi Tab pada lihat jadwal dosen tidak berfungsi. Fungsi ini telah diperbaiki dan berjalan seperti pada gambar 4. Tabel lihat jadwal dosen memiliki navigasi *bootstrap* dengan jenis *tab*.
- Mengubah atau membatalkan permohonan. Masukan ini adalah masukan dari dosen dan mahasiswa, sehingga dibuat pada halaman cetak transkrip pada gambar 5 dan pada halaman perubahan kuliah pada gambar 6.



Gambar 5: Fitur *edit* dan *delete* pada halaman cetak transkrip.



Gambar 6: Fitur *edit* dan *delete* pada perubahan kuliah.



- Menambahkan jam kuliah selesai di perubahan kelas. Kolom ini bersifat opsional untuk dimasukkan. Lihat gambar 7.

ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Tanggal Jawab	Keterangan	Aksi
#80	TUNGGU	Jumat, 8 Januari 2021	AIF184123	Ditiadakan			
#71	TUNGGU	Sabtu, 12 Desember 2020	AIF333	Diganti			

Gambar 7: Penambahan jam selesai pada perubahan kuliah.

- Notifikasi email untuk mahasiswa jika permintaan sudah diselesaikan. Setelah dilihat fitur ini telah ada pada *bluetape*.
- Halaman histori dan request transkrip terpisah. Masukan ini belum dikerjakan dan akan diselesaikan sebelum memasuki semester berikutnya.
- Pagination tidak terstyle dengan baik. Pagination telah diperbaiki dan dapat di klik. Lihat gambar 8.

#1170	DITOLAK	Minggu, 27 Desember 2020	69ve	Diganti	
#1175	DITOLAK	Minggu, 27 Desember 2020	23tb	Ditiadakan	
#1165	TERKONFIRMASI	Sabtu, 26 Desember 2020	78dv	Tambahan	

1 2 3 4 5 6

Gambar 8: Pagination pada halaman manajemen perubahan kuliah

- Format *Datetimepicker* tidak konsisten. Dari hari&jam memiliki format YYYY/MM/DD. Hasil yang telah diperbaiki dapat dilihat pada gambar 9

Gambar 9: Format *datetimepicker* yang telah diperbaiki

## 5. Melakukan pengujian fungsional dan eksperimental

**Status:** Ada sejak rencana kerja skripsi.

**Hasil:** Akan dilakukan pada skripsi 2

## 6. Menulis dokumen skripsi

**Status :** Ada sejak rencana kerja skripsi.

**Hasil :** Dokumen skripsi telah dikerjakan sampai Bab 3. Bab 1 telah selesai ditulis, Bab 2 sudah ditulis 80%, dan Bab 3 baru sebagian yang berisi analisis survei.

## 6 Pencapaian Rencana Kerja

Langkah-langkah kerja yang berhasil diselesaikan dalam Skripsi 1 ini adalah sebagai berikut:

1. Memahami dan mengerti cara kerja *framework code igniter* dan *bootstrap*.
2. Melakukan survei ke mahasiswa, tata usaha, dan dosen untuk memahami kebutuhan pengguna.
3. Menganalisa fitur kebutuhan bersama pembimbing.
4. Merancang dan mengimplementasi hasil survei.

## 7 Kendala yang Dihadapi

Kendala - kendala yang dihadapi selama mengerjakan skripsi:

- Tugas mata kuliah lain yang banyak
- Mudah bosan

Bandung, 12/01/2021

Stephen Hadi

Menyetujui,

Nama: Pascal Alfadian  
Pembimbing Tunggal