

# SKRIPSI

KONVERSI ZURB FOUNDATION KE BOOTSTRAP 4



Hapsari Laksmi W

NPM: 2015730037

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN  
2020



**UNDERGRADUATE THESIS**

**CONVERTING ZURB FOUNDATION TO BOOTSTRAP 4**



**Hapsari Laksmi W**

**NPM: 2015730037**

**DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY  
2020**



## **LEMBAR PENGESAHAN**

### **KONVERSI ZURB FOUNDATION KE BOOTSTRAP 4**

**Hapsari Laksmi W**

**NPM: 2015730037**

**Bandung, «tanggal» Juni 2020**

**Menyetujui,**

**Pembimbing Utama**

**Pembimbing Pendamping**

**Pascal Alfadian, M.Comp.**

**Ketua Tim Penguji**

**Anggota Tim Penguji**

**Raymond Chandra Putra, M.T.**

**«penguji 2»**

**Mengetahui,**

**Ketua Program Studi**

**Mariskha Tri Adithia, P.D.Eng**



## **PERNYATAAN**

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

### **KONVERSI ZURB FOUNDATION KE BOOTSTRAP 4**

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,  
Tanggal «tanggal» Juni 2020

Meterai Rp. 6000
---------------------

Hapsari Laksmi W  
NPM: 2015730037



## ABSTRAK

Proses mendesain sebuah website merupakan pekerjaan yang memakan waktu, sehingga untuk mempermudah dan mempercepat proses tersebut para pengembang web memanfaatkan tool seperti CSS Framework. Tool tersebut merupakan library CSS yang dapat langsung digunakan untuk mendesain web dengan cara menggunakan kelas - kelas yang sudah disediakan untuk masing - masing CSS Framework. Saat ini beragam CSS framework dikembangkan dan sudah digunakan untuk kebutuhan desain web seperti Foundation dan Bootstrap. Skripsi dengan judul "Konversi Zurb Zurb Foundation ke Bootstrap 4" menjabarkan proses perubahan CSS Framework dalam website BlueTape. BlueTape merupakan aplikasi yang digunakan FTIS UNPAR untuk melayani layanan yang berbasis paper-based menjadi paperless, saat ini web tersebut menggunakan Foundation 6 untuk CSS Frameworknya. Tujuan skripsi ini adalah merubah penggunaan Foundation 6 pada aplikasi BlueTape menjadi menggunakan Bootstrap 4 untuk CSS frameworknya.

Skripsi ini akan mempelajari keseluruhan framework untuk membangun website BlueTape. Sumber data yang digunakan berasal dari kode yang berada di github untuk proyek BlueTape dan Official Documentation dari framework yang digunakan dan di. Proses perubahan diawali dengan kode website BlueTape dianalisa keseluruhan komponen beserta kelas yang digunakan lalu diubah kode nya menggunakan komponen dan kelas dari Bootstrap 4.

Berdasarkan proses perubahan desain yang dilakukan dengan memperhatikan komponen dan kelas yang digunakan pada masing-masing CSS framework, diperoleh kesimpulan bahwa komponen yang digunakan dalam website BlueTape pada Foundation 6 ada pada Bootstrap 4 sehingga proses konversi dapat dilakukan.

**Kata-kata kunci:** css framework, Bootstrap, Foundation, BlueTape, konversi



## ABSTRACT

The process of designing a website is a time consuming task, so to simplify and speed up the process, web developers use tools such as CSS Framework. A CSS library that can be used directly to design the web by using classes that have been provided for each CSS Framework. Currently various CSS frameworks have been developed and used for web design needs such as Foundation and Bootstrap. The thesis titled "Convert Zurb Zurb Foundation to Bootstrap 4" describes the process of changing the CSS Framework on the BlueTape website. BlueTape is an application used by UNPAR's FTIS to serve paper-based, paperless-based services, at this time the web uses Foundation 6 for its CSS Framework. The purpose of this thesis is to change the use of Foundation 6 in the BlueTape application to use Bootstrap 4 for the CSS framework.

This thesis will study the entire framework for building a BlueTape website. The data source used comes from uploaded code at github for the BlueTape project and Official Documentation of the framework used therein. The change process begins with the BlueTape website code, analyzes all components and classes used, then changes the code using components and classes from Bootstrap 4.

Based on the design change process that is done by taking into account the components and classes used in each CSS framework, it is concluded that the components used on the BlueTape website on Foundation 6 are on Bootstrap 4 so that the conversion process can be carried out.

**Keywords:** css framework, bootstrap, Foundation, BlueTape, convert



*Kepada papa dan adik saya yang selalu mendoakan yang terbaik.*



## KATA PENGANTAR

Puji syukur dan terima kasih kepada Tuhan Yang Maha Esa, yang sudah menuntun penulis untuk menyelesaikan Skripsi dengan judul "Konversi Zurb Foundation ke Bootstrap 4". Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan program Studi Strata 1 Jurusan Teknik Informatika di Universitas Katolik Parahyangan.

Selama penulisan dan penyelesaian skripsi ini, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Pascal Alfadian M.Comp selaku pembimbing skripsi yang telah memberikan bantuan dan nasehatnya serta mengarahkan penulis dalam penyusunan skripsi dari awal hingga akhir.
2. Ibu Mariskha Tri Adithia P.D.Eng sebagai ketua program studi Teknik Informatika FTIS, Universitas Katolik Parahyangan.
3. Seluruh dosen Teknik Informatika yang membantu penulis memahami materi selama belajar di Teknik Infomatika.
4. Teman-teman Teknik Informatika 2015 terutama Nadya Vio, Aria, Ario, Enrico, Reyner, Adit, Yonathan, Kippi yang telah mendukung dan memberikan semangat selama ini.
5. Pihak-pihak yang secara tidak langsung memberikan motivasi kepada penulis untuk menjadi pribadi yang lebih baik.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, sehingga kritik dan saran dari pembaca dapat membantu dalam penyempurnaan skripsi ini.

Bandung, Juni 2020

Penulis



## DAFTAR ISI

<b>KATA PENGANTAR</b>	<b>xv</b>
<b>DAFTAR ISI</b>	<b>xvii</b>
<b>DAFTAR GAMBAR</b>	<b>xix</b>
<b>DAFTAR TABEL</b>	<b>xxi</b>
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	1
1.3 Tujuan . . . . .	1
1.4 Batasan Masalah . . . . .	2
1.5 Metodologi . . . . .	2
1.6 Sistematika Pembahasan . . . . .	2
<b>2 LANDASAN TEORI</b>	<b>3</b>
2.1 BlueTape . . . . .	3
2.1.1 Login . . . . .	3
2.1.2 Dosen . . . . .	3
2.1.3 Dosen Informatika . . . . .	4
2.1.4 Mahasiswa . . . . .	4
2.1.5 Staf Tata Usaha . . . . .	5
2.2 CodeIgniter . . . . .	5
2.2.1 Application Flow Chart . . . . .	6
2.2.2 CodeIgniter URLs . . . . .	6
2.2.3 Model . . . . .	7
2.2.4 View . . . . .	8
2.2.5 Controller . . . . .	11
2.3 jQuery . . . . .	12
2.3.1 Penggunaan Dasar jQuery . . . . .	12
2.3.2 jQuery Selectors . . . . .	13
2.3.3 jQuery Event . . . . .	13
2.3.4 jQuery Traversing . . . . .	15
2.3.5 jQuery HTML . . . . .	16
2.4 Foundation 6 . . . . .	17
2.4.1 Struktur File . . . . .	17
2.4.2 Sistem Grid . . . . .	18
2.4.3 Navigation dan Media Attributes . . . . .	19
2.4.4 Komponen . . . . .	20
2.5 Bootstrap 4 . . . . .	27
2.5.1 Struktur File . . . . .	27
2.5.2 Sistem Grid . . . . .	28

2.5.3	Konten . . . . .	29
2.5.4	Komponen . . . . .	32
2.6	DateTimePicker . . . . .	41
<b>3</b>	<b>ANALISIS WEBSITE BLUETAPE DENGAN FOUNDATION 6</b>	<b>43</b>
3.1	Akses bagi Admin . . . . .	43
3.1.1	Akses Email . . . . .	43
3.1.2	Akses Database . . . . .	44
3.2	Kelas dan file yang dipakai pada tampilan Website . . . . .	44
3.2.1	Penggunaan <i>framework</i> Foundation 6 pada Website . . . . .	44
3.2.2	Halaman Login . . . . .	45
3.2.3	Menu Navigasi . . . . .	46
3.2.4	Halaman Permintaan Cetak Transkrip . . . . .	48
3.2.5	Halaman Manajemen Cetak Transkrip . . . . .	50
3.2.6	Halaman Permintaan Perubahan Kuliah . . . . .	52
3.2.7	Halaman Manajemen Perubahan Kuliah . . . . .	54
3.2.8	Halaman Entri Jadwal Dosen . . . . .	57
3.2.9	Halaman Lihat Jadwal Dosen . . . . .	59
3.3	Perancangan Halaman Website dengan Bootstrap 4 . . . . .	60
3.3.1	Halaman Login . . . . .	60
3.3.2	Menu Navigasi . . . . .	61
3.3.3	Halaman Permintaan Cetak Transkrip . . . . .	62
3.3.4	Halaman Manajemen Cetak Transkrip . . . . .	64
3.3.5	Halaman Permintaan Perubahan Kuliah . . . . .	66
3.3.6	Halaman Manajemen Perubahan Kuliah . . . . .	68
3.3.7	Halaman Entri Jadwal Dosen . . . . .	70
3.3.8	Halaman Lihat Jadwal Dosen . . . . .	72
<b>4</b>	<b>IMPLEMENTASI WEBSITE BLUETAPE DENGAN BOOTSTRAP 4</b>	<b>75</b>
4.1	Daftar File yang Diubah dan Diganti pada Folder . . . . .	75
4.2	Pengaturan Akses bagi <i>admin</i> . . . . .	77
4.2.1	Perubahan Akses URL . . . . .	77
4.2.2	Perubahan Akses Email Admin . . . . .	78
4.3	Perubahan Kode pada Tampilan . . . . .	78
4.3.1	Penggunaan <i>library</i> Bootstrap 4 pada Website . . . . .	78
4.3.2	Halaman Login . . . . .	79
4.3.3	Menu Navigasi . . . . .	81
4.3.4	Halaman Permintaan Cetak Transkrip . . . . .	82
4.3.5	Halaman Manajemen Cetak Transkrip . . . . .	86
4.3.6	Halaman Permintaan Perubahan Kuliah . . . . .	92
4.3.7	Halaman Manajemen Perubahan Kuliah . . . . .	98
4.3.8	Halaman Entri Jadwal Dosen . . . . .	102
4.3.9	Halaman Lihat Jadwal Dosen . . . . .	106
<b>DAFTAR REFERENSI</b>		<b>109</b>

## DAFTAR GAMBAR

2.1	Flow Chart Aplikasi CodeIgniter . . . . .	6
2.2	jQuery Closest . . . . .	15
2.3	jQuery Find . . . . .	16
2.4	Grid pada Foundation 6 . . . . .	19
2.5	<i>Basic Navigation Menu</i> pada Foundation . . . . .	19
2.6	Menu <i>align to right</i> dalam Foundation . . . . .	20
2.7	Menu <i>active state</i> dalam Foundation . . . . .	20
2.8	<i>Menu text</i> dalam Foundation . . . . .	20
2.9	Basic Button pada Foundation . . . . .	21
2.10	Coloring Button pada Foundation . . . . .	21
2.11	Basic Table pada Foundation . . . . .	22
2.12	Basic Table pada Foundation . . . . .	23
2.13	Table Scrolling pada Foundation . . . . .	23
2.14	Text Input pada Foundation . . . . .	24
2.15	Select Menu pada Foundation . . . . .	25
2.16	Select Menu pada Foundation . . . . .	25
2.17	Tabs pada Foundation 6 . . . . .	26
2.18	Reveal pada Foundation 6 . . . . .	27
2.19	Grid pada Bootstrap . . . . .	28
2.20	Pilihan kelas grid pada Bootstrap . . . . .	29
2.21	Tabel default pada Bootstrap . . . . .	29
2.22	Tabel default pada Bootstrap . . . . .	30
2.23	Tabel default pada Bootstrap . . . . .	31
2.24	Menyelaraskan gambar ke kanan dan kiri pada bootstrap . . . . .	32
2.25	<i>Forms Basic</i> pada Bootstrap . . . . .	33
2.26	<i>Column Sizing</i> pada Bootstrap . . . . .	34
2.27	<i>Disabled Basic</i> pada Bootstrap . . . . .	35
2.28	<i>Button</i> pada Bootstrap . . . . .	35
2.29	Tombol <i>dropdown</i> pada Bootstrap . . . . .	36
2.30	Badge pada Bootstrap . . . . .	37
2.31	Card pada Bootstrap . . . . .	37
2.32	Navigation Bar pada Bootstrap . . . . .	39
2.33	Modal pada Bootstrap . . . . .	40
2.34	Ikon <i>Coffee</i> pada Font Awesome . . . . .	40
2.35	Alert pada Bootstrap . . . . .	41
2.36	Datetimepicker pada Bootstrap . . . . .	42
3.1	Analisis halaman login. . . . .	46
3.2	Analisis menu navigasi di layar <i>large</i> . . . . .	47
3.3	Analisis menu navigasi di layar <i>medium</i> dan <i>small</i> . . . . .	47
3.4	Analisis halaman cetak transkrip. . . . .	48
3.5	Analisis modal lihat permintaan cetak transkrip. . . . .	49

3.6	Analisis manajemen cetak transkrip.	50
3.7	Analisis modal pada halaman manajemen cetak transkrip.	52
3.8	Analisis halaman permintaan perubahan kuliah.	53
3.9	Analisis modal lihat.	54
3.10	Analisis halaman manajemen perubahan kuliah.	55
3.11	Analisis modal pada halaman manajemen perubahan kuliah.	56
3.12	Analisis halaman entri jadwal dosen.	57
3.13	Analisis modal edit entri jadwal dosen.	58
3.14	Analisis halaman lihat jadwal dosen.	59
3.15	Konversi halaman login.	60
3.16	Konversi menu navigasi di layar <i>large</i> .	61
3.17	Konversi menu navigasi di layar <i>small</i> dan <i>medium</i> .	62
3.18	Konversi halaman cetak transkrip.	62
3.19	Konversi modal lihat permintaan cetak transkrip.	63
3.20	Konversi manajemen cetak transkrip.	64
3.21	Konversi modal pada halaman manajemen cetak transkrip.	65
3.22	Konversi halaman permintaan perubahan kuliah.	66
3.23	Konversi modal lihat permintaan perubahan kuliah.	67
3.24	Konversi halaman manajemen perubahan kuliah.	68
3.25	Konversi modal pada halaman manajemen perubahan kuliah.	70
3.26	Konversi halaman entri jadwal dosen.	71
3.27	Konversi modal edit pada halaman entri jadwal dosen.	72
3.28	Konversi modal lihat jadwal dosen.	73

## DAFTAR TABEL

3.1 Penggunaan kelas pada halaman login. . . . .	46
3.2 Penggunaan kelas pada bagian notifikasi user. . . . .	46
3.3 Penggunaan kelas pada menu navigasi di layar <i>large</i> . . . . .	47
3.4 Penggunaan kelas pada menu navigasi di layar <i>medium</i> dan <i>small</i> . . . . .	48
3.5 Penggunaan kelas pada halaman utama cetak transkrip. . . . .	49
3.6 Penggunaan komponen label pada kolom status. . . . .	49
3.7 Penggunaan kelas pada modal lihat permintaan transkrip. . . . .	50
3.8 Penggunaan kelas pada halaman manajemen cetak transkrip. . . . .	50
3.9 Penggunaan komponen label pada kolom status histori permohonan. . . . .	51
3.10 Penggunaan kelas font awesome pada kolom aksi. . . . .	51
3.11 Penggunaan kelas pada komponen modal manajemen cetak transkrip. . . . .	52
3.12 Penggunaan kelas pada halaman permintaan perubahan kuliah. . . . .	53
3.13 Penggunaan komponen label pada kolom status histori permohonan. . . . .	53
3.14 Penggunaan kelas komponen modal lihat permintaan perubahan kuliah. . . . .	54
3.15 Penggunaan kelas pada halaman manajemen perubahan kuliah. . . . .	55
3.16 Penggunaan komponen label pada kolom status permohonan perubahan kuliah. . . . .	55
3.17 Penggunaan ikon font awesome pada halaman manajemen perubahan kuliah. . . . .	55
3.18 Penggunaan kelas modal manajemen perubahan kuliah. . . . .	57
3.19 Penggunaan kelas pada halaman entri jadwal dosen. . . . .	58
3.20 Penggunaan kelas pada modal edit jadwal dosen. . . . .	59
3.21 Penggunaan kelas pada halaman lihat jadwal dosen. . . . .	60
3.22 Tabel konversi pada halaman login. . . . .	61
3.23 Tabel konversi pada menu navigasi di layar <i>large</i> . . . . .	61
3.24 Tabel konversi pada menu navigasi di layar <i>medium</i> dan <i>small</i> . . . . .	62
3.25 Tabel konversi pada halaman permintaan cetak transkrip. . . . .	63
3.26 Tabel konversi pada modal lihat permintaan cetak transkrip. . . . .	64
3.27 Tabel konversi pada halaman manajemen cetak transkrip. . . . .	64
3.27 Tabel konversi pada halaman manajemen cetak transkrip (cont.). . . . .	65
3.28 Tabel konversi modal pada manajemen cetak transkrip. . . . .	66
3.28 Tabel konversi modal pada manajemen cetak transkrip(cont.). . . . .	66
3.29 Tabel konversi pada halaman permintaan perubahan kuliah. . . . .	67
3.30 Tabel konversi pada modal lihat permintaan perubahan kuliah. . . . .	68
3.31 Tabel konversi pada halaman manajemen perubahan kuliah. . . . .	69
3.32 Tabel konversi pada modal manajemen perubahan kuliah. . . . .	70
3.33 Tabel konversi pada halaman entri jadwal dosen. . . . .	71
3.34 Tabel konversi pada modal edit pada halaman entri jadwal dosen. . . . .	72
3.35 Tabel konversi kelas pada lihat jadwal dosen. . . . .	73
4.1 Penjelasan kode yang dipakai pada halaman manajemen perubahan kuliah. . . . .	80
4.2 Penjelasan kode konversi halaman login. . . . .	81
4.3 Penjelasan kode konversi navigasi bar. . . . .	82
4.4 Penjelasan kode permintaan cetak transkrip. . . . .	86

4.5	Penjelasan kode konversi manajemen cetak transkrip.	91
4.6	Penjelasan kode permintaan perubahan kuliah.	97
4.7	Penjelasan kode manajemen perubahan kuliah.	102
4.8	Penjelasan kode permintaan perubahan kuliah.	106
4.9	Penjelasan kode permintaan perubahan kuliah.	108

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

BlueTape merupakan aplikasi berbasis *web* yang berfungsi mengolah beberapa kebutuhan administrasi fakultas secara *paperless* yang digunakan dalam lingkungan FTIS UNPAR. Aplikasi ini mempunyai fitur untuk manajemen transkrip nilai, perubahan kuliah dan jadwal dosen. *Framework* yang digunakan dalam aplikasi BlueTape ada dua yaitu CodeIgniter(bagian 2.2) dan Foundation(bagian 2.4).

Foundation adalah kerangka kerja atau *Framework* untuk semua perangkat, media, dan semua aksesibilitas. Foundation adalah bagian dari *front-end framework* yang responsif dan memiliki kemudahan untuk merancang situs web, aplikasi, dan email. Sehingga akan terlihat lebih menarik saat dilihat dari perangkat mana pun. Foundation bersifat semantik, mudah dibaca, fleksibel, dan sepenuhnya *customizable*. [1].

Sejak Bootstrap diluncurkan pada Agustus 2011, *framework* ini mulai populer. Bootstrap telah berkembang sepenuhnya menjadi proyek yang digerakkan oleh CSS untuk menggunakan sejumlah plugin JavaScript dan ikon yang sejalan dengan *forms* dan *buttons*. Pada dasarnya, ini memungkinkan untuk mendesain web yang responsif. Bootstrap memiliki fitur grid 12-kolom dan *container* selebar 940px yang kuat. Salah satu yang menarik adalah *build tool* di situs web Bootstrap, di mana *developer* dapat menyesuaikan pembangunan sesuai dengan kebutuhan, seperti memilih fitur CSS dan JavaScript yang ingin disertakan dalam situs. [2]

Pada skripsi ini akan dirubah keseluruhan antarmuka untuk setiap modul yang ada di dalam aplikasi BlueTape menggunakan *framework* Bootstrap 4. Saat ini, setiap view menggunakan template yang menampilkan nama *module*, menu navigasi, dan *flash message*.

### 1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas dalam penelitian ini:

1. Bagaimana merubah *template* manajemen cetak transkrip, manajemen perubahan kuliah dan manajemen jadwal dosen dari framework **Foundation 6** ke **Bootstrap 4**
2. Bagaimana *plugin* yang ada di BlueTape saat ini dapat berjalan pada *Bootstrap 4*.

### 1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini :

1. Merubah *template* cetak transkrip nilai, *template* manajemen cetak transkrip, *template* perubahan kuliah, *module* manajemen perubahan kuliah, *template* entri jadwal dosen dan *template* lihat jadwal dosen dengan *framework* *Bootstrap 4*.
2. Memastikan *plugin* yang ada di BlueTape saat ini dapat berjalan dengan baik pada *Bootstrap 4*.

## 1.4 Batasan Masalah

Dalam penelitian ini ditetapkan batasan-batasan masalah sebagai berikut.

1. Aplikasi ini tidak merubah struktur database dan file yang berisi fungsi-fungsi CRUD.
2. Aplikasi ini tidak menambah tampilan baru, hanya merubah penggunaan framework *Foundation 6* sesuai dengan tampilan yang sudah ada menggunakan *Bootstrap 4*

## 1.5 Metodologi

Metode penelitian yang digunakan dalam skripsi ini adalah :

1. Studi literatur memahami mengenai :
  - (a) *framework* CodeIgniter
  - (b) *framework* Bootstrap 4
  - (c) *framework* Foundation 6 dan *plugin - plugin* nya.
2. Membangun antarmuka sesuai tampilan website BlueTape. Proses pembuatan antarmuka dibagi menjadi 3 tahap :
  - (a) Analisis tampilan antarmuka website BlueTape
  - (b) Perancangan tampilan antarmuka
  - (c) Implementasi

## 1.6 Sistematika Pembahasan

Untuk penulisan skripsi ini akan dibagikan dalam 6 bab sebagai berikut :

Bab Pendahuluan

Bab 1 menjelaskan mengenai latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian dan sistematika penulisan.

Bab Landasan Teori

Bab 2 berisi dasar-dasar teori pembuatan antarmuka BlueTape. Dasar-dasar teori yang digunakan diantaranya adalah pemrograman PHP, *framework* CodeIgniter, *framework* Foundation 6, *framework* Bootstrap 4.

Bab Analisis

Bab 3 berisi analisis antarmuka yang sudah ada dan detil kelas yang akan digunakan pada tahap implementasi.

Bab Implementasi

Bab 4 berisi mengenai isi program yang sudah diimplementasi.

Bab Kesimpulan dan Saran

Bab 5 berisi mengenai kesimpulan dan saran dari peng�aan skripsi.

## BAB 2

### LANDASAN TEORI

#### 2.1 BlueTape

Bluetape merupakan aplikasi berbasis web, berguna sebagai aplikasi yang menunjang proses administrasi dalam lingkungan FTIS UNPAR. Web ini dapat diakses pada link: <http://www.bluetape.azurewebsites.net>. [3]

##### 2.1.1 Login

Halaman utama aplikasi BlueTape akan mengarahkan *user* untuk *login* dengan menggunakan Google, user akan login dengan melihat beberapa kondisi ini:

- Apabila *user* belum pernah login menggunakan akun UNPAR(xxx@student.unpar.ac.id atau yyy@unpar.ac.id) maka *user* akan diminta untuk memasukan email UNPAR dan password
- Apabila user sudah pernah login menggunakan akun UNPAR, maka *user* akan diminta untuk memilih akun beserta password.
- User akan terhubung otomatis dengan akun @gmail.com. Apabila BlueTape menolak autentifikasi user maka: User akan diminta untuk buka halaman Gmail lalu klik avatar di kanan atas dan memilih akun UNPAR yang tepat pada tombol "Add Account"

User akan melihat beberapa menu sesuai dengan *role* user, sebagai mahasiswa, staf TU, dll.

##### 2.1.2 Dosen

###### Perubahan Kuliah

Modul perubahan kuliah berguna untuk mengirimkan permintaan perubahan mata kuliah yang dikirim oleh dosen kepada staf Tata Usaha. Kolom - kolom yang terdapat dalam modul ini:

- Kode MK (Mata Kuliah)
- Nama Mata Kuliah
- Kelas
- Jenis perubahan (diganti / tambahan / ditiadakan)
- Dari (hari/jam dan ruang) dan ke(hari/jam dan tempat)
- Keterangan

Apabila ada kolom yang belum dapat diisi(contoh : dosen belum tahu tempat kelas pengganti) maka kolom kelas dapat dikosongkan. Dosen juga dapat membuat lebih dari 1 kelas pengganti, dengan mengklik tombol "Tambah Pertemuan Ekstra". Setelah dosen klik "Kirim Permohonan", maka sistem akan mengirim permohonan ke halaman BlueTape bagian Tata Usaha utnuk diperiksa, disetujui, dan dicetak sebagai pengumuman. Jika staf Tata Usaha telah selesai mengkonfirmasi(atau menolak), maka dosen akan mendapatkan e-mail notifikasi.

### 2.1.3 Dosen Informatika

#### Entri Jadwal Dosen

Dosen informatika dapat menggunakan menu ini untuk mengisikan jadwal mingguan. Hasil dari pengisian jadwal dapat diekspor ke XLS, atau dapat dilihat oleh mahasiswa informatika melalui portal BlueTape.

#### Tambah Jadwal

Pada bagian entri jadwal, dosen informatika dapat mengisikan hari, jam mulai, durasi, label, dan sejenisnya. Berikut ini jenis yang dapat dipilih:

- Konsultasi : Waktu yang dosen siapkan untuk konsultasi mahasiswa. Pada tabel akan diberi *background* berwarna kuning.
- Terjadwal: Kegiatan mingguan dosen informatika yang telah terjadwal. Contoh : rapat jurusan
- Kelas : Kelas kuliah maupun praktikum.

Lalu dosen dapat klik tombol "Tambah" untuk menambahkan jadwal baru.

#### Ubah/Hapus Jadwal

Dosen dapat mengubah atau menghapus jadwal yang tertera pada tabel. Lalu *pop-up* window akan terbuka dengan pilihan-pilihan yang sesuai dengan permintaan dosen.

#### Hapus Semua

Tombol "Delete All" dapat digunakan untuk menghapus secara cepat seluruh jadwal yang telah dosen buat sebelumnya. Penggunaan tombol ini biasanya digunakan pada awal semester, dimana jadwal yang dosen miliki berubah seluruhnya.

#### Ekspor ke XLS

Tombol "Ekspor ke XLS" berfungsi untuk membuat file excel (.xls) untuk jadwal dosen.

### 2.1.4 Mahasiswa

#### Cetak Transkrip

Mahasiswa dapat menggunakan menu ini untuk mengirimkan permohonan cetak transkrip Mahasiswa mengirimkan permohonan pencetakan transkrip dengan mengisi kolom-kolom pada formulir "Permohonan Baru". Mahasiswa hanya dapat mengirimkan permohonan:

- Maksimal 1x dalam satu semester (kecuali permohonan ditolak)
- Jika ada permohonan yang belum dijawab.

#### Mahasiswa Informatika

#### Lihat Jadwal Dosen

Mahasiswa dapat melihat jadwal mingguan seluruh dosen dengan memilih nama dosen pada seleksi tab, dan tabel jadwal dosen akan ditampilkan pada bagian bawah halaman. Selain itu tabel juga berisi informasi tanggal terakhir dosen meng-update jadwal sehingga mahasiswa dapat melihat apakah jadwal tersebut merupakan jadwal semester ini atau semester lalu. Lalu terdapat tombol

"Ekspor ke XLS" pada halaman lihat jadwal dosen, sehingga mahasiswa dapat menyimpan atau mencetak jadwal tersebut.

### 2.1.5 Staf Tata Usaha

#### Manajemen Perubahan Kuliah

Staf Tata Usaha dapat melakukan manajemen permintaan perubahan kuliah. Sebuah tabel akan menampilkan daftar permohonan dengan menampilkan tanggal kapan permohonan dibuat. Setiap daftar permohonan akan memiliki beberapa tombol :

-  berfungsi untuk melihat detail permohonan sehingga dapat menentukan apakah permohonan disetujui atau tidak.
-  berfungsi untuk membuka pop-up print-out pengumuman.
-  berfungsi sebagai konfirmasi bahwa pengumuman telah dicetak dan disebarluaskan.
-  berfungsi untuk menyatakan bahwa permohonan ditolak. Staf Tata Usaha akan mengisi alasan mengapa permohonan ditolak sehingga tidak membingungkan pemohon.
-  berfungsi untuk menghapus permohonan secara permanen. Staf Tata Usaha diimbau agar tidak menggunakan tombol ini kecuali dalam keadaan terpaksa.

#### Manajemen Cetak Transkrip

Staf Tata Usaha dapat melihat daftar permintaan transkrip dalam bentuk tabel. Keterangan mengenai transkrip dapat dilihat menggunakan tombol  (detail). Selain itu terdapat dua pilihan jawaban dalam setiap daftar permintaan yaitu  (tolak) dan  (cetak). Masing-masing tombol memerlukan keterangan tambahan mengeai alasan mengapa transkrip dapat dicetak maupun ditolak. Modul ini berguna untuk manajemen permohonan cetka transkrip. Terdapat sebuah tabel yang menampilkan daftar pemohonan dengan tanggal yang terurut. Staf Tata Usaha dapat mencari daftar permintaan berdasarkan NPM pemohon. Beberapa tombol yang tersedia untuk setiap permohonan :

-  berfungsi untuk melihat detail permohonan sehingga dapat menentukan apakah permohonan disetujui atau tidak. :
-  berfungsi untuk membuka pop-up print-out pengumuman. Dalam pop-up akan disediakan sebuah link menuju halaman percetakan transkrip pada SIAkad.
-  berfungsi untuk menyatakan bahwa permohonan ditolak. Staf Tata Usaha akan mengisi alasan mengapa permohonan ditolak sehingga tidak membingungkan pemohon.
-  berfungsi untuk menghapus permohonan secara permanen. Staf Tata Usaha diimbau agar tidak menggunakan tombol ini kecuali dalam keadaan terpaksa.

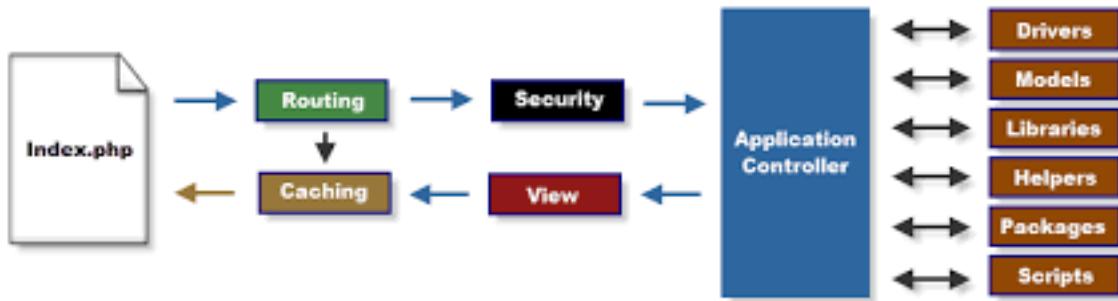
## 2.2 CodeIgniter

CodeIgniter adalah kerangka pengembangan aplikasi atau *toolkit* yang digunakan programmer untuk membangun web menggunakan PHP. Tujuannya adalah memungkinkan programmer mengembangkan proyek-proyek dengan lebih cepat daripada menulis kode dari awal, tersedia banyak library untuk tugas-tugas yang biasa diperlukan dalam membangun aplikasi, serta antarmuka dan struktur logis yang sederhana untuk mengakses library tersebut. Pada bagian ini akan dibahas konsep dasar untuk menjalankan *framework* CodeIgniter:

- *Application Flow Chart* : Penjelasan bagaimana data berjalan dalam sistem.
- URLs : Penjelasan penggunaan *routing* dalam CodeIgniter.
- Konsep Model-View-Controller : Penjelasan pola desain arsitektur pengembangan aplikasi yang digunakan untuk mengatur file sehingga kode mudah dikelola. MVC membuat data, presentasi dan alur dalam aplikasi berjalan secara terpisah.

Pola Model, View, Controller (MVC)

### 2.2.1 Application Flow Chart



Gambar 2.1: Flow Chart Aplikasi CodeIgniter

Gambar 2.1 mengilustrasikan bagaimana alur kerja pembuatan aplikasi berbasis *framework* CodeIgniter:

1. **Index.php** : bertindak sebagai *front controller*, menginisiasi *base resources* yang dibutuhkan untuk menjalankan CodeIgniter.
2. **Router** : akan memeriksa permintaan HTTP untuk menetapkan hal apa yang harus dilakukan dengan permintaan tersebut.
3. **Cache** : Apabila terdapat *cache*, maka *cache* tersebut akan dikirimkan langsung ke browser, dengan melewati sistem eksekusi normal.
4. **Security** : Sebelum *controller* dimuat, *HTTP request* dan *user* mana pun yang mengirimkan data diseleksi dahulu untuk keamanan.
5. **Controller** : Terdiri dari *model*, *core libraries*, *helpers*, dan *resources* yang dibutuhkan untuk proses *request* tertentu.
6. **View** : Tampilan yang telah selesai dirubah kemudian dikirim ke *web browser* untuk dilihat. Jika *caching* diaktifkan, tampilan di *cache* terlebih dahulu sehingga pada permintaan selanjutnya dapat dilayani. [4]

### 2.2.2 CodeIgniter URLs

URL routing (route) adalah salah satu metode untuk memetakan URL ke dalam sumber daya tertentu dengan memberikan nama lain dari sumber daya yang dimaksud. Penerapan *route* pada CodeIgniter menggunakan pendekatan berbasis segmen :

Listing 2.1: Kode URLs pada CodeIgniter

<sup>1</sup> `example.com/class/function/ID`

Segmen URL menggunakan pendekatan Model-View-Controller. Penjelasan mengenai MVC ada pada bagian [2.2.3](#), [2.2.4](#) dan [2.2.5](#). Berikut ini penjelasan dari kode [2.1](#).

1. Segmen pertama menyatakan kelas *controller* yang harus dipanggil.
2. Segmen kedua menyatakan fungsi kelas, atau metode, yang harus dipanggil.
3. Segmen ketiga dan setiap segmen setelahnya menyatakan ID dan variabel apa pun yang akan diteruskan ke *controller*.

### 2.2.3 Model

*Model* merepresentasikan struktur data. Biasanya kelas *model* akan berisi fungsi yang membantu untuk *retrieve*, *insert*, dan *update* informasi di database. Contohnya pada kode [2.3](#) menggambarkan sebuah model yang berisi fungsi *insert*, *update* dan *retrieve*:

Listing 2.2: Struktur model pada codeIgniter.

```

1 class Model_name extends CI_Model {
2     class Blog_model extends CI_Model {
3         public $title;
4         public $content;
5         public $date;
6
7         public function get_last_ten_entries()
8         {
9             $query = $this->db->get('entries', 10);
10            return $query->result();
11        }
12
13        public function insert_entry()
14        {
15            $this->moredelimtitle    = $_POST['title'];
16            $this->content    = $_POST['content'];
17            $this->date      = time();
18
19            $this->db->insert('entries', $this);
20        }
21
22        public function update_entry()
23        {
24            $this->moredelimtitle    = $_POST['title'];
25            $this->content    = $_POST['content'];
26            $this->date      = time();
27
28            $this->db->update('entries', $this, array('id' => $_POST['id']));
29        }
30
31    }
32 }
```

Bagian model akan menjelaskan bagaimana anatomi suatu model , me-load model tersebut dan menghubungkan nya dengan database.

## Anatomi Model

Kelas model akan disimpan dalam direktori `application/models/directory`. Kode [2.3](#) adalah prototipe dasar dari sebuah model kelas :

Listing 2.3: Struktur model pada codeIgniter.

```
1 class Model_name extends CI_Model {  
2  
3 }
```

Dimana `Model_name` merupakan nama dari kelas. Kelas harus menggunakan huruf kapital pada huruf pertama dan sisanya menggunakan huruf kecil. Apabila kode [2.3](#) digunakan maka file akan terlihat seperti kode [2.4](#)

Listing 2.4: Contoh model codeIgniter.

```
1 application/models/User_model.php
```

## Loading a Model

Kemudian model akan dimuat dan dipanggil didalam metode *controller*. Untuk memuat sebuah model dapat menggunakan metode pada kode [2.5](#):

Listing 2.5: Contoh pemanggilan model pada codeIgniter.

```
1 $this->load->model('model_name');
```

## Koneksi ke Database

Model yang sudah dimuat tidak terhubung secara langsung ke database. Secara manual dapat diatur konektivitas database melalui parameter ketiga yang tertera pada kode [2.6](#) :

Listing 2.6: Struktur database pada CodeIgniter.

```
1 $config['hostname'] = 'localhost';  
2 $config['username'] = 'myusername';  
3 $config['password'] = 'mypassword';  
4 $config['database'] = 'mydatabase';  
5 $config['dbdriver'] = 'mysqli';  
6 $config['dbprefix'] = '';  
7 $config['pconnect'] = FALSE;  
8 $config['db_debug'] = TRUE;  
9  
10 $this->load->model('model_name', '', $config);
```

### 2.2.4 View

*View* adalah informasi yang disajikan kepada pengguna. Tampilan atau *View* biasanya akan menjadi halaman web, tetapi dalam CodeIgniter tampilan juga bisa berupa fragmen halaman seperti header atau footer. *Views* tidak pernah dipanggil secara langsung, harus dimuat dalam sebuah *controller*. Dalam *MVC framework*, *controller* bertanggung jawab untuk mengambil *view* tertentu.

Berikut ini akan dijelaskan bagaimana membuat sebuah view, menge-*load* view , menyimpan view tersebut di dalam subdirektori dan menambahkan data dinamis ke dalam view.

## Membuat sebuah View

Untuk membuat sebuah view dalam CodeIgniter, pertama dengan membuat file php yang berisi kode HTML. Misalkan file `blogview.php` yang terlihat pada kode 2.7:

Listing 2.7: Contoh view pada CodeIgniter.

```

1 <html>
2   <head>
3     <title>My Blog</title>
4   </head>
5   <body>
6     <h1>Welcome to my Blog!</h1>
7   </body>
8 </html>
```

Kemudian file php tersebut disimpan dalam direktori `application/views/`.

## Loading sebuah View

Untuk memuat sebuah *view*, maka dapat menggunakan sintaks 2.8.

Listing 2.8: Sintaks load view pada CodeIgniter.

```

1 $this->load->view('name');
```

Apabila terdapat file `Blog.php` yang berisi objek `Blog` dan ingin me-*load* *view*-nya yang terdapat di `blogview.php` maka dapat dipanggil sesuai dengan kode 2.9 pada line ke-6:

Listing 2.9: Contoh load view pada CodeIgniter.

```

1 <?php
2   class Blog extends CI_Controller {
3
4     public function index()
5     {
6       $this->load->view('blogview');
7     }
8 }
```

Setelah diload sebuah view yang dibuat akan menghasilkan URL baru seperti terlihat pada kode 2.10:

Listing 2.10: Contoh load view pada CodeIgniter.

```

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

## Memuat Beberapa View

CodeIgniter mampu menangani beberapa panggilan dari dalam *controller*. Apabila ada lebih dari satu panggilan yang terjadi, maka *views* akan dilampirkan secara bersamaan.

ini kode 2.11 yang digunakan jika *developer* ingin mempunyai sebuah halaman yang terdiri dari **header**, **menu**, **content** dan **footer**.

Listing 2.11: Contoh load beberapa view pada CodeIgniter.

```

1 <?php
2
3   class Page extends CI_Controller {
```

```

4
5     public function index()
6     {
7         $data['page_title'] = 'Your title';
8         $this->load->view('header');
9         $this->load->view('menu');
10        $this->load->view('content', $data);
11        $this->load->view('footer');
12    }
13
14 }
15 ?>

```

### Menyimpan Views didalam Sub Direktori

Untuk menyimpan didalam sub direktori maka dapat menyertakan nama direktori yang memuat *view*, sintaks yang digunakan tertera pada kode [2.12](#):

Listing 2.12: Struktur penyimpanan views dalam sub direktori.

```

1 $this->load->view('directory_name/file_name');

```

### Menambahkan data dinamis ke View

Data yang dikirim dari controller menuju view berbentuk array atau objek, sehingga dapat dilampirkan pada parameter kedua dalam view untuk metode loading.

Kode [2.13](#) menjelaskan penggunaan dengan array:

Listing 2.13: Struktur data dinamis.

```

1 $data = array(
2     'title' => 'My Title',
3     'heading' => 'My Heading',
4     'message' => 'My Message'
5 );
6
7 $this->load->view('blogview', $data);

```

Kemudian, kode [2.14](#) menjelaskan penggunaan dengan objek:

Listing 2.14: Struktur data dinamis berbentuk objek.

```

1 $data = new Someclass();
2 $this->load->view('blogview', $data);

```

Sehingga apabila dimasukan ke controller, kode yang ditambahkan seperti terlihat pada kode [2.15](#):

Listing 2.15: Struktur data dinamis dalam controller.

```

1 class Blog extends CI_Controller {
2
3     public function index()
4     {
5         $data['title'] = "My Real Title";
6         $data['heading'] = "My Real Heading";

```

```

7     $this->load->view('blogview', $data);
8 }
9 }
10 }
```

Untuk mengakses data dari model Blog, maka dapat dipanggil dengan variabel nya seperti terlihat pada kode 2.16 :

Listing 2.16: Akses data dinamis dalam file HTML.

```

1 <html>
2   <head>
3     <title><?php echo $title;?></title>
4   </head>
5   <body>
6     <h1><?php echo $heading;?></h1>
7   </body>
8 </html>
```

## 2.2.5 Controller

*Controller* bertindak sebagai penengah antara Model, View dan *resources* lain yang dibutuhkan untuk proses *HTTP requests* dan untuk menghasilkan sebuah halaman web. *Controller* secara sederhana merupakan sebuah file yang dinamakan dengan aturan tertentu sehingga dapat dihubungkan dengan sebuah URL.

Apabila terdapat kelas `Blog` yang terlihat pada kode 2.17.

Listing 2.17: Contoh controller pada codeIgniter.

```

1 <?php
2   class Blog extends CI_Controller {
3
4     public function index()
5     {
6       echo 'Hello World'
7     }
8   }
9 ?>
```

Maka controller yang dipanggil akan menjadi tertera berikut:

Listing 2.18: Controller pada codeIgniter.

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

Dalam kode 2.18, *CodeIgniter* berusaha menemukan *controller* bernama `Blog.php` untuk dimuat. Ketika sebuah nama *controller* sesuai dengan segmen pertama dari sebuah URL, maka URL akan memuatnya.

## Method

Dalam sebuah kelas *controller* akan memiliki beberapa method, lalu untuk memanggil fungsi didalamnya maka *programmer* dapat mengisi segmen kedua dari sebuah url dengan sebuah method. Misalnya pada kode 2.19 terdapat controller dengan dua method: `index()` dan `comments()`.

Listing 2.19: Controller pada codeIgniter.

```

1 <?php
2 class Blog extends CI_Controller {
3
4     public function index()
5     {
6         echo 'Hello World!';
7     }
8
9     public function comments()
10    {
11        echo 'Look at this!';
12    }
13 }
14 ?>

```

Pemanggilan method index dapat secara otomatis dilakukan apabila segmen kedua kosong. Namun ada cara lain untuk menampilkan pesan "Hello World" yang dapat dilakukan seperti terlihat pada kode 2.20:

Listing 2.20: Cara lain pemanggilan method pada codeIgniter.

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

Kemudian untuk memuat method comment() dapat dituliskan seperti kode 2.21:

Listing 2.21: Load metode comments() pada codeIgniter.

```
1 example.com/index.php/blog/comments/
```

## 2.3 jQuery

jQuery merupakan *library* Javascript dimana setiap kode yang dibuat akan disimpan ke dalam *methods* yang dapat dipanggil *user* dengan satu baris kode. Bagian ini akan menjelaskan penggunaan dasar pada jQuery. [5]

### 2.3.1 Penggunaan Dasar jQuery

Sintaks dalam jQuery dibuat untuk memilih elemen dalam HTML dan melakukan beberapa aksi dalam elemen.

Sintaks dasar dalam jQuery tertera pada kode 2.22:

Listing 2.22: Pemanggilan jQuery.

```
1 $(selector).action()
```

Penjelasan sintaks sebagai berikut:

- **\$**: berfungsi untuk mendefinisikan library jQuery.
- **selector**: berfungsi untuk menjalankan *query* atau menemukan elemen HTML.
- **action()**: berfungsi untuk menjalankan elemen.

Berikut ini contoh penggunaan sintaks dalam jQuery:

- **\$(this).hide()**: menyembunyikan elemen saat ini.

- `$(“p”).hide()`: menyembunyikan elemen `<p>`.
- `$(“.test”).hide()`: menyembunyikan elemen `class="test"`.
- `$(“#test”).hide()`: menyembunyikan elemen `id="test"`.

### 2.3.2 jQuery Selectors

`jQuery selector` memungkinkan *user* untuk memilih dan memanipulasi elemen dalam HTML berdasarkan nama, id, kelas, tipe, atribut dan *value* dari atribut. Semua *selectors* dalam jQuery dimulai dengan simbol dollar dan tanda kurung: `$()`.

Kode [2.23](#) menjelaskan ketika user memilih sebuah tombol maka semua elemen `<p>` akan tersembunyi.

Listing 2.23: jQuery Selector pada elemen `<p>`.

```

1 $(document).ready(function(){
2   $("button").click(function(){
3     $("p").hide();
4   });
5 });

```

### 2.3.3 jQuery Event

Semua aksi yang dilakukan *user* dalam web dapat direspon oleh sistem yang dinamakan *events*. Events menunjukkan momen apabila suatu kegiatan terjadi seperti: memindahkan mouse ke sebuah elemen, memilih sebuah radio button dan menekan sebuah elemen.

`$(document).ready`

Seluruh metode jQuery akan berada didalam sebuah `$(document).ready()` :

Listing 2.24: jQuery document ready event.

```

1 $(document).ready(function{
2   // jQuery methods go here...
3 });

```

Sintaks [2.24](#) untuk menghindari kode jQuery berjalan sebelum dokumen siap untuk *di-load*. Contoh aksi yang akan gagal apabila kode berjalan sebelum dokumen siap seperti:

- Mencoba untuk menghilangkan elemen yang belum selesai dibuat.
- Mencoba untuk mendapatkan ukuran gambar yang belum *di-load* seluruhnya.

`click()`

Metode `click()` melampirkan fungsi event handler ke elemen HTML, sehingga fungsi akan dieksekusi ketika *user* mengklik elemen HTML.

Kode [2.25](#) menjelaskan ketika user memilih elemen `<p>`, maka sistem akan memberikan teks berupa kalimat *alert*.

Listing 2.25: jQuery click().

```

1 $("p").click(function(){
2   alert("The paragraph was clicked.");
3 });

```

**change()**

Metode `change()` akan memicu perubahan ketika kode dijalankan. Change event hanya bisa merubah elemen `<input>`, `<textarea>` dan `<select>`. Ada dua sintaks yang dipakai :

- `$(selector).change()`: memicu perubahan event untuk elemen terpilih.
- `$(selector).change(function)`: memicu perubahan event dengan menerapkan fungsi tertentu.

Contoh penggunaan metode ini terlihat pada kode 2.26: sistem memberikan teks peringatan ketika sebuah field `<input>` berubah.

Listing 2.26: jQuery change().

```

1 $("input").change(function(){
2   alert("The text has been changed.");
3 });

```

**select()**

Event select akan memilih teks terpilih di dalam text area atau text field. Metode `select()` memicu event atau fungsi untuk berjalan ketika sebuah event terpilih dijalankan. Ada dua sintaks yang dipakai :

- `$(selector).select()`: memicu perubahan event untuk elemen terpilih.
- `$(selector).select(function)`: memicu perubahan event dengan menerapkan fungsi tertentu.

Contoh penggunaan metode terlihat pada kode 2.27: sistem memberikan sebuah peringatan ketika sebuah teks dipilih didalam sebuah *text field*.

Listing 2.27: jQuery Selector.

```

1 $("input").select(function(){
2   alert("Text marked!");
3 });

```

**preventDefault()**

Metode `preventDefault()` membatalkan event yang *user* kehendaki. Pada kode 2.28 terlihat, apabila terdapat sebuah link maka user tidak bisa menuju link tersebut. Atau apabila ada button "Submit" dalam sebuah form, maka user tidak bisa mengumpulkan form tersebut.

Listing 2.28: jQuery preventDefault().

```

1 document.getElementById("myAnchor").addEventListener("click",
2 function(event){
3   event.preventDefault()
4 });

```

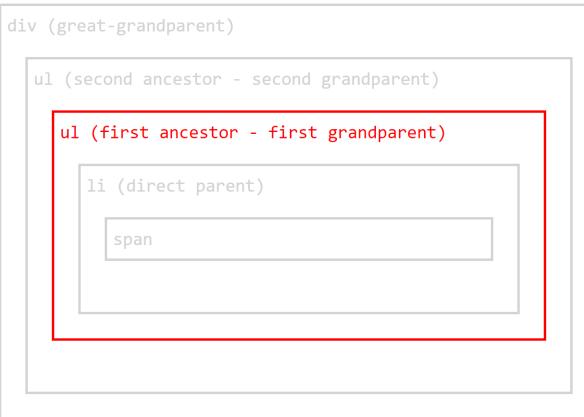
### 2.3.4 jQuery Traversing

Berfungsi untuk menemukan elemen HTML berdasarkan relasinya dengan elemen lain. Metode yang dipakai untuk fungsi [2.2](#) adalah:

#### `closest()`

Fungsi `closest()` akan mengembalikan *ancestor* pertama dari elemen terpilih.

```
body (great-great-grandparent)
```



Gambar 2.2: jQuery Closest

Listing 2.29: jQuery Closest.

```
1 $(document).ready(function(){
2   $("span").closest("ul").css({"color": "red", "border": "2px solid red"});
3 })
```

Kode [2.29](#) akan mengembalikan *ancestor* dari `<span>`, yaitu sebuah elemen `<ul>`. Hasil :

#### `find()`

Metode `find()` mengembalikan elemen turunan/descendant dari elemen yang dipilih oleh *user*. Contoh terlihat pada kode [2.30](#): mengembalikan semua elemen `<span>` yang merupakan turunan dari elemen `<ul>`.

Listing 2.30: jQuery find().

```
1 $(document).ready(function(){
2   $("ul").find("span").css({"color": "red", "border": "2px solid red"});
3 })
```

Hasil dari kode [2.30](#) terlihat pada gambar [2.3](#):

```

body (great-grandparent)
class="descendants">
div (grandparent)

    ul (parent)
        • li (child) span (grandchild)

```

Gambar 2.3: jQuery Find

### 2.3.5 jQuery HTML

#### `remove()`

`remove()` - berfungsi untuk menghapus elemen HTML yang ada. Contoh penggunaan terlihat pada kode 2.31: menghapus elemen dengan yang memiliki `id="div1"`.

Listing 2.31: jQuery remove().

```
1 $("#div1").remove();
```

#### `$removeAttr()`

Metode `removeAttr()` menghapus satu atau banyak atribut dari elemen terpilih. Sintaks dari metode ini terlihat pada kode 2.32:

Listing 2.32: jQuery removeAttr().

```
1 $(selector).removeAttr(attribute)
```

Contoh pemakaian metode ada pada kode 2.33: menghapus atribut style dari semua elemen `<p>`.

Listing 2.33: contoh jQuery removeAttr().

```
1 $("button").click(function(){
2     $("p").removeAttr("style");
3});
```

#### `$clone()`

Metode `clone()` pada kode 2.34 membuat salinan dari elemen terpilih, termasuk child nodes, teks atau atribut. Sintaks dari metode ini :

Listing 2.34: jQuery clone().

```
1 $(selector).clone(true|false)
```

Contoh pemakaian metode ini terlihat pada kode 2.35: Clone semua elemen `<p>` dan masukan semua elemen pada akhir elemen `<body>`.

Listing 2.35: contoh jQuery clone().

```
1 $("button").click(function(){
2     $("p").clone().appendTo("body");
3});
```

**insertBefore()**

Metode `insertBefore()` pada kode [2.36](#) akan memasukan elemen HTML sebelum elemen terpilih. Sintaks dari metode ini :

Listing 2.36: insertBefore().

```
1 $(content).insertBefore(selector)
```

Dimana :

- **content** : Required. Detail konten yang harus dimasukan (harus berisi tags HTML).
- **selector** : Required. Menjelaskan dimana konten akan dimasukan.

Contoh pemakaian metode ini terlihat pada kode [2.37](#): memasukkan sebuah elemen `<span>` sebelum `<p>`.

Listing 2.37: jQuery insertBefore().

```
1 $("button").click(function(){
2   $("<span>Hello world!</span>").insertBefore("p");
3 }) ;
```

## 2.4 Foundation 6

Foundation adalah *front-end framework* yang responsif yang membuatnya mudah untuk merancang situs web, aplikasi, dan email di perangkat apa pun.

### 2.4.1 Struktur File

Berikut ini isi dari *package* Foundation 6 terlihat pada listing [2.38](#).

Listing 2.38: Struktur File Foundation.

```
.
|-- css
|   |-- css
|   |-- foundatcss
|   |-- foundation-datetimepicker.css
|   |-- foundation-fcss
|   |-- foundation-icss
|   |-- foundation-ieot
|   |-- foundation-isvg
|   |-- foundation-ittf
|   |-- foundation-icoff
|-- js
|   |-- vendor
|   |-- app.js
|   |-- foundation.js
|-- img
.
```

Framework Foundation terdiri dari 3 folder utama:

- Folder `css` terdiri dari semua *CSS Style* yang digunakan dalam Foundation 6. Didalam folder terdapat versi yg diperkecil yaitu `foundation.min.css` atau versi yang tidak dikompresi `foundation.css`. Lalu seluruh modifikasi *stylesheets* ditempatkan didalam folder ini agar lebih terstruktur.
- Folder `img` tempat meletakkan semua gambar untuk projek web.
- Folder `js` terdiri dari semua file Javascript.

[1]

#### 2.4.2 Sistem Grid

Kode 2.39 terlihat penggunaan grid pada Foundation dapat dilakukan dengan menambahkan sebuah elemen dengan sebuah kelas `.row` sehingga akan membuat blok horizontal yang berisi kolom vertikal. Kemudian kelas `.column` akan ditambahkan pada baris tersebut, lalu masing-masing kolom ditentukan kelasnya dengan tiga pilihan yaitu `.small-#`, `.medium-#` dan `.large-#`.

Listing 2.39: Struktur Grid Foundation 6.

```

1 <div class="row">
2   <div class="columns small-2 large-4"><!-- ... --></div>
3   <div class="columns small-4 large-4"><!-- ... --></div>
4   <div class="columns small-6 large-4"><!-- ... --></div>
5 </div>
6 <div class="row">
7   <div class="columns large-3"><!-- ... --></div>
8   <div class="columns large-6"><!-- ... --></div>
9   <div class="columns large-3"><!-- ... --></div>
10 </div>
11 <div class="row">
12   <div class="columns small-6 large-2"><!-- ... --></div>
13   <div class="columns small-6 large-8"><!-- ... --></div>
14   <div class="columns small-12 large-2"><!-- ... --></div>
15 </div>
16 <div class="row">
17   <div class="columns small-3"><!-- ... --></div>
18   <div class="columns small-9"><!-- ... --></div>
19 </div>
20 <div class="row">
21   <div class="columns large-4"><!-- ... --></div>
22   <div class="columns large-8"><!-- ... --></div>
23 </div>
24 <div class="row">
25   <div class="columns small-6 large-5"><!-- ... --></div>
26   <div class="columns small-6 large-7"><!-- ... --></div>
27 </div>
28 <div class="row">
29   <div class="columns large-6"><!-- ... --></div>
30   <div class="columns large-6"><!-- ... --></div>
31 </div>
```



Gambar 2.4: Grid pada Foundation 6

### 2.4.3 Navigation dan Media Attributes

Komponen menu yang fleksibel pada Foundation membuat pembangunan navigasi secara umum lebih mudah karena semua pola memiliki markup yang sama.

#### Basic Menu

Menu terdiri dari sebuah `<ul>` yang diisi oleh beberapa tag `<li>`. Secara default, menu akan berorientasi horizontal.

Berikut ini contoh penggunaan kode navigasi pada menu yang terlihat pada kode 2.40):

Listing 2.40: Basic Menu Foundation 6.

```

1 <ul class="menu">
2   <li><a href="#">One</a></li>
3   <li><a href="#">Two</a></li>
4   <li><a href="#">Three</a></li>
5   <li><a href="#">Four</a></li>
6 </ul>

```

One Two Three Four

Gambar 2.5: Basic Navigation Menu pada Foundation

#### Item Alignment

Secara default, setiap item dalam menu akan berlajur ke arah kiri. Menu dapat diubah lajurnya ke arah kanan dengan menggunakan kelas `.align-right` (terlihat pada kode 2.41) didalam kelas `.menu`.

Listing 2.41: Item Alignment Foundation 6.

```

1 <ul class="menu align-right">
2   <li><a href="#">One</a></li>
3   <li><a href="#">Two</a></li>
4   <li><a href="#">Three</a></li>
5   <li><a href="#">Four</a></li>
6 </ul>

```

One Two Three Four

Gambar 2.6: Menu *align to right* dalam Foundation

### Active State

Kelas `.is-active` dapat ditambahkan ke dalam tag `<li>` untuk membuat menu terpilih yang aktif terlihat saat di klik.

Berikut penggunaan kelas yang dicantumkan pada kode [2.42](#):

Listing 2.42: Active State Foundation 6.

```

1 <ul class="menu">
2   <li class="is-active"><a>Home</a></li>
3   <li><a>About</a></li>
4   <li><a>Nachos</a></li>
5 </ul>

```

Hasil dari kode diatas terlihat pada gambar [2.7](#).



Gambar 2.7: Menu *active state* dalam Foundation

### Text

Karena *padding* untuk setiap item menu menggunakan tag `<a>`, maka apabila sebuah item yang hanya berisi teks, maka teks tersebut tidak selaras dengan item menu lainnya. Sehingga untuk menyiasatinya, dapat menggunakan kelas `.menu-text` yang dituliskan dalam tag `<li>` dengan menyertakan sebuah teks tanpa link.

Berikut penggunaan kelas yang dicantumkan pada kode [2.43](#):

Listing 2.43: Text Foundation 6.

```

1 <ul class="menu">
2   <li class="menu-text">Site Title</li>
3   <li><a href="#">One</a></li>
4   <li><a href="#">Two</a></li>
5   <li><a href="#">Three</a></li>
6 </ul>

```

Hasil dari kode diatas terlihat pada gambar [2.8](#):



Gambar 2.8: *Menu text* dalam Foundation

### 2.4.4 Komponen

Selain elemen HTML biasa, Foundation berisi elemen antarmuka lain yang umum digunakan. Ini termasuk tombol dengan fitur canggih (misalnya, pengelompokan tombol atau tombol dengan opsi drop-down, daftar navigasi, tab horizontal dan vertikal, navigasi, label, dan pemformatan untuk pesan seperti peringatan).

## Button

*Basic button* dapat digunakan untuk banyak tujuan, sehingga penting untuk *developer* menggunakan tag yang tepat. Berikut ini penjelasan penggunaan *Basic button* dalam Foundation

- Penggunaan tag `<a>` digunakan apabila tombol memiliki *link* ke halaman lain, atau *link* menuju ke halaman itu sendiri. Penggunaan links tidak membutuhkan JavaScript.
- Penggunaan tag `<button>` jika tombol melakukan tindakan yang mengubah sesuatu pada halaman seperti proses *delete* atau *save*. Elemen `<button>` akan membutuhkan JavaScript agar proses tersebut berfungsi.

Berikut penggunaan kelas yang dicantumkan pada kode 2.44:

Listing 2.44: Button pada foundation 6.

```

1 <!-- Anchors (links) -->
2 <a href="about.html" class="button">Learn More</a>
3 <a href="#features" class="button">View All Features</a>
4
5 <!-- Buttons (actions) -->
6 <button type="button" class="success button">Save</button>
7 <button type="button" class="alert button">Delete</button>

```

Hasil dari kode diatas terlihat pada gambar 2.9:



Gambar 2.9: Basic Button pada Foundation

Warna pada button dapat diterapkan untuk memperlihatkan fungsi yang sesuai dengan aksi yang digunakan. Berikut penggunaan kelas yang dicantumkan pada kode 2.45:

Listing 2.45: Basic Button pada foundation 6.

```

1 <a class="button primary" href="#">Primary</a>
2 <a class="button secondary" href="#">Secondary</a>
3 <a class="button success" href="#">Success</a>
4 <a class="button alert" href="#">Alert</a>
5 <a class="button warning" href="#">Warning</a>

```

Hasil dari kode diatas terlihat pada gambar 2.10:



Gambar 2.10: Coloring Button pada Foundation

## Tabel

Tabel dalam foundation akan menjadikan proses penampilan data bersifat responsif dan memiliki tata letak yang bisa disesuaikan oleh kebutuhan *developer*.

Berikut penggunaan kelas yang dicantumkan pada kode 2.46:

Listing 2.46: Tabel pada foundation 6.

```

1 <table>

```

```

2 <thead>
3   <tr>
4     <th width="200">Table Header</th>
5     <th>Table Header</th>
6     <th width="150">Table Header</th>
7     <th width="150">Table Header</th>
8   </tr>
9 </thead>
10 <tbody>
11   <tr>
12     <td>Content Goes Here</td>
13     <td>This is longer content Donec id elit non mi porta gravida at eget
metus.</td>
14     <td>Content Goes Here</td>
15     <td>Content Goes Here</td>
16   </tr>
17   <tr>
18     <td>Content Goes Here</td>
19     <td>This is longer Content Goes Here Donec id elit non mi porta gravida at
eget metus.</td>
20     <td>Content Goes Here</td>
21     <td>Content Goes Here</td>
22   </tr>
23   <tr>
24     <td>Content Goes Here</td>
25     <td>This is longer Content Goes Here Donec id elit non mi porta gravida at
eget metus.</td>
26     <td>Content Goes Here</td>
27     <td>Content Goes Here</td>
28   </tr>
29 </tbody>
30 </table>

```

Hasil dari kode diatas terlihat pada gambar 2.11:

Table Header	Table Header	Table Header	Table Header
Content Goes Here	This is longer content Donec id elit non mi porta gravida at eget metus.	Content Goes Here	Content Goes Here
Content Goes Here	This is longer Content Goes Here Donec id elit non mi porta gravida at eget metus.	Content Goes Here	Content Goes Here
Content Goes Here	This is longer Content Goes Here Donec id elit non mi porta gravida at eget metus.	Content Goes Here	Content Goes Here

Gambar 2.11: Basic Table pada Foundation

### Stacked Table

Stacked Table diaplikasikan menggunakan kelas `.stack` untuk membuat tumpukan tabel pada layar kecil.

Berikut penggunaan kelas yang dicantumkan pada kode 2.47:

Listing 2.47: Tabel stack pada foundation 6.

```
1 <table class="stack"></table>
```

Hasil dari penggunaan kelas diatas terlihat pada gambar 2.12:

Chocolate Chip
Tastey
120cal
7.5/10
Snickerdoodle
Delicious
95cal
8/10
Oatmeal Raisin
Superb
10cal
11/10

Gambar 2.12: Basic Table pada Foundation

## Scrolling Table

Apabila suatu tabel memiliki banyak data sehingga data yang ditampilkan memenuhi layar secara horizontal, maka dapat menggunakan kelas `.table-scroll`.

Berikut penggunaan kelas yang dicantumkan pada kode 2.48:

Listing 2.48: Scroll tabel pada foundation 6.

```
1 <table class="stack"></table>
```

Hasil dari penggunaan kelas diatas terlihat pada gambar 2.13:

This is the description!	One	Two	Three
These are all the words that people use to describe Foundation 6!	Cool	Swag	Chill
These are some words that people use to describe other web frameworks.	Whatevs	Ugh.	LOL
Here are some great super heros.	Batman	Superman	Spiderman
<b>Here's a footer, just in case</b>			
<span style="float: left;">◀</span> <span style="float: right;">▶</span>			

Gambar 2.13: Table Scrolling pada Foundation

## Forms

Forms pada Foundation dibuat dengan kombinasi standar dari elemen `<form>`, serta `grid rows` dan `columns` atau `cells`.

## Text Inputs

Kode berikut ini akan membuat sebuah `text field` yang bisa diterapkan untuk `field : text, date, datetime, datetime-local, email, month, number, password, search, tel, time, url`, dan

week.

Berikut penggunaan kelas yang dicantumkan pada kode 2.49:

Listing 2.49: Text inputs pada foundation 6.

```

1 <form>
2   <div class="grid-container">
3     <div class="grid-x grid-padding-x">
4       <div class="medium-6 cell">
5         <label>Input Label
6           <input type="text" placeholder=".medium-6.cell">
7         </label>
8       </div>
9       <div class="medium-6 cell">
10        <label>Input Label
11          <input type="text" placeholder=".medium-6.cell">
12        </label>
13      </div>
14    </div>
15  </div>
16 </form>
```

Hasil dari kode diatas terlihat pada gambar 2.14:



Gambar 2.14: Text Input pada Foundation

### Select Menus

Penggunaan `select menus` digunakan apabila *programmer* menginginkan banyak pilihan data dalam satu menu.

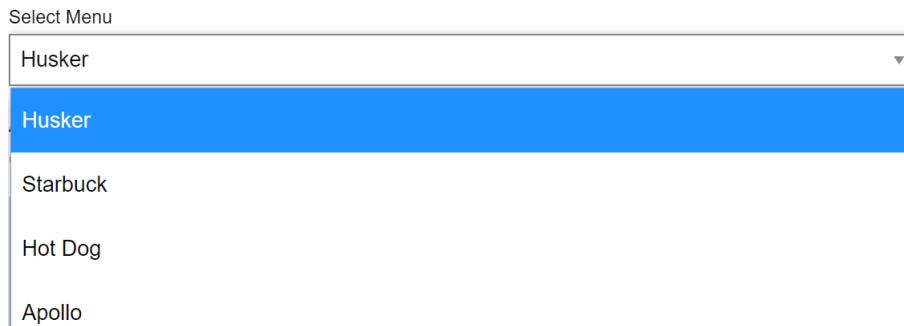
Berikut penggunaan kelas yang dicantumkan pada kode 2.50:

Listing 2.50: Select menus pada foundation 6.

```

1 <label>Select Menu
2   <select>
3     <option value="husker">Husker</option>
4     <option value="starbuck">Starbuck</option>
5     <option value="hotdog">Hot Dog</option>
6     <option value="apollo">Apollo</option>
7   </select>
8 </label>
```

Hasil dari kode diatas terlihat pada gambar 2.15:



Gambar 2.15: Select Menu pada Foundation

### Inline Labels and Buttons

Untuk melampirkan teks tambahan atau kontrol ke kiri atau kanan bidang input, elemen akan berada didalam `.input-grup`, kelas-kelas ini akan ditambahkan ke elemen:

- `.input-grup-field` pada bidang teks.
- `.input-grup-label` pada label teks.
- `.input-grup-button` pada tombol.

Berikut penggunaan kelas yang dicantumkan pada kode 2.51:

Listing 2.51: Select menus pada foundation 6.

```

1 <div class="input-group">
2   <span class="input-group-label">$</span>
3   <input class="input-group-field" type="number">
4   <div class="input-group-button">
5     <input type="submit" class="button" value="Submit">
6   </div>
7 </div>
```

Hasil dari kode diatas terlihat pada gambar 2.16:



Gambar 2.16: Select Menu pada Foundation

### Tabs

Tab semakin banyak digunakan dalam desain web karena programmer dapat menyajikan konten secara seragam. Ini memungkinkan untuk menyimpan banyak dokumen dalam satu *window*. Programmer dapat menggunakan tab sebagai widget navigasi untuk beralih antar konten sehingga menghasilkan tata letak yang sistematis dan bersih.

Berikut penggunaan kelas yang dicantumkan pada kode 2.52:

Listing 2.52: Tabs pada foundation 6.

```

1 <ul class="tabs" data-tabs id="tab_component">
2   <li class="tabs-title"><a href="#pub1">Section 1</a></li>
3   <li class="tabs-title is-active"><a href="#pub2">Section 2</a></li>
4   <li class="tabs-title"><a href="#pub3">Section 3</a></li>
```

```

5   <li class="tabs-title"><a href="#pub4">Section 4</a></li>
6 </ul>
7 <div class="tabs-content" data-tabs-content="tab_component">
8   <div class="tabs-panel" id="pub1">
9     <p>Far far away, behind the word mountains, far from the countries Vokalia
10    and Consonantia, there live the blind texts.</p>
11  </div>
12  <div class="tabs-panel is-active" id="pub2">
13    <p> Separated they live in Bookmarksgrove right at the coast of the
14    Semantics, a large language ocean. </p>
15  </div>
16  <div class="tabs-panel" id="pub3">
17    <p>A small river named Duden flows by their place and supplies it with the
18    necessary regelialia.</p>
19  </div>
20  <div class="tabs-panel" id="pub4">
21    <p>It is a paradisematic country, in which roasted parts of sentences fly
22    into your mouth. </p>
23  </div>
24</div>

```

Hasil dari kode diatas terlihat pada gambar 2.17:



Gambar 2.17: Tabs pada Foundation 6

## Reveal

Modal hanyalah wadah kosong, sehingga *developer* dapat menaruh segala jenis konten di dalamnya, seperti teks ke formulir hingga video ke seluruh *grid*. Untuk membuat modal, tambahkan kelas `.reveal`, atribut `data-reveal`, dan ID yang unik ke dalam *container*.

Berikut implementasi modal yang tertera pada kode 2.53:

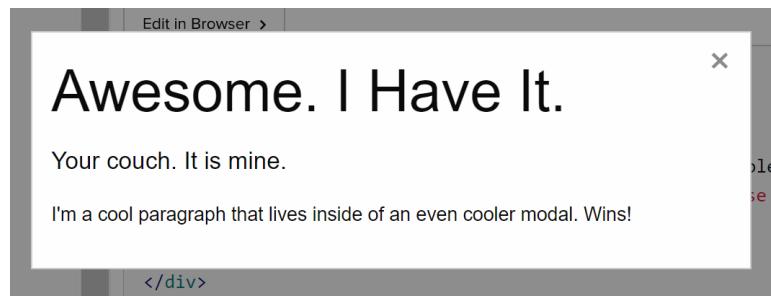
Listing 2.53: Reveal pada foundation 6.

```

1 <div class="reveal" id="exampleModal1" data-reveal>
2   <h1>Awesome. I Have It.</h1>
3   <p class="lead">Your couch. It is mine.</p>
4   <p>I'm a cool paragraph that lives inside of an even cooler modal. Wins!</p>
5   <button class="close-button" data-close aria-label="Close modal" type="button">
6     <span aria-hidden="true">&times;</span>
7   </button>
8 </div>

```

Hasil dari kode diatas terlihat pada gambar 2.18:



Gambar 2.18: Reveal pada Foundation 6

## 2.5 Bootstrap 4

Bootstrap adalah *open source toolkit* yang dikembangkan dengan HTML, CSS, dan JS. Programmer dapat membangun seluruh aplikasi dengan sistem grid responsif, komponen prebuilt yang luas, dan plugin yang dibangun di jQuery. [2]

### 2.5.1 Struktur File

Listing 2.54 merupakan struktur paling dasar dari Bootstrap:

Listing 2.54: Struktur file pada bootstrap 4

```
.bootstrap/
|-- css/
|   |-- bootstrap-grid.css
|   |-- bootstrap-grid.css.map
|   |-- bootstrap-grid.min.css
|   |-- bootstrap-grid.min.css.map
|   |-- bootstrap-reboot.css
|   |-- bootstrap-reboot.css.map
|   |-- bootstrap-reboot.min.css
|   |-- bootstrap-reboot.min.css.map
|   |-- bootstrap.css
|   |-- bootstrap.css.map
|   |-- bootstrap.min.css
|   |-- bootstrap.min.css.map
|-- js/
|   |-- bootstrap.bundle.js
|   |-- bootstrap.bundle.js.map
|   |-- bootstrap.bundle.min.js
|   |-- bootstrap.bundle.min.js.map
|   |-- bootstrap.js
|   |-- bootstrap.js.map
|   |-- bootstrap.min.js
|   |-- bootstrap.min.js.map
.
```

File yang dikompilasi terdiri dari:

- CSS dan JS yang dikompilasi (`bootstrap.*`).
- CSS dan JS yang dikompilasi dan diperkecil (`bootstrap.min.*`).

- Peta sumber (`bootstrap.*.map`) tersedia untuk digunakan dengan alat pengembang browser tertentu.
- File JS yang dibundel (`bootstrap.bundle.js` dan `bootstrap.bundle.min.js` yang diperkecil) termasuk Popper, tetapi bukan jQuery.

### 2.5.2 Sistem Grid

Sistem grid Bootstrap menggunakan *container*, *rows*, dan *columns* untuk tata letak dan penyelarasan konten. Selain itu sistem ini dibangun dengan *flexbox* dan seluruhnya *responsive*.

Dalam kode 2.55 akan dibuat tiga kolom yang memiliki lebar yang sama baik dalam *device small*, *medium*, *large* dan *extra large* menggunakan kelas grid yang sudah ditentukan sebelumnya oleh Bootstrap. Penggunaan `.container` akan membuat kolom berada ditengah halaman.

Listing 2.55: Sistem grid pada bootstrap 4

```

1 <div class="container">
2   <div class="row">
3     <div class="col-sm">
4       One of three columns
5     </div>
6     <div class="col-sm">
7       One of three columns
8     </div>
9     <div class="col-sm">
10      One of three columns
11    </div>
12  </div>
13 </div>
```

Hasil dari kode diatas terlihat pada gambar 2.19:



Gambar 2.19: Grid pada Bootstrap

### Cara Kerja Grid Bootstrap

Secara detil, bootstrap bekerja dengan cara:

- *Container* disediakan agar konten berada ditengah halaman dan mengisi konten tersebut secara horizontal. Penggunaan `.container` untuk menentukan lebar pixel secara responsif atau `.container-fluid` untuk membuat lebar: 100% di semua ukuran *viewport* dan perangkat.
- Sebuah baris akan membungkus kolom - kolom. Setiap kolom akan memiliki *padding* secara horizontal yang disebut *gutter* untuk mengatur jarak antar kolom.
- Penggunaan flexbox akan membuat lebar pada kolom tidak perlu dispesifikasikan. Misalnya empat variabel dari `.col-sm` akan secara otomatis membuat lebar kolom sebesar 25%.
- Kelas kolom menunjukkan jumlah kolom yang ingin digunakan, dengan maksimal 12 kolom per baris. Apabila *developer* menginginkan tiga kolom yang memiliki lebar yang sama maka dapat menggunakan `.col-4`.

- Lebar kolom diatur dalam persentase, sehingga kolom akan memiliki lebar yang berubah-ubah dan ukuran bergantung dengan elemen *parent* nya.

## Pilihan Grid

Bootstrap menggunakan px untuk grid breakpoint dan lebar container. Ini dikarenakan lebar *viewport* ditentukan dengan satuan pixels. Berikut ini tabel 2.20 yang menjelaskan penggunaan kelas grid dalam berbagai perangkat:

	<b>Extra small</b> ≤576px	<b>Small</b> ≥576px	<b>Medium</b> ≥768px	<b>Large</b> ≥992px	<b>Extra large</b> ≥1200px
<b>Max container width</b>	None (auto)	540px	720px	960px	1140px
<b>Class prefix</b>	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
<b># of columns</b>	12				
<b>Gutter width</b>	30px (15px on each side of a column)				
<b>Nestable</b>	Yes				
<b>Column ordering</b>	Yes				

Gambar 2.20: Pilihan kelas grid pada Bootstrap

### 2.5.3 Konten

#### Tabel

Bagian tabel akan menjelaskan penggunaan kelas tabel dengan menggunakan tiga kelas: kelas tabel secara default (`.table`), kelas tabel dengan garis batas (`table-bordered`) dan kelas tabel dengan warna baris berbeda (`table-striped`).

#### Tabel Default

Dengan penggunaan kelas `.table` pada seluruh tag `<table>` maka *style* pada bootstrap akan diterapkan, sehingga setiap tabel yang *nested* akan diatur sesuai dengan *parent* nya.

Gambar 2.21 merupakan hasil penggunaan tabel secara *default*:

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

Gambar 2.21: Tabel default pada Bootstrap

Kode dari gambar diatas tertera pada kode 2.56:

Listing 2.56: Tabel pada bootstrap 4

```

1 <table class="table">
2   <thead>
3     <tr>
4       <th scope="col">#</th>
5       <th scope="col">First</th>
6       <th scope="col">Last</th>
7       <th scope="col">Handle</th>
8     </tr>
9   </thead>
10  <tbody>

```

```

11 <tr>
12   <th scope="row">1</th>
13   <td>Mark</td>
14   <td>Otto</td>
15   <td>@mdo</td>
16 </tr>
17 <tr>
18   <th scope="row">2</th>
19   <td>Jacob</td>
20   <td>Thornton</td>
21   <td>@fat</td>
22 </tr>
23 <tr>
24   <th scope="row">3</th>
25   <td>Larry</td>
26   <td>the Bird</td>
27   <td>@twitter</td>
28 </tr>
29 </tbody>
30 </table>

```

### Tabel dengan Garis Batas

Penggunaan kelas `.table-bordered` akan membuat tabel memiliki garis batas untuk semua sisi didalam tabel dan *cells*.

Gambar 2.22 merupakan hasil penggunaan tabel:

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter

Gambar 2.22: Tabel default pada Bootstrap

Kode dari gambar diatas tertera pada kode 2.57:

Listing 2.57: Kode tabel dengan garis batas pada bootstrap 4

```

1 <table class="table table-bordered">
2   <thead>
3     <tr>
4       <th scope="col">#</th>
5       <th scope="col">First</th>
6       <th scope="col">Last</th>
7       <th scope="col">Handle</th>
8     </tr>
9   </thead>
10  <tbody>
11    <tr>

```

```

12   <th scope="row">1</th>
13   <td>Mark</td>
14   <td>Otto</td>
15   <td>@mdo</td>
16 </tr>
17 <tr>
18   <th scope="row">2</th>
19   <td>Jacob</td>
20   <td>Thornton</td>
21   <td>@fat</td>
22 </tr>
23 <tr>
24   <th scope="row">3</th>
25   <td colspan="2">Larry the Bird</td>
26   <td>@twitter</td>
27 </tr>
28 </tbody>
29 </table>

```

### Tabel dengan Warna Baris Berbeda

Penggunaan kelas `.table-striped` akan membuat tabel memiliki warna baris berbeda batas antara baris genap dan ganjil didalam tag `<tbody>`. Gambar 2.23 merupakan hasil penggunaan tabel:

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

Gambar 2.23: Tabel default pada Bootstrap

Listing 2.58: Kode tabel dengan warna baris berbeda pada bootstrap 4.

```

1 <table class="table table-striped">
2   <thead>
3     <tr>
4       <th scope="col">#</th>
5       <th scope="col">First</th>
6       <th scope="col">Last</th>
7       <th scope="col">Handle</th>
8     </tr>
9   </thead>
10  <tbody>
11    <tr>
12      <th scope="row">1</th>
13      <td>Mark</td>
14      <td>Otto</td>
15      <td>@mdo</td>

```

```

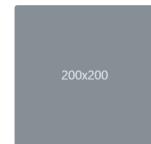
16   </tr>
17   <tr>
18     <th scope="row">2</th>
19     <td>Jacob</td>
20     <td>Thornton</td>
21     <td>@fat</td>
22   </tr>
23   <tr>
24     <th scope="row">3</th>
25     <td colspan="2">Larry the Bird</td>
26     <td>@twitter</td>
27   </tr>
28 </tbody>
29 </table>

```

## Gambar

Gambar dalam Bootstrap akan memiliki sifat *responsive* dengan menerapkan kelas `.img-fluid` serta mengatur lebar gambar dengan properties `max-width: 100%` dan `height: auto`. Sehingga gambar tidak pernah lebih besar dari *parent* nya.

Programmer dapat menyelaraskan (align) sebuah gambar ke kiri atau kanan dengan *helper float classes* atau *text alignment classes*.



Gambar 2.24: Menyelaraskan gambar ke kanan dan kiri pada bootstrap

Kode dari gambar diatas tertera pada kode [2.59](#):

Listing 2.59: Image Align pada bootstrap 4.

```

1 
2 

```

## 2.5.4 Komponen

### Formulir

*Form* pada Bootstrap menyediakan beragam tipe input sesuai dengan kebutuhan *user*. Contohnya penggunaan kelas `email` untuk *input* email atau `number` untuk input berupa angka.

### Form Controls

Programmer dapat membuat form menggunakan kelas `.form-control`. Kelas ini terdiri dari beberapa tag seperti tag `<input>`, `<select>` dan `<textarea>`.

Kode [2.60](#) menjelaskan implementasi dari *form controls*:

Listing 2.60: Form controls pada bootstrap 4.

```

1 <form>
2   <div class="form-group">

```

```

3   <label for="exampleFormControlInput1">Email address</label>
4   <input type="email" class="form-control" id="exampleFormControlInput1"
5     placeholder="name@example.com">
6   </div>
7   <div class="form-group">
8     <label for="exampleFormControlSelect1">Example select</label>
9     <select class="form-control" id="exampleFormControlSelect1">
10    <option>1</option>
11    <option>2</option>
12    <option>3</option>
13    <option>4</option>
14    <option>5</option>
15  </select>
16 </div>
17 <div class="form-group">
18   <label for="exampleFormControlSelect2">Example multiple select</label>
19   <select multiple class="form-control" id="exampleFormControlSelect2">
20     <option>1</option>
21     <option>2</option>
22     <option>3</option>
23     <option>4</option>
24     <option>5</option>
25   </select>
26 </div>
27 <div class="form-group">
28   <label for="exampleFormControlTextarea1">Example textarea</label>
29   <textarea class="form-control" id="exampleFormControlTextarea1"
30     rows="3"></textarea>
</div>
</form>
```

Hasil dari kode diatas terlihat pada gambar 2.25

The screenshot shows a web form with four fields:

- Email address:** A text input field containing "name@example.com".
- Example select:** A single-select dropdown menu showing the option "1".
- Example multiple select:** A multiple-select dropdown menu showing options "2", "3", "4", and "5".
- Example textarea:** A text area with a placeholder "I".

Gambar 2.25: *Forms Basic* pada Bootstrap

### Column Sizing

Bootstrap memungkinkan programmer untuk menempatkan sejumlah .col di dalam baris .row atau .form dengan lebar tertentu. Misalnya ada tiga buah kolom, kolom pertama memiliki lebar 7 dengan menggunakan kelas .col-7 maka dua kolom sisanya akan memiliki lebar yang memenuhi baris tersebut.

Kode 2.60 menjelaskan implementasi dari *column sizing*:

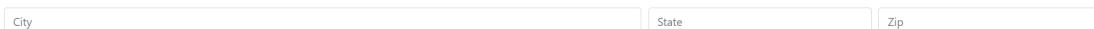
Listing 2.61: Column sizing pada bootstrap 4.

```

1 <form>
2   <div class="form-row">
3     <div class="col-7">
4       <input type="text" class="form-control" placeholder="City">
5     </div>
6     <div class="col">
7       <input type="text" class="form-control" placeholder="State">
8     </div>
9     <div class="col">
10      <input type="text" class="form-control" placeholder="Zip">
11    </div>
12  </div>
13 </form>

```

Hasil dari kode diatas terlihat pada gambar 2.26:

Gambar 2.26: *Column Sizing* pada Bootstrap

## Disabled Forms

Penambahan atribut boolean `disabled` pada sebuah input membuat *user* tidak bisa mengisi data pada *field* tersebut. Untuk non-aktifkan seluruh *field* pada sebuah kolom dapat menambahkan atribut `disabled` pada tag `<fieldset>`.

Kode 2.62 menjelaskan implementasi dari *disabled forms*:

Listing 2.62: Disabled forms pada bootstrap 4.

```

1 <form>
2   <fieldset disabled>
3     <div class="form-group">
4       <label for="disabledTextInput">Disabled input</label>
5       <input type="text" id="disabledTextInput" class="form-control"
6           placeholder="Disabled input">
7     </div>
8     <div class="form-group">
9       <label for="disabledSelect">Disabled select menu</label>
10      <select id="disabledSelect" class="form-control">
11        <option>Disabled select</option>
12      </select>
13    </div>
14    <div class="form-check">
15      <input class="form-check-input" type="checkbox"
16          id="disabledFieldsetCheck" disabled>
17      <label class="form-check-label" for="disabledFieldsetCheck">
18        Can't check this
19      </label>
20    </div>
21    <button type="submit" class="btn btn-primary">Submit</button>
22  </fieldset>
23 </form>

```

Hasil dari kode diatas terlihat pada gambar 2.26:



Gambar 2.27: *Disabled Basic* pada Bootstrap

## Button

Bootstrap memasukan beberapa button dengan *style* yang sudah didefinisikan sebelumnya, membuat setiap button akan memiliki makna nya sendiri.

Kode 2.63 menjelaskan implementasi dari *button*:

Listing 2.63: Button pada bootstrap 4.

```

1 <button type="button" class="btn btn-primary">Primary</button>
2 <button type="button" class="btn btn-secondary">Secondary</button>
3 <button type="button" class="btn btn-success">Success</button>
4 <button type="button" class="btn btn-danger">Danger</button>
5 <button type="button" class="btn btn-warning">Warning</button>
6 <button type="button" class="btn btn-info">Info</button>
7 <button type="button" class="btn btn-light">Light</button>
8 <button type="button" class="btn btn-dark">Dark</button>
9
10 <button type="button" class="btn btn-link">Link</button>
```

Hasil dari kode diatas terlihat pada gambar 2.28:



Gambar 2.28: *Button* pada Bootstrap

## Button with Dropdowns

Kode 2.64 menjelaskan implementasi dari *button with dropdowns*:

Listing 2.64: Button dropdown pada bootstrap 4.

```

1 <div class="input-group mb-3">
2   <div class="input-group-prepend">
3     <button class="btn btn-outline-secondary dropdown-toggle" type="button"
4           data-toggle="dropdown"
5           aria-haspopup="true" aria-expanded="false">Dropdown</button>
6     <div class="dropdown-menu">
7       <a class="dropdown-item" href="#">Action</a>
8       <a class="dropdown-item" href="#">Another action</a>
9       <a class="dropdown-item" href="#">Something else here</a>
10      <div role="separator" class="dropdown-divider"></div>
11      <a class="dropdown-item" href="#">Separated link</a>
12    </div>
13  </div>
```

```

13   <input type="text" class="form-control" aria-label="Text input with dropdown
14     button">
15   </div>
16
17 <div class="input-group">
18   <input type="text" class="form-control" aria-label="Text input with dropdown
19     button">
20   <div class="input-group-append">
21     <button class="btn btn-outline-secondary dropdown-toggle" type="button"
22       data-toggle="dropdown"
23       aria-haspopup="true" aria-expanded="false">Dropdown</button>
24     <div class="dropdown-menu">
25       <a class="dropdown-item" href="#">Action</a>
26       <a class="dropdown-item" href="#">Another action</a>
27       <a class="dropdown-item" href="#">Something else here</a>
28       <div role="separator" class="dropdown-divider"></div>
29       <a class="dropdown-item" href="#">Separated link</a>
  </div>
</div>
</div>

```

Hasil dari kode diatas terlihat pada gambar 2.29:



Gambar 2.29: Tombol *dropdown* pada Bootstrap

## Badge

Kelas badge dan .badge-\* di dalam sebuah <a> akan memberikan badge yang dapat diberi atribut *hover* dan *focus*. Kode 2.65 menjabarkan implementasi dari komponen *badge*:

Listing 2.65: Badge pada bootstrap 4.

```

1 <a href="#" class="badge badge-primary">Primary</a>
2 <a href="#" class="badge badge-secondary">Secondary</a>
3 <a href="#" class="badge badge-success">Success</a>
4 <a href="#" class="badge badge-danger">Danger</a>
5 <a href="#" class="badge badge-warning">Warning</a>
6 <a href="#" class="badge badge-info">Info</a>
7 <a href="#" class="badge badge-light">Light</a>
8 <a href="#" class="badge badge-dark">Dark</a>

```

Hasil dari kode diatas terlihat pada gambar 2.30:



Gambar 2.30: Badge pada Bootstrap

## Card

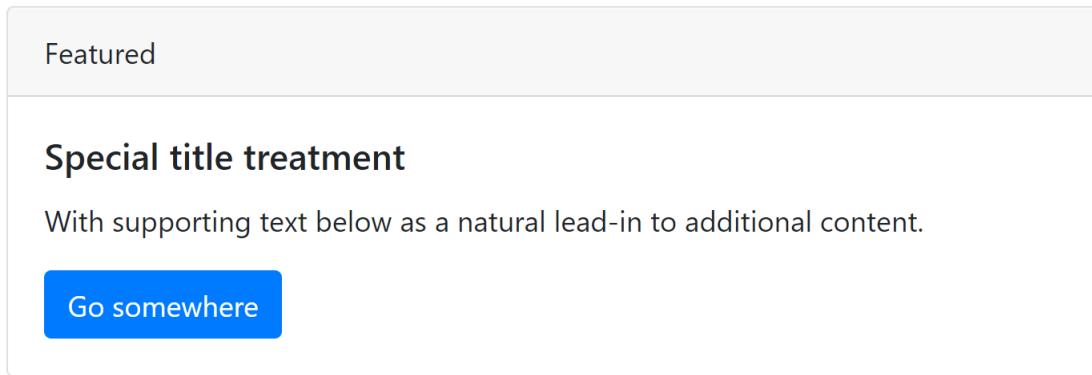
Card adalah kontainer konten yang fleksibel dan bisa diatur lebarnya. Sebuah card memiliki sebuah *headers* dan *footers*.

Berikut ini implementasi yang terlihat pada kode [2.66](#):

Listing 2.66: Card pada bootstrap 4.

```
1 <div class="card">
2   <div class="card-header">
3     Featured
4   </div>
5   <div class="card-body">
6     <h5 class="card-title">Special title treatment</h5>
7     <p class="card-text">With supporting text below as
8       a natural lead-in to additional content.</p>
9     <a href="#" class="btn btn-primary" card>Go somewhere</a>
10    </div>
11 </div>
```

Hasil dari kode diatas terlihat pada gambar [2.31](#):



Gambar 2.31: Card pada Bootstrap

## Navigation Bar

Navbar pada Bootstrap terdiri dari beberapa sub-komponen yang bisa digunakan sesuai dengan kebutuhan:

- `.navbar-brand` : Komponen untuk menampilkan nama perusahaan, nama produk atau nama proyek.
- `.navbar-nav` : Komponen untuk membuat navigasi memiliki lebar yang memenuhi layar.
- `.navbar-toggler` : Komponen yang digunakan bersamaan dengan plugin untuk membuat efek jatuh dan perilaku navigasi lainnya.

- `.form-inline` : Komponen untuk pengaturan formulir dan aksi.
- `.collapse.navbar-collapse` : Komponen untuk mengelompokkan dan menyembunyikan `navigation bar` dengan sebuah breakpoint induknya.

Berikut ini implementasi yang terlihat pada kode 2.67, dimana semua sub-komponen yang termasuk dalam navigation bar. Navbar mengimplementasikan tema `light-themed` yang secara otomatis menyembunyikan menu pada breakpoint `lg`

Listing 2.67: Navigation bar pada bootstrap 4.

```

1 <nav class="navbar navbar-expand-lg navbar-light bg-light">
2   <a class="navbar-brand" href="#">Navbar</a>
3   <button class="navbar-toggler" type="button" data-toggle="collapse"
4     data-target="#navbarSupportedContent"
5     aria-controls="navbarSupportedContent" aria-expanded="false"
6     aria-label="Toggle navigation">
7     <span class="navbar-toggler-icon"></span>
8   </button>
9
10  <div class="collapse navbar-collapse" id="navbarSupportedContent">
11    <ul class="navbar-nav mr-auto">
12      <li class="nav-item active">
13        <a class="nav-link" href="#">Home <span
14          class="sr-only">(current)</span></a>
15      </li>
16      <li class="nav-item">
17        <a class="nav-link" href="#">Link</a>
18      </li>
19      <li class="nav-item dropdown">
20        <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown"
21          role="button" data-toggle="dropdown" aria-haspopup="true"
22          aria-expanded="false">
23          Dropdown
24        </a>
25        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
26          <a class="dropdown-item" href="#">Action</a>
27          <a class="dropdown-item" href="#">Another action</a>
28          <div class="dropdown-divider"></div>
29          <a class="dropdown-item" href="#">Something else here</a>
30        </div>
31      </li>
32      <li class="nav-item">
33        <a class="nav-link disabled" href="#">Disabled</a>
34      </li>
35    </ul>
36    <form class="form-inline my-2 my-lg-0">
37      <input class="form-control mr-sm-2" type="search" placeholder="Search"
38        aria-label="Search">
39      <button class="btn btn-outline-success my-2 my-sm-0"
40        type="submit">Search</button>
41    </form>
42  </div>
43</nav>
```

Hasil dari kode diatas terlihat pada gambar [2.32](#):

Navbar Home Link Dropdown ▾ Disabled

Search

Gambar 2.32: Navigation Bar pada Bootstrap

## Modal

Bagaimana Modal bekerja :

- Modal dibangun dengan HTML, CSS dan Javascript.
- Menekan modal "backdrop" otomatis menutup komponen modal.
- Bootstrap hanya mendukung satu modal dalam sebuah window pada satu waktu. Penggunaan modal yang bercabang dalam Bootstrap dipercaya memberikan *user experience* yang buruk.
- Modal menggunakan `position: fixed` yang diletakkan pada posisi teratas dalam kode agar terhindar dari *bug* yang disebabkan elemen lain yang memiliki posisi *fixed*.

Komponen modal terdiri dari modal header, modal body dan modal footer (opsional). Berikut ini implementasi yang terlihat pada kode [2.68](#):

Listing 2.68: Modal bar pada bootstrap 4.

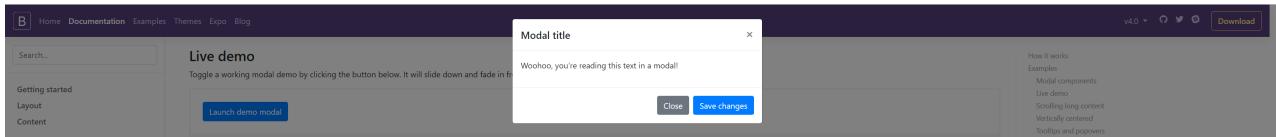
```

1 <!-- Button trigger modal -->
2 <button type="button" class="btn btn-primary" data-toggle="modal"
3   data-target="#myModal">
4   Launch demo modal
5 </button>
6
7 <!-- Modal -->
8 <div class="modal fade" id="exampleModal" tabindex="-1" role="dialog"
9   aria-labelledby="exampleModalLabel" aria-hidden="true">
10  <div class="modal-dialog" role="document">
11    <div class="modal-content">
12      <div class="modal-header">
13        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>
14        <button type="button" class="close" data-dismiss="modal"
15          aria-label="Close">
16          <span aria-hidden="true">&times;</span>
17        </button>
18      </div>
19      <div class="modal-body">
20        ...
21      </div>
22      <div class="modal-footer">
23        <button type="button" class="btn btn-secondary"
24          data-dismiss="modal">Close</button>
25        <button type="button" class="btn btn-primary">Save changes</button>
26      </div>
27    </div>
28  </div>

```

```
25   </div>
26 </div>
```

Hasil dari kode diatas terlihat pada gambar 2.33:



Gambar 2.33: Modal pada Bootstrap

## Ikon

Bootstrap tidak memiliki *library* ikon secara *default*, sehingga ikon yang digunakan diambil dari **Font Awesome**. Penggunaan ikon dengan menggunakan tag `<i>` yang disertai dengan kelas `fa` (font-awesome). Berikut ini implementasi yang terlihat pada kode 2.69:

Listing 2.69: Ikon pada bootstrap 4.

```
1 <i class="fa fa-coffee"></i>
```

Hasil dari kode diatas terlihat pada gambar 2.34:



Gambar 2.34: Ikon *Coffee* pada Font Awesome

## Alert

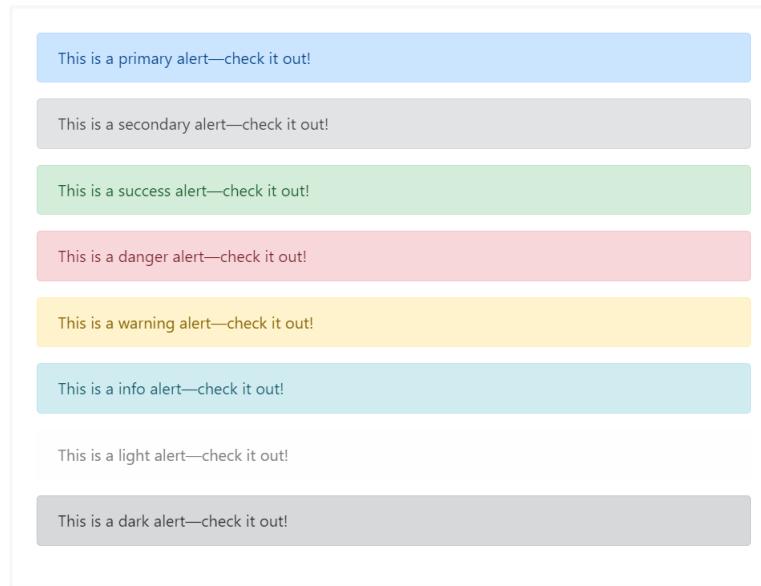
Alert menyediakan pesan umpan balik untuk user untuk berbagai tipe pesan peringatan yang tersedia. Untuk gaya yang sesuai *developer* dapat menggunakan delapan kelas yang tersedia. Berikut ini implementasi yang terlihat pada kode 2.70:

Listing 2.70: Alert pada bootstrap 4.

```
1 <div class="alert alert-primary" role="alert">
2   This is a primary alertcheck it out!
3 </div>
4 <div class="alert alert-secondary" role="alert">
5   This is a secondary alertcheck it out!
6 </div>
7 <div class="alert alert-success" role="alert">
8   This is a success alertcheck it out!
9 </div>
10 <div class="alert alert-danger" role="alert">
11   This is a danger alertcheck it out!
12 </div>
13 <div class="alert alert-warning" role="alert">
14   This is a warning alertcheck it out!
15 </div>
16 <div class="alert alert-info" role="alert">
17   This is a info alertcheck it out!
18 </div>
19 <div class="alert alert-light" role="alert">
```

```
20 This is a light alert—check it out!
21 </div>
22 <div class="alert alert-dark" role="alert">
23   This is a dark alert—check it out!
24 </div>
```

Hasil dari kode diatas terlihat pada gambar 2.35:



Gambar 2.35: Alert pada Bootstrap

## 2.6 DateTimePicker

DateTimePicker dengan menggunakan jQuery untuk memilih tanggal dan waktu pada bagian forms.

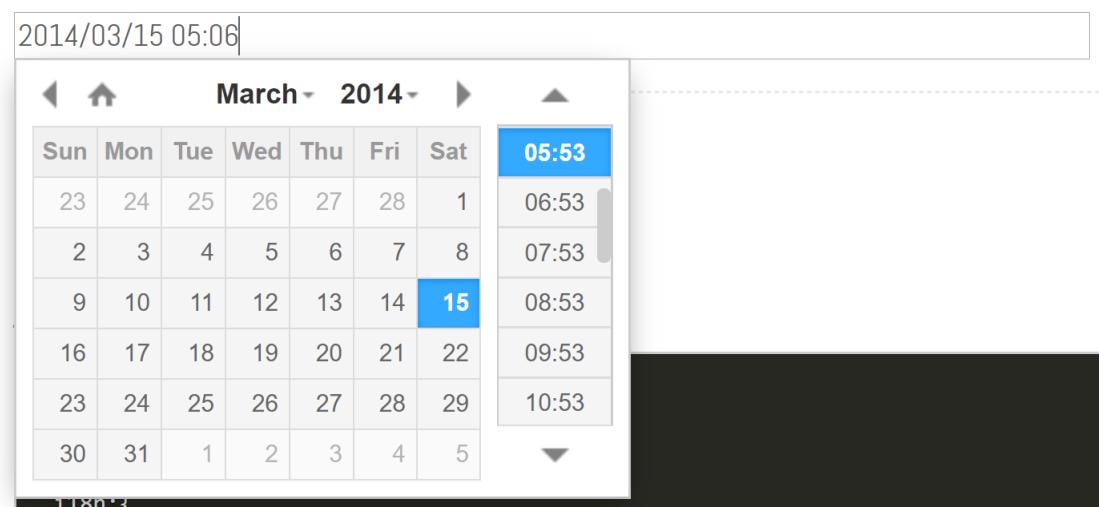
### Inline DateTimePicker

Penggunaan plugin ini, memungkinkan *users* untuk memilih tanggal dan waktu secara bersamaan. Penggunaannya dalam kode HTML tertera pada kode 2.71:

Listing 2.71: Kode HTML pada plugin.

```
1 <input id="datetimetypepicker" type="text" >
```

Hasil dari kode diatas terlihat pada gambar 2.36:



Gambar 2.36: Datetimepicker pada Bootstrap

Kemudian penggunaan dalam kode Javascript tertera pada kode [2.72](#):

Listing 2.72: Kode Inline DateTimePicker di jQuery.

```
1  jQuery('#datetimepicker').datetimepicker();
```

## BAB 3

# ANALISIS WEBSITE BLUETAPE DENGAN FOUNDATION 6

Bagian ini akan menganalisis kelas yang digunakan dalam website BlueTape dengan menggunakan *framework* Foundation 6. Pertama pengembang akan melihat akses apa saja yang diperlukan untuk menjalankan website pada komputer lokal. Kedua pengembang akan melihat keseluruhan kode pada seluruh tampilan website dan file apa yang perlu diubah nantinya pada saat konversi dengan framework Bootstrap 4.

### 3.1 Akses bagi Admin

Sebelum mengakses website, *developer* perlu menambahkan email admin agar dapat mengakses keseluruhan fitur dalam BlueTape. Kemudian untuk melihat data, akses database baru akan dibuat pada komputer lokal.

#### 3.1.1 Akses Email

Daftar email admin yang dapat mengakses keseluruhan fitur dalam website terletak di `www/application/config/modules.php`. Keseluruhan email sebelumnya sudah diregistrasi di aplikasi Google API Console.

File ini berisi tiga macam modul yaitu :

- Modul yang mengarakan ke halaman website.
- Modul yang menentukan halaman website yang bisa di akses berdasarkan *role user*.
- Modul yang medefinisikan *role* berdasarkan email.

Kode 3.1 menjabarkan isi dari modul-modul diatas.

Listing 3.1: Sunting akses email

```
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
15 $config['modules'] = array(
16   'TranskripRequest' => array('root', 'mahasiswa.ftis'),
17   'TranskripManage' => array('root', 'tu.ftis'),
18   'PerubahanKuliahRequest' => array('root', 'staf.unpar'),
19   'PerubahanKuliahManage' => array('root', 'tu.ftis'),
20   'EntriJadwalDosen' => array('root', 'dosen.informatika' ),
21   'LihatJadwalDosen' => array('root', 'mahasiswa.informatika', 'dosen.informatika')
22 );
23
24 $config['roles'] = array(
25   'root' => array('pascal@unpar.ac.id', 'shao.wei@unpar.ac.id',
26     'amihapsahapsa@gmail.com'),
27   'tu.ftis' => array('shao.wei@unpar.ac.id', 'pranyoto@unpar.ac.id',
28     'walip@unpar.ac.id', 'dwina@unpar.ac.id'),
29   'mahasiswa.ftis' => '(7[123]\d{5})|
(20[1-9] [0-9]7[123] [0-9]{4})|
(61[678] [0-9]{7})@student\\.unpar\\.ac\\.id',
30   'staf.unpar' => '.+@unpar\\.ac\\.id',
31   'dosen.informatika' => array ('cheni@unpar.ac.id', 'mariskha@unpar.ac.id',
32     'anung@unpar.ac.id', 'moertini@unpar.ac.id', 'natalia@unpar.ac.id',
33     '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', 'pascal@unpar.ac.id', 'rosad5@unpar.ac.id',
     'vania.natali@unpar.ac.id', 'kristopher.h@unpar.ac.id',
     'raymond.chandra@unpar.ac.id', 'keenan.leman@unpar.ac.id'),
     'mahasiswa.informatika' => '73\d{5}@student\\.unpar\\.ac\\.id'
);

```

### 3.1.2 Akses Database

Data akan disimpan dalam database lokal, sehingga *developer* akan mengatur *base URL* yang terletak pada `www/application/config/config.php`. Kode 3.2 mengatur URL pada komputer lokal.

Listing 3.2: Setting database lokal

```

1 $config['base_url'] = 'http://127.0.0.1/';

```

## 3.2 Kelas dan file yang dipakai pada tampilan Website

Bagian ini akan menjelaskan detil penggunaan kelas pada website dengan menggunakan framework Foundation 6. Penjelasan akan berupa *screenshot* halaman website beserta penjelasan kelas yang dipakai.

### 3.2.1 Penggunaan *framework* Foundation 6 pada Website

Semua halaman website terhubung ke library Foundation 6 dengan mengakses dua file PHP utama yaitu `script.foundation.php` untuk memuat file javascript dan `head_loggedin.php` untuk me-

muat file css dan jquery.

### File untuk mengakses file Javascript

Kode [3.3](#) merupakan file dari `www/application/views/templates/script.foundation.php` untuk menambahkan file `.js` dari Foundation 6 dan `plugin xdan-datetimepicker`:

Listing 3.3: Penambahan library js

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?>
4 <script src="/public/js/vendor/jquery.min.js"></script>
5 <script src="/public/js/vendor/what-input.min.js"></script>
6 <script src="/public/js/foundation.min.js"></script>
7 <script src="/public/js/app.js"></script>
8 <script src="/public/lib/xdan-datetimepicker/jquery.datetimepicker.full.min.js"></script>
```

### File untuk mengakses file CSS

Kode [3.4](#) merupakan file dari `www/application/views/templates/head_loggedin.php` untuk menambahkan file `.css` dari Foundation 6 dan `plugin xdan-datetimepicker`:

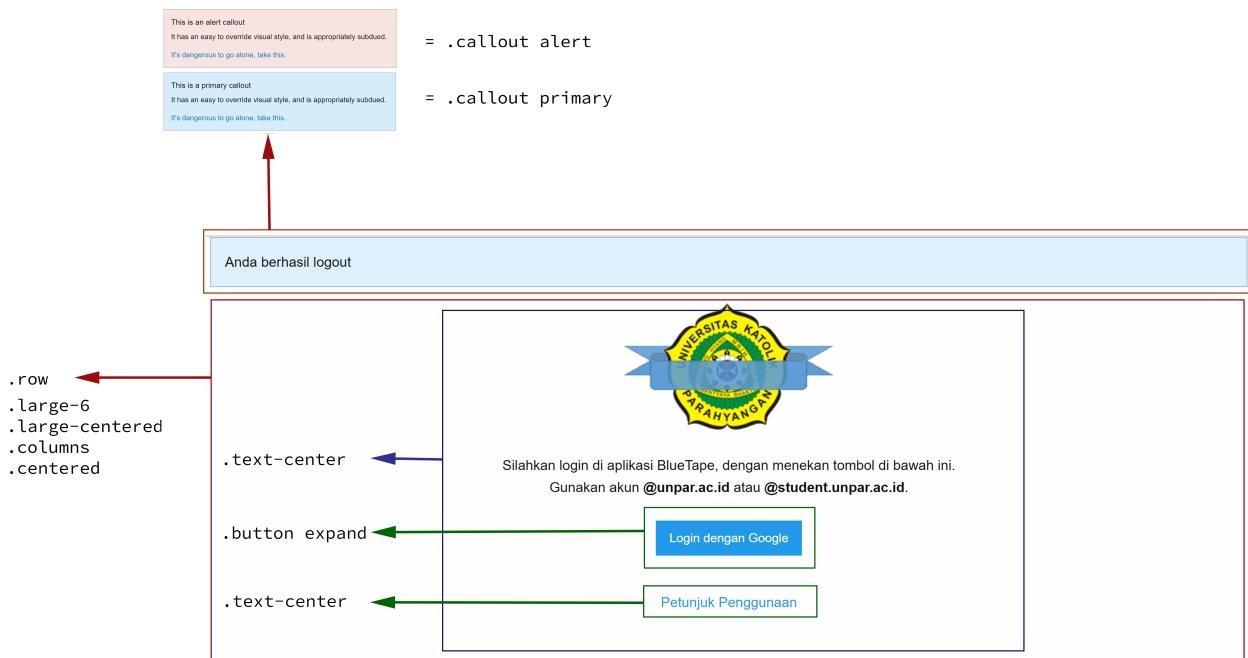
Listing 3.4: Penambahan library css

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3 ?><head>
4   <meta charset="utf-8" />
5   <meta http-equiv="x-ua-compatible" content="ie=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7   <title><?= $this->config->item('module-names')[\$currentModule] ?></title>
8   <link rel="stylesheet" href="/public/css/foundation.css" />
9   <link rel="stylesheet" href="/public/css/foundation-icons.css" />
10  <link rel="stylesheet" href="/public/css/app.css" />
11  <link rel="stylesheet" href="/public/lib/xdan-datetimepicker/jquery.datetimepicker.min.css" />
12 </head>
```

### 3.2.2 Halaman Login

Gambar [3.1](#) adalah halaman pertama yang diakses *user*, penjelasan untuk setiap tombol yang ada direferensikan pada [2.1.1](#).



Gambar 3.1: Analisis halaman login.

Kelas yang digunakan pada halaman login (gambar 3.1) tertera pada tabel 3.1:

Tabel 3.1: Penggunaan kelas pada halaman login.

Kelas	Penggunaan
.row, .large-6, .column	Bagian konten akan terletak di tengah halaman utama yang memiliki lebar 6 grid.
.text-center	Kalimat "Silahkan login .." dan link "Petunjuk Bagian" akan terletak ditengah container.
.button expand	Tombol "Login dengan Google" akan memiliki <i>styling</i> grid yang lebarnya memenuhi panjang grid nya.

Pada bagian atas halaman login terdapat notifikasi bagi user saat proses login dan logout. Kelas yang digunakan tertera pada tabel 3.2.

Tabel 3.2: Penggunaan kelas pada bagian notifikasi user.

Kelas	Penggunaan
.callout primary	Notifikasi berwarna biru menandakan user berhasil <i>login</i> .
.callout alert	Notifikasi berwarna merah menandakan user berhasil <i>logout</i> .

### 3.2.3 Menu Navigasi

Komponen menu navigasi digunakan pada keseluruhan tampilan website, bentuk menu akan menyesuaikan sesuai dengan layar dimana website diakses.

Berikut ini gambar 3.2 merupakan hasil penggunaan dari implementasi menu navigasi:



Gambar 3.2: Analisis menu navigasi di layar *large*.

Tabel 3.3 menjelaskan kelas-kelas yang digunakan pada menu navigasi di layar *large*.

Tabel 3.3: Penggunaan kelas pada menu navigasi di layar *large*.

Kelas	Penggunaan
.top-bar	Menu akan terletak pada bagian atas dari halaman.
.menu	Kelas yang membuat daftar judul halaman website menjadi menu.
.menu-active	Kelas untuk menandakan menu yang sedang dipilih user.
.menu-text	Kelas untuk menyelaraskan nama menu berbentuk teks agar sejajar dengan <i>navigation bar</i> .
.top-bar-left	Kelas yang mengatur daftar menu disebelah kiri.
.top-bar-right	Kelas yang mengatur daftar menu disebelah kanan.

Lalu untuk layar *medium* dan *small* menggunakan komponen **Advanced Layout** pada gambar 3.3 dimana daftar halaman website yang ada pada menu berada di mode *hide* dan digantikan oleh ikon menu.



Gambar 3.3: Analisis menu navigasi di layar *medium* dan *small*.

Komponen **Advanced Layout** terdiri dari kelas - kelas yang dijabarkan dalam tabel 3.4.

Tabel 3.4: Penggunaan kelas pada menu navigasi di layar *medium* dan *small*.

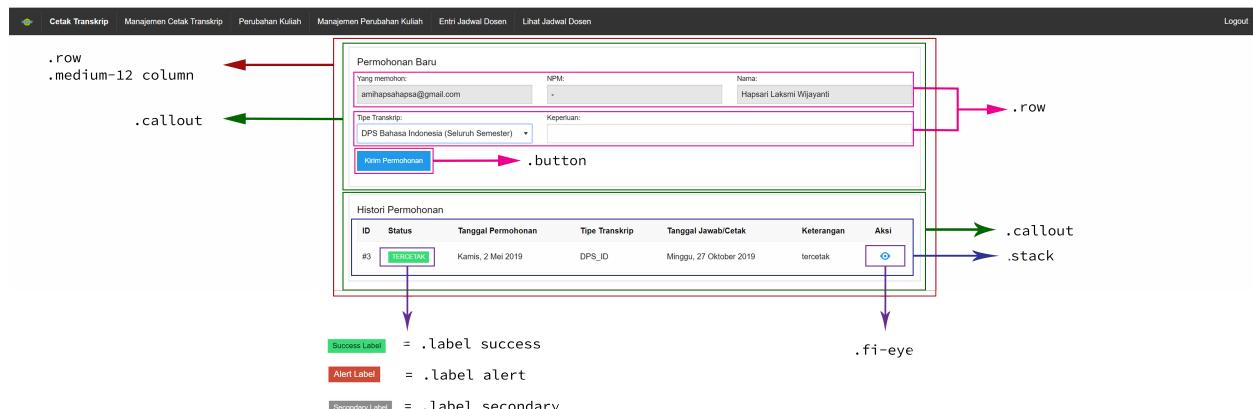
Kelas	Penggunaan
.title-bar	Kelas yang mendefinisikan penggunaan menu dari <i>Advanced Layout</i> .
.title-bar-title	Kelas yang mendefinisikan ikon BlueTape sebagai judul website.
.menu-icon	Kelas untuk menampilkan icon menu.
data-responsive-toggle	Atribut untuk membuat daftar menu responsif.
data-toggle	Atribut ini akan memanggil data yang disimpan dalam <code>data-toggle</code> .
data-hide-for	Atribut yang mengatur kapan menu navigasi akan responsif.

### 3.2.4 Halaman Permintaan Cetak Transkrip

#### Halaman Utama

Halaman permintaan cetak transkrip terdiri dari dua konten yaitu: "Permohonan Baru" dan "Histori Permohonan". Setiap konten akan dipisahkan dengan border. Pada konten "Permohonan Baru" terdiri dari sebuah form yang berisi *field* yang terdiri dari 2 *rows* dan sebuah tombol "Kirim Permohonan". Lalu konten "Histori Permohonan" terdiri dari sebuah tabel dimana data akan dipanggil dari *database*. dan kolom 'Aksi' mennggarahkan *user* ke sebuah modal lihat permintaan transkrip yang direpresentasikan dengan ikon mata.

Berikut ini gambar 3.4 merupakan hasil penggunaan dari implementasi menu navigasi:



Gambar 3.4: Analisis halaman cetak transkrip.

Kelas yang digunakan dalam konten permohonan baru tertera pada tabel 3.5.

Tabel 3.5: Penggunaan kelas pada halaman utama cetak transkrip.

Kelas	Penggunaan
.row	Kelas ini digunakan untuk menyimpan container menjadi satu <i>cell</i> dan mengatur <i>field-form</i> menjadi satu baris.
.medium-12 column	Keseluruhan konten akan memiliki lebar 12 grid pada layar medium.
.button	Jenis kelas yang digunakan pada tombol ‘Kirim Permohonan’
.callout	Untuk membuat border yang memisahkan konten permohonan baru dan histori permohonan.
.fi-eye	Kelas pada font awesome untuk link ke modal lihat.
.stack	Tabel lihat cetak transkrip dengan lebar yang dapat disesuaikan berdasarkan ukuran layar.

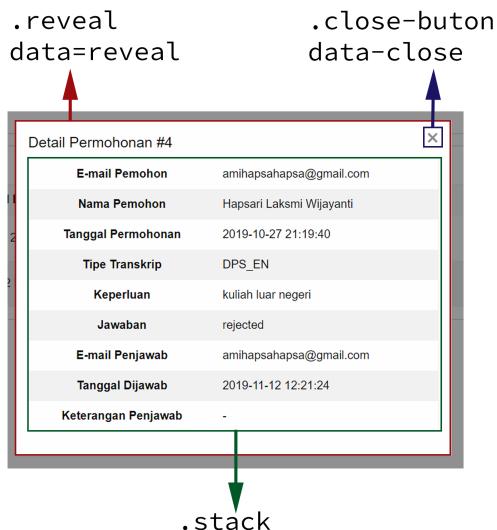
Tabel 3.6 menjabarkan pada kolom ‘status’ digunakan highlight yang terdiri dari tiga warna. Kelas yang digunakan adalah .label.

Tabel 3.6: Penggunaan komponen label pada kolom status.

Kelas	Penggunaan
.label success	Label hijau menandakan transkrip yang telah tercetak.
.label alert	Label merah menandakan yang gagal tercetak.
.label secondary	Label abu-abu menandakan transkrip belum tercetak.

### Modal: Lihat

Gambar 3.5 implementasi apabila user memilih ikon mata, maka website akan mengarahkan ke modal lihat permintaan cetak transkrip.



Gambar 3.5: Analisis modal lihat permintaan cetak transkrip.

Modal akan menampilkan detail permohonan cetak transkrip berdasarkan ID yang tercatat. Kelas yang digunakan pada modal lihat tertera pada tabel 3.7.

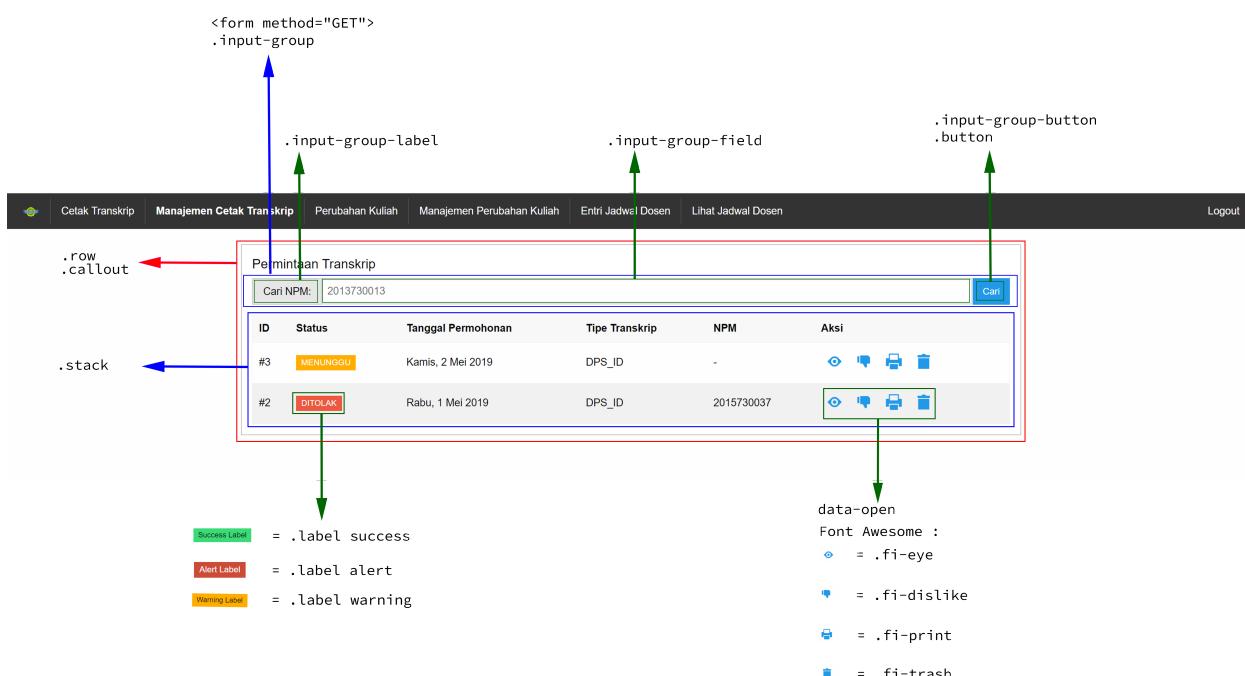
Tabel 3.7: Penggunaan kelas pada modal lihat permintaan transkrip.

Kelas	Penggunaan
.reveal data-reveal	Kelas yang menampilkan modal lihat cetak transkrip.
.close-button data-close	Kelas yang memungkinkan <i>user</i> untuk menutup halaman modal.
.stack	Kelas untuk membuat tabel detail permohonan.

### 3.2.5 Halaman Manajemen Cetak Transkrip

Halaman ini terdiri dari satu konten yaitu "Permintaan Transkrip" tertera pada gambar 3.6. Bagian form pencarian NPM menggunakan *group field*, dimana nama form , isi form dan tombol merupakan satu kesatuan. Lalu dibawah form terdapat tabel yang menampilkan data permintaan cetak transkrip.

#### Halaman Utama



Gambar 3.6: Analisis manajemen cetak transkrip.

Kelas yang digunakan dalam halaman ini tertera pada tabel 3.8:

Tabel 3.8: Penggunaan kelas pada halaman manajemen cetak transkrip.

Kelas	Penggunaan
<b>.row</b>	Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa <i>field-form</i> menjadi satu baris.
<b>.callout</b>	Untuk membuat border yang memisahkan konten permohonan baru dan histori permohonan.
<b>.stack</b>	Jenis tabel yang digunakan tabel histori permohonan, sehingga pada layar medium lebar tabel akan menyesuaikan kan tiap <i>cell</i> ditampilkan secara bertumpuk.

Pada tabel permintaan transkrip terdapat status permintaan transkrip yang di *highlight* menggunakan warna yang berbeda - beda. Kelas yang digunakan tertera pada tabel 3.9.

Tabel 3.9: Penggunaan komponen label pada kolom status histori permohonan.

Kelas	Penggunaan
.label success	Label untuk transkrip yang telah tercetak.
.label alert	Label untuk transkrip yang gagal tercetak.
.label warning	Label untuk transkrip yang menunggu untuk tercetak.

Untuk kolom aksi, terdapat empat kelas dari Font Awesome. Setiap ikon mengarahkan *user* ke halaman modal dan halaman *preview* cetak transkrip. Kelas - kelas yang digunakan tertera pada tabel 3.10.

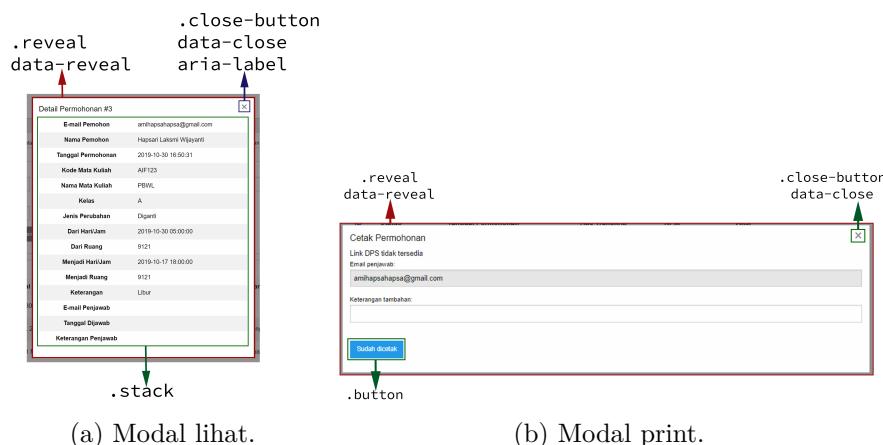
Tabel 3.10: Penggunaan kelas font awesome pada kolom aksi.

Kelas	Penggunaan
fi-eye	<i>Hypertext reference</i> berbentuk ikon yang mengarahkan ke modal lihat transkrip.
fi-dislike	<i>Hypertext reference</i> berbentuk ikon yang mengarahkan ke modal tolak cetak transkrip.
fi-print	<i>Hypertext reference</i> berbentuk ikon yang mengarahkan ke modal print cetak transkrip.
fi-trash	<i>Hypertext reference</i> berbentuk ikon yang mengarahkan ke modal hapus permintaan transkrip.

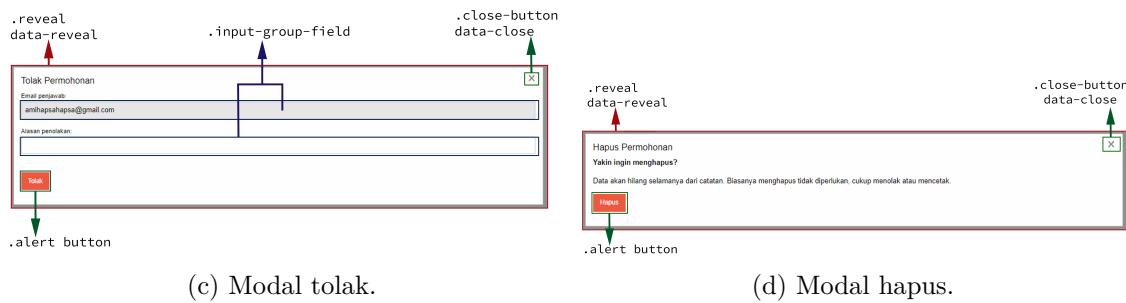
### Modal: Lihat, Tolak, Print dan Hapus

Bagian modal pada halaman manajemen cetak transkrip terdiri dari empat bagian.

Gambar 3.7 menunjukkan kelas yang digunakan untuk setiap modal:



Kelas yang digunakan untuk seluruh modal tertera pada tabel 3.11.



Gambar 3.7: Analisis modal pada halaman manajemen cetak transkrip.

Tabel 3.11: Penggunaan kelas pada komponen modal manajemen cetak transkrip.

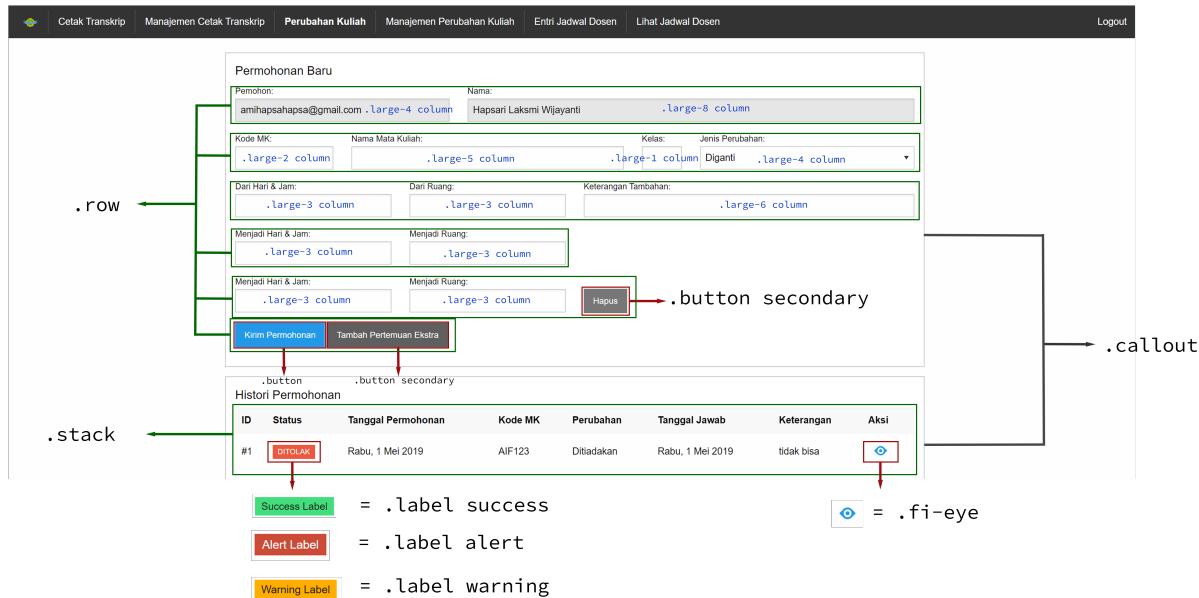
Kelas	Penggunaan
.reveal data-reveal	Kelas dan atribut untuk membuat komponen modal.
.close-button	Memungkinkan user untuk menutup modal yang telah terbuka.
data-close	
aria-label	
.stack	Membuat tabel dalam modal lihat cetak transkrip.
.button	Tombol "Sudah dicetak" berwarna biru untuk mengirimkan status permintaan sudah dicetak.
.alert button	Tombol "Tolak" dan "Hapus" berwarna merah untuk menolak dan menghapus permintaan transkrip
.input-group-field	Membuat <i>form</i> pada kolom "Email Penjawab" dan "Alasan Penolakan" pada modal hapus permintaan traskrip.

### 3.2.6 Halaman Permintaan Perubahan Kuliah

Halaman permintaan perubahan kuliah terdiri dari dua konten: "Permohonan Baru" dan "Histori Permohonan" tertera pada gambar 3.8.

#### Halaman Utama

Setiap konten dipisahkan oleh border dalam Foundation disebut **callout**. Pada "Permohonan Baru" terdapat sebuah form yang berisi field-fiel dengan lebar yang berbeda - beda dan dikelompokan pada beberapa baris. Terdapat tiga tombol yaitu tombol biru "Kirim Permohonan" dan tombol abu-abu "Tambah pertemuan Ektra". Konten "Histori Permohonan" berisi tabel bergaris yang menampilkan data histori permohonan.



Gambar 3.8: Analisis halaman permintaan perubahan kuliah.

Penjelasan detil mengenai kelas - kelas dari gambar diatas terdapat pada tabel 3.12

Tabel 3.12: Penggunaan kelas pada halaman permintaan perubahan kuliah.

Kelas	Penggunaan
.row	Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa <i>field-form</i> menjadi satu baris.
.large-* column	Mendefinisikan lebar grid untuk masing - masing <i>field</i> pada <i>form</i> .
.callout	Untuk membuat border yang memisahkan konten permohonan baru dan histori permohonan.
.stack	Jenis tabel yang digunakan tabel histori permohonan, sehingga pada layar medium tabel akan tersusun secara bertumpuk.
.fi-eye, data-open	Ikon untuk menuju modal lihat detail permohonan berdasarkan ID.

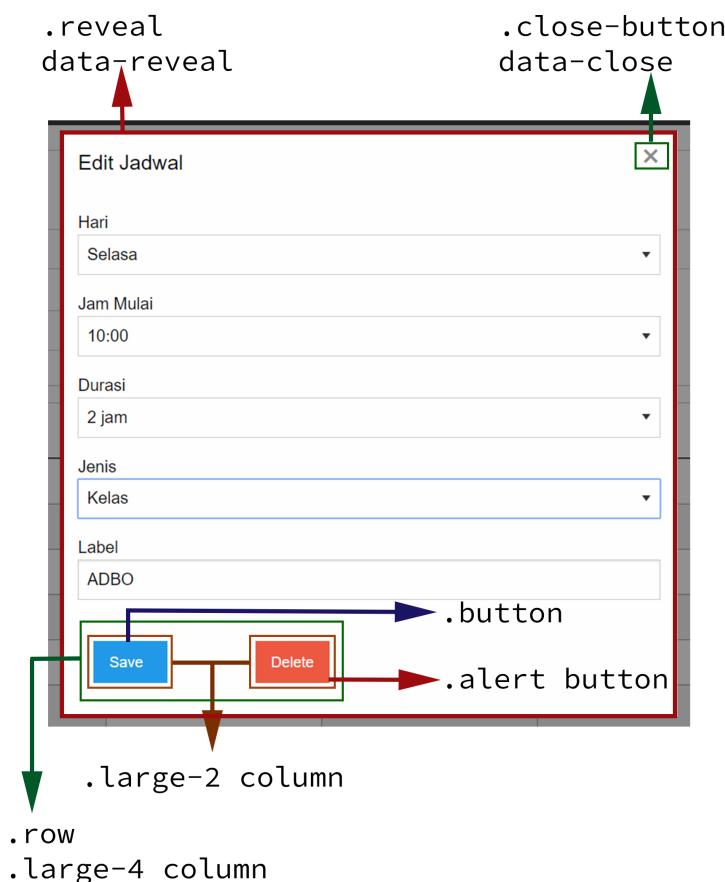
Pada tabel 3.13 kolom 'Status' mendefinisikan hasil permohonan perubahan kuliah, tiga jenis status menggunakan tiga jenis kelas yang berbeda-beda yaitu:

Tabel 3.13: Penggunaan komponen label pada kolom status histori permohonan.

Kelas	Penggunaan
.label success	Label untuk perubahan kuliah berhasil.
.label alert	Label untuk perubahan kuliah gagal.
.label warning	Label untuk perubahan kuliah diproses.

### Modal: Lihat

Kolom 'Aksi' berisi ikon mata yang akan mereferensikan ke modal lihat pada data permohonan perubahan kuliah. Modal lihat tertera pada gambar 3.9:



Gambar 3.9: Analisis modal lihat.

Kelas yang digunakan dalam modal ditampilkan pada tabel 3.14.

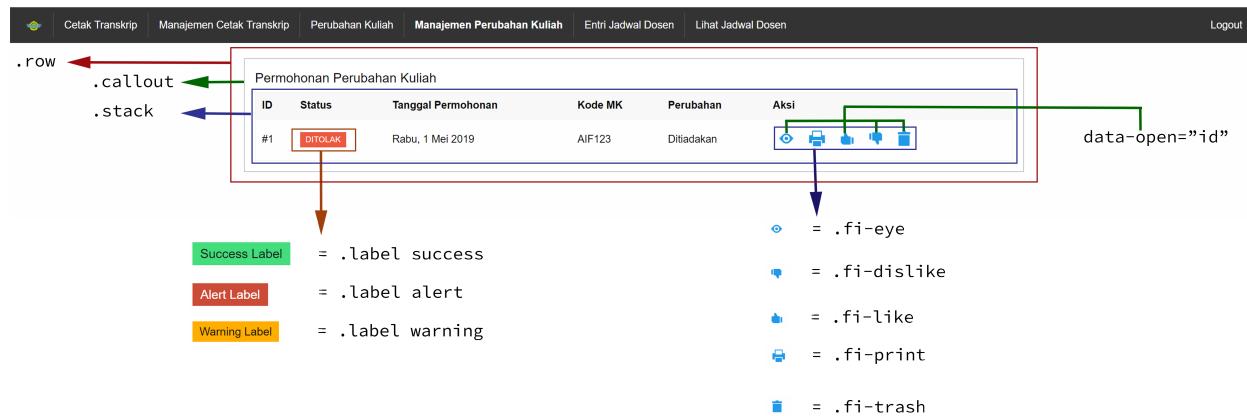
Tabel 3.14: Penggunaan kelas komponen modal lihat permintaan perubahan kuliah.

Kelas	Penggunaan
.reveal data-reveal	Membuat modal yang menampung tabel detail permohonan.
.close-button data-close aria-label	Menutup modal yang telah terbuka dengan memberikan label 'x' pada tombol.
.stack	Membuat tabel detail permohonan perubahan kuliah.

### 3.2.7 Halaman Manajemen Perubahan Kuliah

Halaman pada gambar 3.10 menampilkan data permohonan perubahan kuliah berisi sebuah tabel dimana pada kolom 'Status' menggunakan kelas .callout untuk *highlight* status dan pada kolom 'Aksi' menampilkan lima jenis ikon dari *Library Font Awesome*. Setiap ikon merupakan link menuju modal.

## Halaman Utama



Gambar 3.10: Analisis halaman manajemen perubahan kuliah.

Kelas - kelas yang digunakan pada gambar diatas tertera pada tabel 3.15:

Tabel 3.15: Penggunaan kelas pada halaman manajemen perubahan kuliah.

Kelas	Penggunaan
.row	Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa <i>field-form</i> menjadi satu baris.
.callout	Untuk membuat border yang memisahkan konten permohonan baru dan histori permohonan.
.stack	Jenis tabel yang digunakan tabel histori permohonan, sehingga pada layar medium tabel akan tersusun secara bertumpuk.

Pada kolom 'Status' kelas yang digunakan dari komponen callout tertera pada tabel 3.16.

Tabel 3.16: Penggunaan komponen label pada kolom status permohonan perubahan kuliah.

Kelas	Penggunaan
.label success	Label untuk permintaan perubahan kuliah telah disetujui.
.label alert	Label untuk permintaan perubahan kuliah telah ditolak.
.label warning	Label untuk permintaan perubahan kuliah sedang diproses.

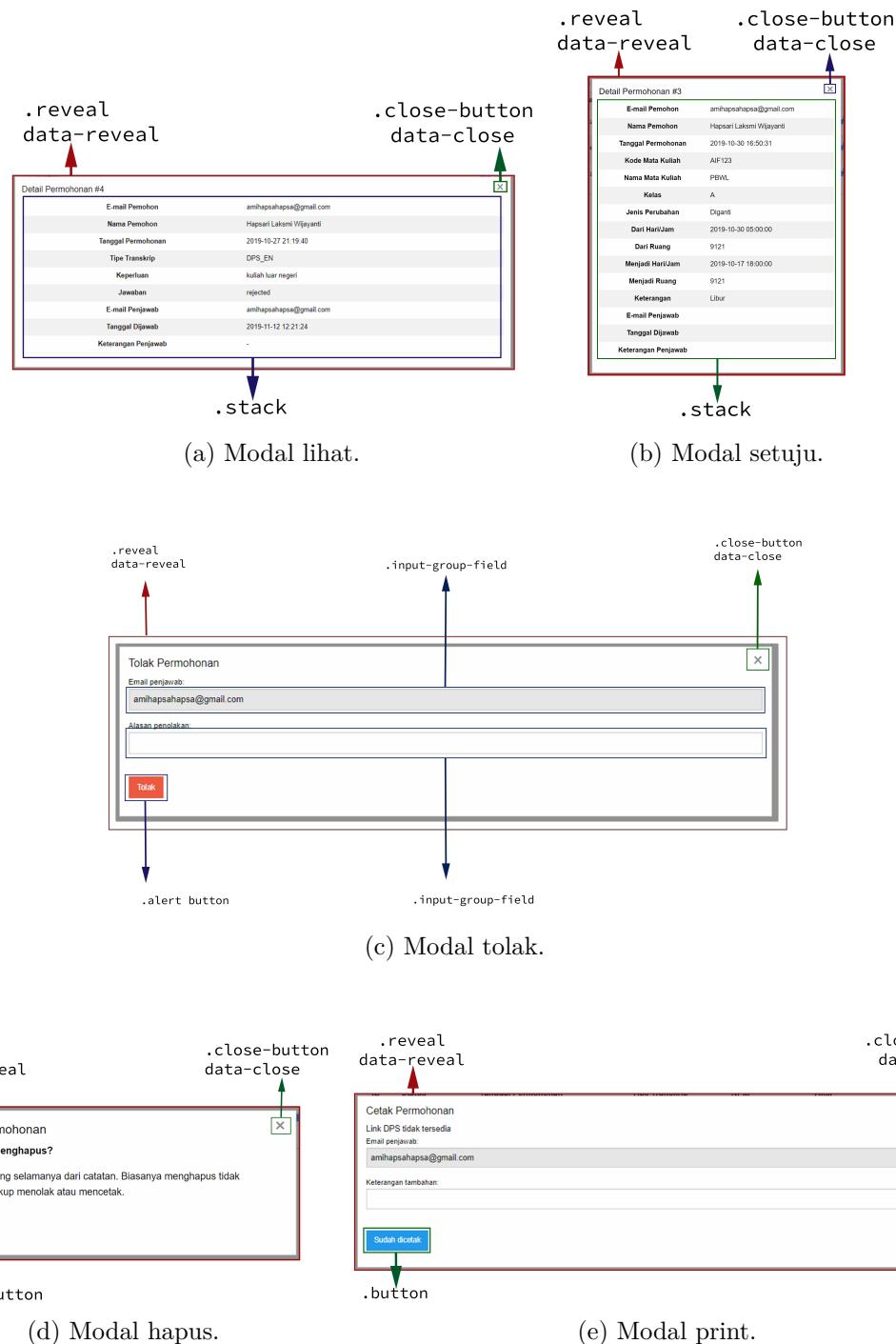
Kemudian pada kolom 'Aksi' kelas yang digunakan dari *library Font Awesome* tertera pada tabel 3.17.

Tabel 3.17: Penggunaan ikon font awesome pada halaman manajemen perubahan kuliah.

Kelas	Penggunaan
fi-eye	Ikon menuju modal lihat permohonan perubahan kuliah.
fi-like	Ikon menuju modal persetujuan permohonan perubahan kuliah.
fi-dislike	Ikon menuju modal persetujuan penolakan perubahan kuliah.
fi-print	Ikon untuk menuju halaman cetak jadwal perubahan kuliah.
fi-trash	Ikon untuk menghapus permintaan perubahan kuliah.

## Modal

Apabila user memilih salah satu ikon dari kolom 'Aksi' maka modal akan muncul. Terdapat lima macam modal yaitu: modal lihat, modal setuju, modal tolak, modal print dan modal hapus perubahan kuliah. Berikut ini gambar 3.11 modal yang ada pada halaman manajemen perubahan kuliah



Gambar 3.11: Analisis modal pada halaman manajemen perubahan kuliah.

Kelas - kelas yang digunakan untuk seluruh modal tertera pada tabel 3.18:

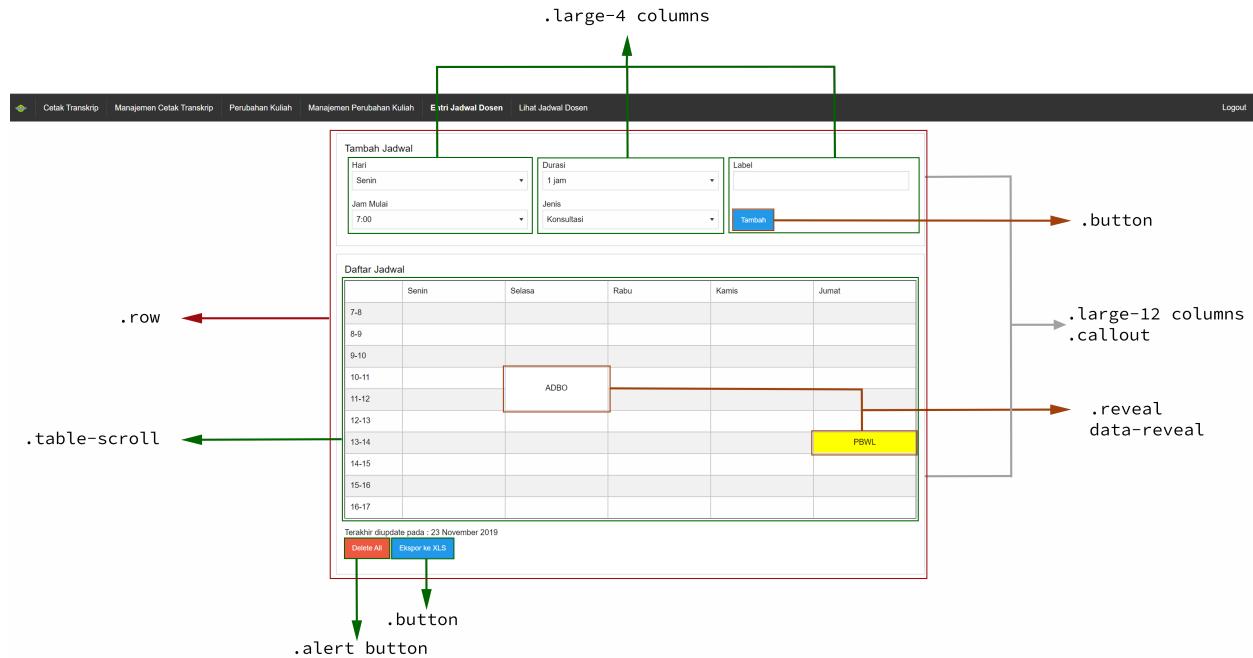
Tabel 3.18: Penggunaan kelas modal manajemen perubahan kuliah.

Kelas	Penggunaan
.reveal	Membuat modal yang menampung tabel detail permohonan.
data-reveal	
.close-button	Menutup modal yang telah terbuka dengan memberikan label 'x' pada tombol.
data-close	
aria-label	
.stack	Membuat tabel detail permohonan perubahan kuliah.
.alert button	Membuat button pada tombol 'tolak' dan 'hapus'.

### 3.2.8 Halaman Entri Jadwal Dosen

Entri Jadwal Dosen berisi dua konten: "Tambah Jadwal" dan "Daftar Jadwal". Konten "Tambah Jadwal" tertera pada gambar 3.12. Form yang terdiri dari label, *field* dari form tersebut dan tombol biru "Tambah". Konten "Daftar Jadwal" terdiri dari sebuah tabel dimana untuk *cell* yang terisi mereferensikan ke modal edit jadwal dosen. Selain itu terdapat dua tombol pada bawah tabel: tombol merah "Delete" dan tombol biru "Ekspor ke XLS".

#### Halaman Utama



Gambar 3.12: Analisis halaman entri jadwal dosen.

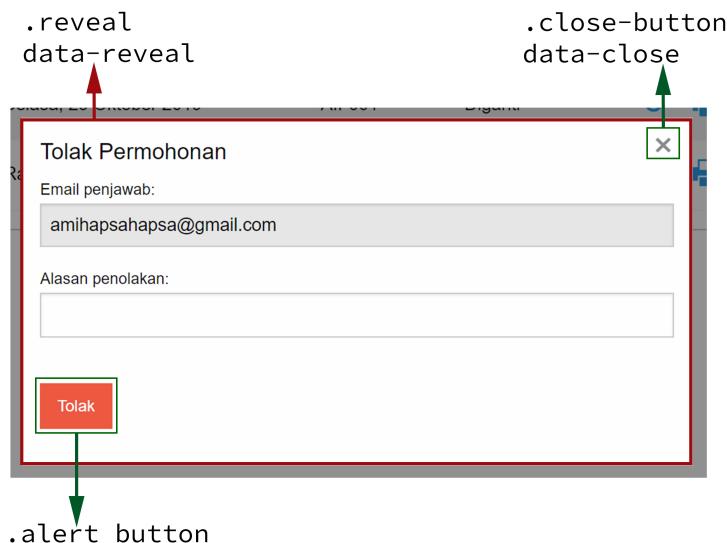
Kelas - kelas yang digunakan dalam halaman entri jadwal dosen tertera pada tabel 3.19:

Tabel 3.19: Penggunaan kelas pada halaman entri jadwal dosen.

Kelas	Penggunaan
.row	Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa <i>field-form</i> menjadi satu baris.
.large-4 column	Setiap field akan pada <i>form</i> Tambah Jadwal akan memiliki lebar masing-masing 4 grid pada layar <i>large</i> .
.large-12 column	Konten Tambah Jadwal dan Daftar Jadwal memiliki lebar 12 grid.
.callout	Untuk membuat border yang memisahkan konten tambah jadwal dan detail jadwal.
.table-scroll	Membuat tabel daftar jadwal dapat digerakkan secara horizontal.
button	Membuat button pada tombol ‘Eksport ke XLS’ untuk konten Daftar Jadwal dan ‘Tambah’ pada konten Tambah Jadwal.
.alert button	Membuat button pada tombol ‘Delete All’.

### Edit Jadwal

Cell yang terisi dengan mata kuliah pada halaman ini dapat diedit. Gambar 3.13 merupakan analisis modal edit yang muncul ketika sebuah jadwal dipilih.



Gambar 3.13: Analisis modal edit entri jadwal dosen.

Kelas yang digunakan modal edit jadwal dosen tertera pada tabel 3.20:

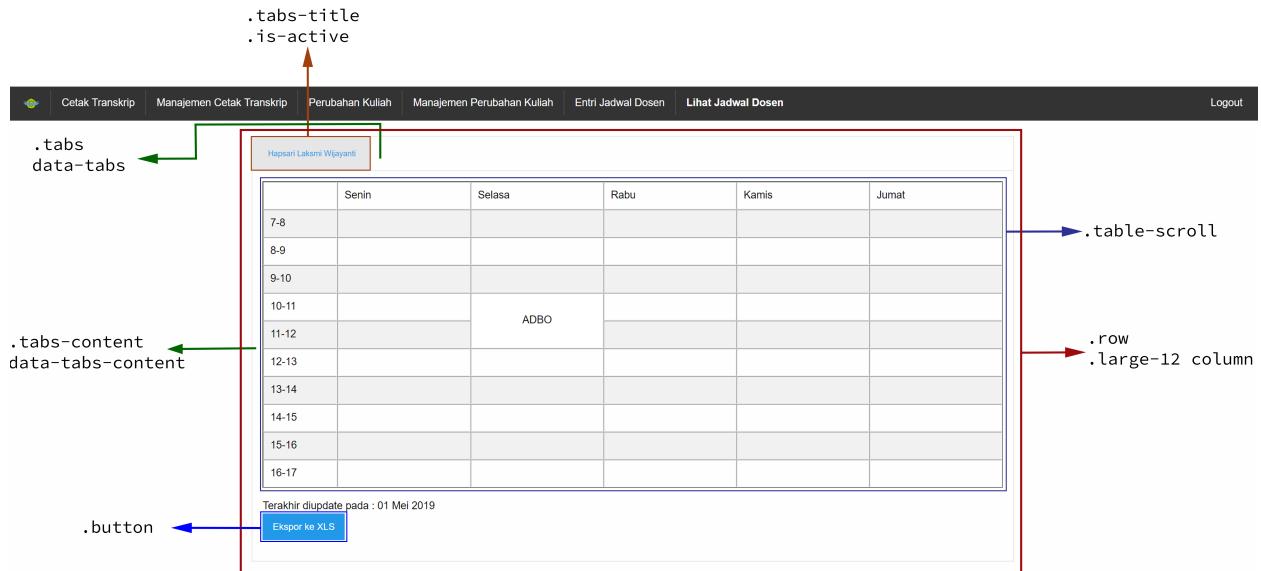
Tabel 3.20: Penggunaan kelas pada modal edit jadwal dosen.

Kelas	Penggunaan
.reveal data-reveal	Membuat modal yang menampung tabel detail permohonan.
.close-button data-close aria-label	Menutup modal yang telah terbuka dengan memberikan label 'x' pada tombol.
.large-4 column	Kolom tombol akan memiliki lebar 4 grid pada layar large.
.large-2 column	Masing - masing tombol 'Save' dan 'Delete' akan memiliki lebar 2 grid pada layar large.
.button	Membuat button 'Save'.
.alert button	Membuat button 'Delete'.

### 3.2.9 Halaman Lihat Jadwal Dosen

Gambar 3.14 merupakan halaman lihat jadwal dosen berisi sebuah tabs, dimana setiap judul tabs yang berada diatas tabel mereferensikan ke **tabs-content** atau halaman jadwal setiap dosen. Kemudian pada bagian bawah tabel terdapat tombol biru "Ekspor ke XLS".

#### Halaman Utama



Gambar 3.14: Analisis halaman lihat jadwal dosen.

Kelas-kelas yang digunakan pada modal edit jadwal tertera pada tabel 3.21:

Tabel 3.21: Penggunaan kelas pada halaman lihat jadwal dosen.

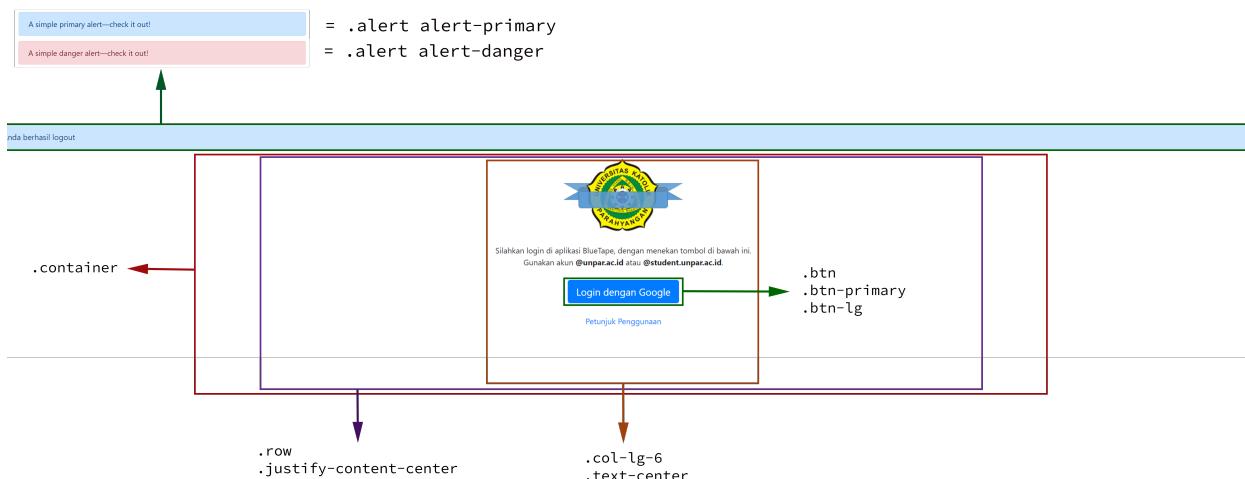
Kelas	Penggunaan
.row	Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa <i>field-form</i> menjadi satu baris.
.large-12 column	Konten Tambah Jadwal dan Daftar Jadwal memiliki lebar 12 grid.
.callout	Untuk membuat border yang memisahkan konten tambah jadwal dan detail jadwal.
.table-scroll	Membuat tabel lihat jadwal dapat digerakkan secara horizontal.
button	Membuat button pada tombol ‘Ekspor ke XLS’ untuk konten Daftar Jadwal dan ‘Tambah’ pada konten Tambah Jadwal.
.tabs data-tabs	Kontainer untuk simpan nama dosen
.tabs-content data-tabs-content	Kontainer untuk simpan isi konten dari tabs.
.tabs-title	Kelas untuk tabs nama-nama dosen.
.is-active	Menunjukkan tabs nama dosen yang sedang dilihat.

### 3.3 Perancangan Halaman Website dengan Bootstrap 4

Setelah menganalisis kelas apa saja yang digunakan dalam website, pada bagian ini akan ditampilkan rancangan penggunaan kelas - kelas dengan menggunakan Bootstrap 4. Perancangan akan dibuat dalam bentuk visual yaitu screenshot setiap halaman dan kelas yang nantinya digunakan dalam tahap implementasi. Selain itu dipaparkan tabel perbandingan antara kelas yang dahulu digunakan dan kelas yang akan digunakan pada tahap implementasi.

#### 3.3.1 Halaman Login

Gambar 3.15 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman login.



Gambar 3.15: Konversi halaman login.

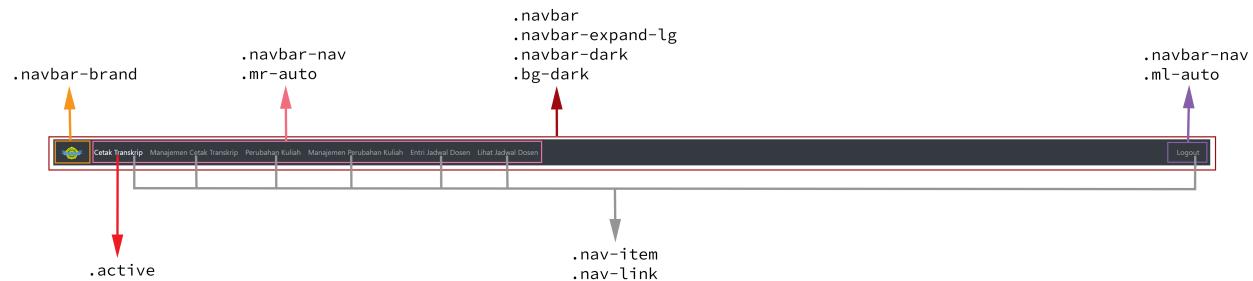
Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada halaman login tertera pada tabel 3.22.

Tabel 3.22: Tabel konversi pada halaman login.

Jenis Komponen	Foundation 6	Bootstrap 4
Grid untuk menampung konten	.row, .column .row	.container .row
Konten berada di tengah	.large-centered	.justify-content-center
Grid dengan lebar 6	.large-6	.col-lg-6
Posisi text rata tengah	.text-center	.text-center
Tombol berwarna biru	.button	.btn
Tombol selebar konten	.btn expand	.btn-lg
Label berwarna merah	.callout alert	.alert alert-primary
Label berwarna biru	.callout primary	.alert alert-danger

### 3.3.2 Menu Navigasi

Gambar 3.16 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada menu navigasi.

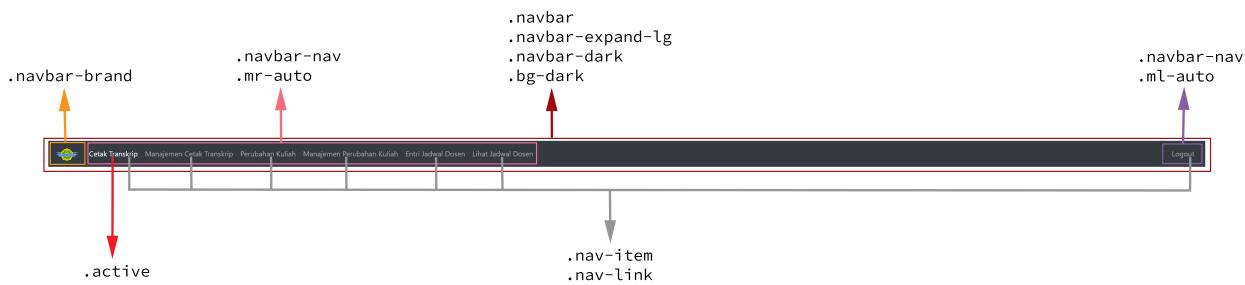


Gambar 3.16: Konversi menu navigasi di layar *large*.

Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada menu navigasi tertera pada tabel 3.23.

Tabel 3.23: Tabel konversi pada menu navigasi di layar *large*.

Jenis Komponen	Foundation 6	Bootstrap 4
Judul website	.title-bar-title	.navbar-brand
Menu	.menu	.top-bar
Sub-menu terpilih	.menu-active	.active
Kelas untuk setiap menu	.menu-text	.nav-item .nav-link
Menu berada di kiri	.top-bar-left	.ml-auto
Menu berada di kanan	.top-bar-right	.mr-auto
Tema untuk menu	-	.navbar-dark .bg-dark



Gambar 3.17: Konversi menu navigasi di layar *small* dan *medium*.

Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada menu navigasi tertera pada tabel 3.24.

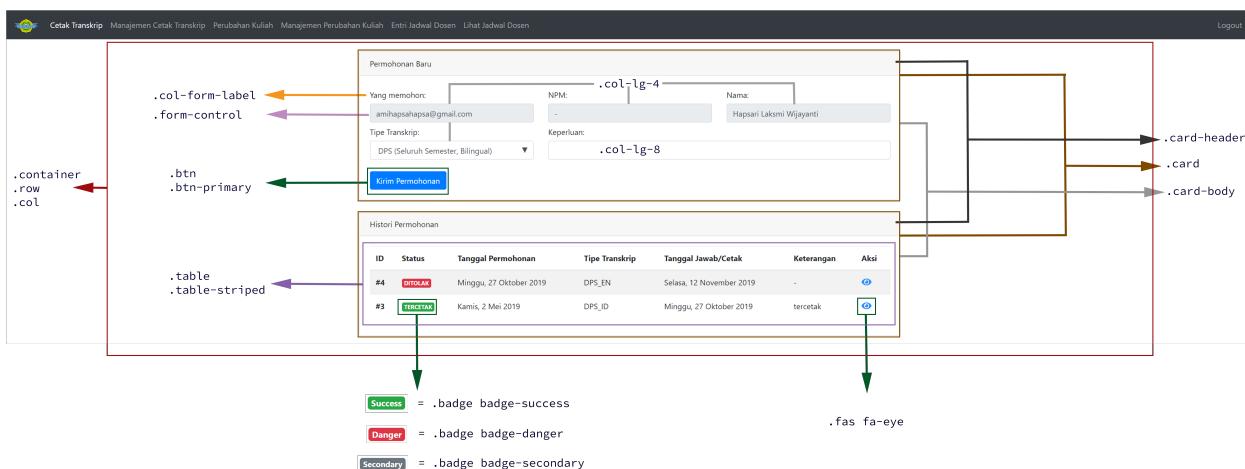
Tabel 3.24: Tabel konversi pada menu navigasi di layar *medium* dan *small*.

Jenis Komponen	Foundation 6	Bootstrap 4
Aksi collapse menu	.title-bar data-responsive-toggle data-hide-for	.collapse .navbar-collapse
Menu dropdown	.menu	.navbar-toggler
Ikon menu	.menu-icon data-toggle	.navbar-toggler-icon

### 3.3.3 Halaman Permintaan Cetak Transkrip

Gambar 3.18 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman permintaan cetak transkrip.

#### Halaman Utama



Gambar 3.18: Konversi halaman cetak transkrip.

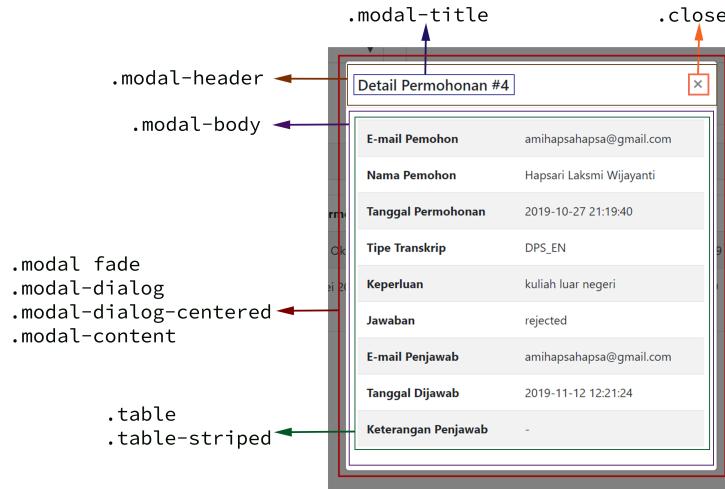
Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada halaman permintaan cetak transkrip tertera pada tabel 3.25.

Tabel 3.25: Tabel konversi pada halaman permintaan cetak transkrip.

Jenis Komponen	Foundation 6	Bootstrap 4
Sistem Grid	.row	.container
Border untuk konten	.callout	.card .card-header .card-body
Lebar grid pada layar medium	.medium-*	.col-md-*
Lebar grid pada layar large	.larger-*	.col-lg-*
Kolom	.column	.col
Tombol berwarna biru	.button	.btn btn-primary
Label berwarna hijau	.label success	.badge badge-success
Label berwarna merah	.label alert	.badge badge-danger
Label berwarna abu	.label secondary	.badge badge-secondary
Ikon bentuk mata	.fi-eye	.fas fa-eye
Tabel	.stack	.table table-striped
Nama <i>form</i>	-	.col-form-label
Input <i>form</i>	-	.form-control

### Modal : Lihat

Gambar 3.19 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada modal lihat halaman permintaan transkrip.



Gambar 3.19: Konversi modal lihat permintaan cetak transkrip.

Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada modal lihat permintaan cetak transkrip tertera pada tabel 3.26.

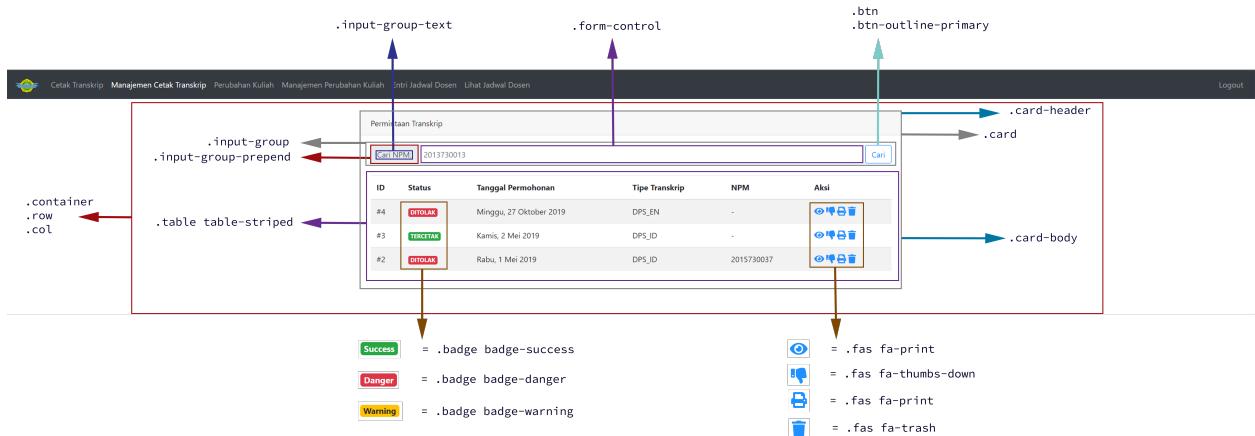
Tabel 3.26: Tabel konversi pada modal lihat permintaan cetak transkrip.

Jenis Komponen	Foundation 6	Bootstrap 4
Modal	.reveal data-reveal	.modal-fade .modal-dialog .modal-dialog-centered .modal-content .modal-title
Judul modal	-	
Isi modal	-	.modal-body
Tutup modal	.close-button data-close aria-label	.close
Tabel	.stack	.table .table-striped

### 3.3.4 Halaman Manajemen Cetak Transkrip

Gambar 3.20 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman manajemen cetak transkrip.

#### Halaman Utama



Gambar 3.20: Konversi manajemen cetak transkrip.

Tabel 3.27: Tabel konversi pada halaman manajemen cetak transkrip.

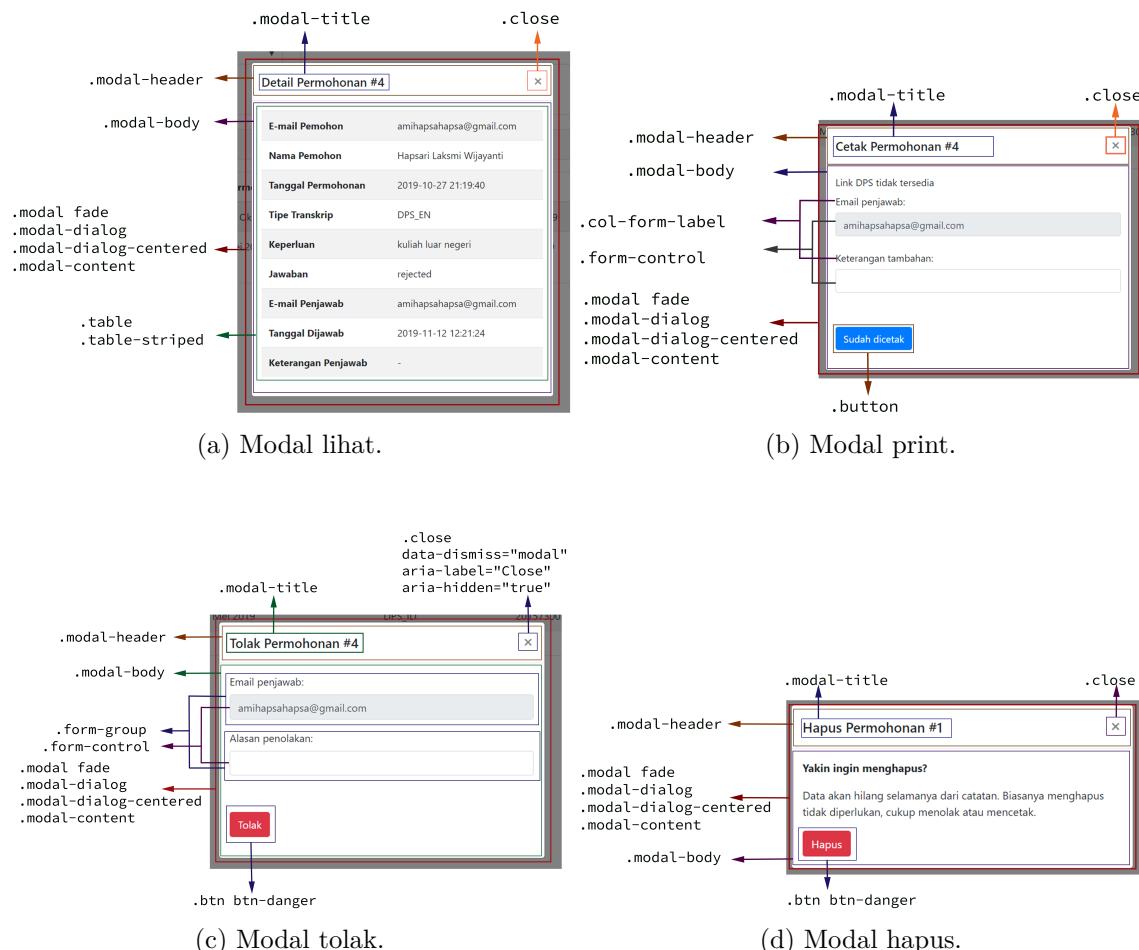
Jenis Komponen	Foundation 6	Bootstrap 4
Sistem Grid	.row	.container
Border untuk konten	.callout	.card .card-header .card-body
Kolom	.column	.col
Label berwarna hijau	.label success	.badge badge-success
Label berwarna merah	.label alert	.badge badge-danger
Label berwarna abu	.label secondary	.badge badge-secondary

Tabel 3.27: Tabel konversi pada halaman manajemen cetak transkrip (cont.).

Jenis Komponen	Foundation 6	Bootstrap 4
Ikon bentuk <i>eye</i>	.fi-eye	.fas fa-eye
Ikon bentuk <i>down</i>	.fi-dislike	.fas fa-thumbs-down
Ikon bentuk <i>print</i>	.fi-print	.fas fa-print
Ikon bentuk <i>trash</i>	.fi-trash	.fas fa-trash
Tabel	.stack	.table table-striped
<i>Input Group</i>	.input-group .input-group-label .input-group-field .input-group-button	.input-group-append .input-group-text .form-control .btn btn-primary

### Modal: Lihat, Tolak, Print, Hapus

Gambar 3.21 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada modal-modal halaman manajemen cetak transkrip.



Gambar 3.21: Konversi modal pada halaman manajemen cetak transkrip.

Penjabaran kelas - kelas yang digunakan ada pada tabel 3.28

Tabel 3.28: Tabel konversi modal pada manajemen cetak transkrip.

Jenis Komponen	Foundation 6	Bootstrap 4
Modal	.reveal data-reveal	.modal-fade .modal-dialog .modal-dialog-centered .modal-content
Judul modal	-	modal-title
Isi modal	-	modal-body
Tutup modal	.close-button data-close aria-label	.close
Tabel	.stack	.table .table-striped

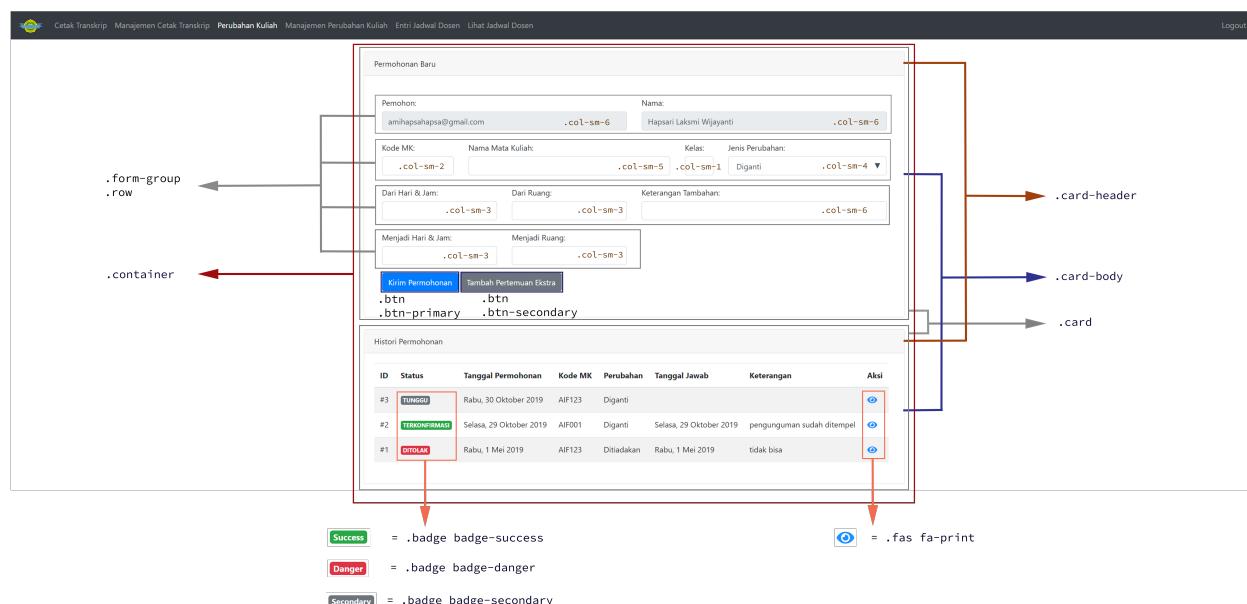
Tabel 3.28: Tabel konversi modal pada manajemen cetak transkrip(cont.).

Jenis Komponen	Foundation 6	Bootstrap 4
Judul form	-	.col-form-label
Field form	.input-group-field	.form-control
Form group	.input-group	.form-group
Tombol berwarna biru	.button	.btn-primary
Tombol berwarna merah	.alert button	.btn btn-danger

### 3.3.5 Halaman Permintaan Perubahan Kuliah

#### Halaman Utama

Gambar 3.22 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman permintaan perubahan kuliah.



Gambar 3.22: Konversi halaman permintaan perubahan kuliah.

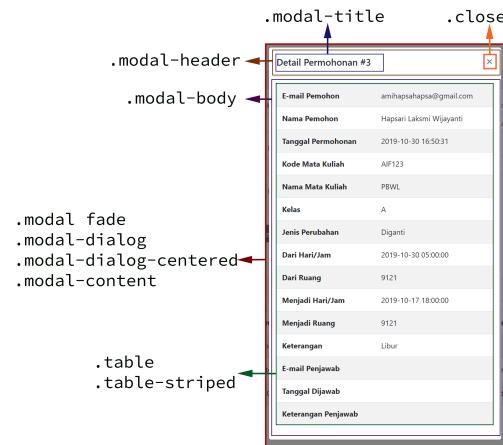
Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada halaman permintaan perubahan kuliah tertera pada tabel 3.29.

Tabel 3.29: Tabel konversi pada halaman permintaan perubahan kuliah.

Jenis Komponen	Foundation 6	Bootstrap 4
Sistem Grid	.row	.container
Border untuk konten	.callout	.card .card-header .card-body
Ukuran grid	.small-*	.col-sm-*
Kolom	.column	.col
Tombol berwarna biru	.button	.btn btn-primary
Label berwarna hijau	.label success	.badge badge-success
Label berwarna merah	.label alert	.badge badge-danger
Label berwarna abu	.label secondary	.badge badge-secondary
Ikon bentuk mata	.fi-eye	.fas fa-eye
Tabel	.stack	.table table-striped
Nama <i>form</i>	-	.col-form-label
Input <i>form</i>	-	.form-control

### Modal: Lihat

Gambar 3.23 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada modal lihat permintaan perubahan kuliah.



Gambar 3.23: Konversi modal lihat permintaan perubahan kuliah.

Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada modal lihat permintaan perubahan kuliah tertera pada tabel 3.30.

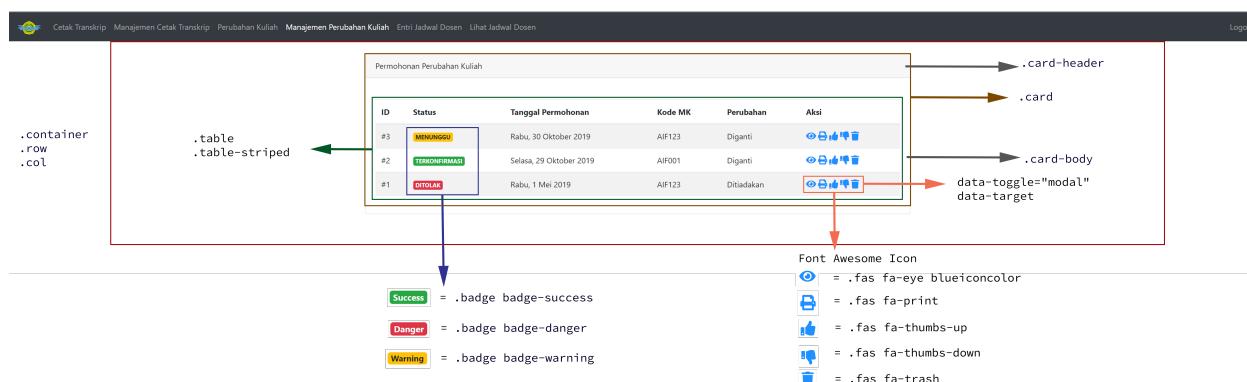
Tabel 3.30: Tabel konversi pada modal lihat permintaan perubahan kuliah.

Jenis Komponen	Foundation 6	Bootstrap 4
Modal	.reveal data-reveal	.modal-fade .modal-dialog .modal-dialog-centered .modal-content
Judul modal	-	modal-title
Isi modal	-	modal-body
Tutup modal	.close-button data-close aria-label	.close
Tabel	.stack	.table .table-striped

### 3.3.6 Halaman Manajemen Perubahan Kuliah

#### Halaman Utama

Gambar 3.24 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman manajemen perubahan kuliah.



Gambar 3.24: Konversi halaman manajemen perubahan kuliah.

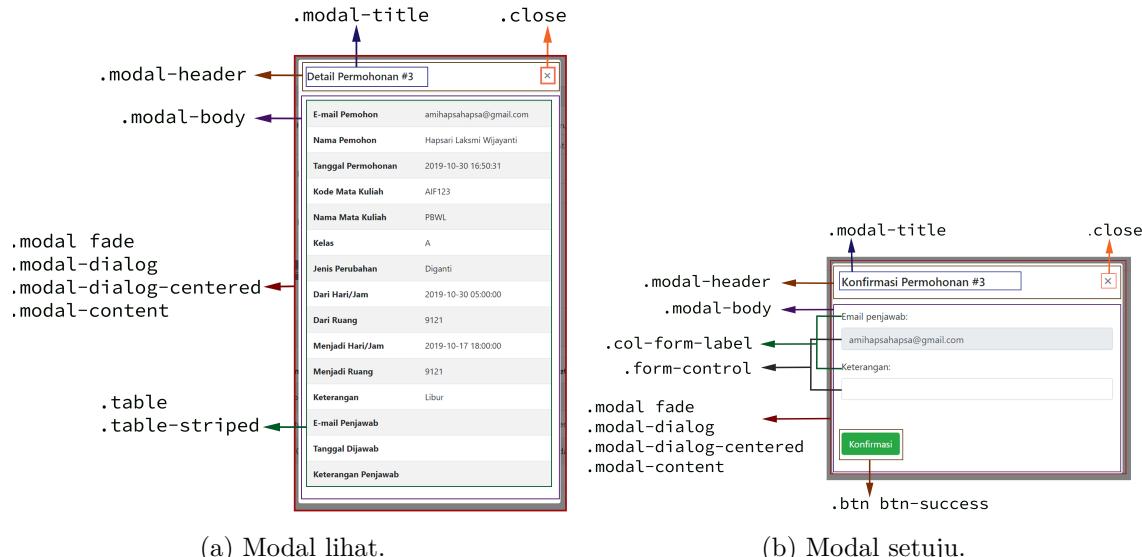
Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada halaman manajemen perubahan kuliah tertera pada tabel 3.31.

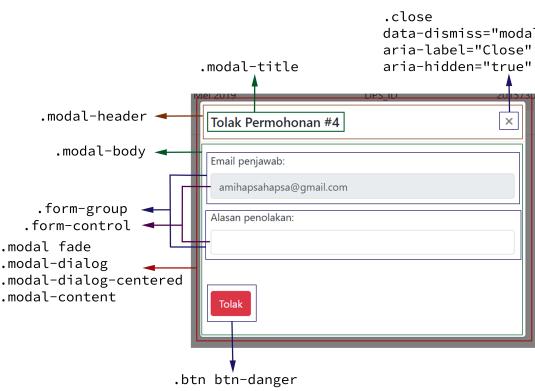
Tabel 3.31: Tabel konversi pada halaman manajemen perubahan kuliah.

Jenis Komponen	Foundation 6	Bootstrap 4
Sistem Grid	.row	.container .row .col
Border untuk konten	.callout	.card .card-header .card-body
Label berwarna hijau	.label success	.badge badge-success
Label berwarna merah	.label alert	.badge badge-danger
Label berwarna kuning	.label warning	.badge badge-warning
Link modal sesuai ID	data-open	data-toggle data-target
Ikon bentuk <i>eye</i>	.fi-eye	.fas fa-eye
Ikon bentuk <i>down</i>	.fi-dislike	.fas fa-thumbs-down
Ikon bentuk <i>down</i>	.fi-like	.fas fa-thumbs-up
Ikon bentuk <i>print</i>	.fi-print	.fas fa-print
Ikon bentuk <i>trash</i>	.fi-trash	.fas fa-trash
Tabel	.stack	.table table-striped
<i>Input Group</i>	.input-group .input-group-label .input-group-field .input-group-button	.input-group-append .input-group-text .form-control .btn btn-primary

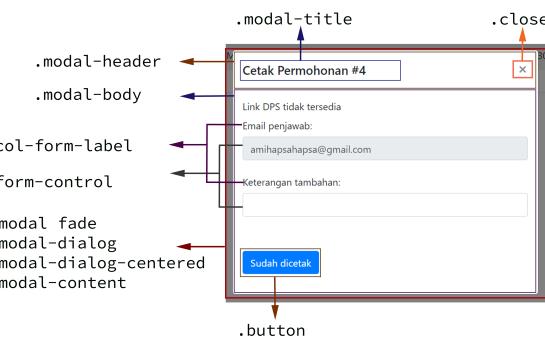
### Modal: Lihat, Setuju, Tolak, Print, Hapus

Gambar 3.25 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman modal manajemen perubahan kuliah.





(c) Modal tolak.



(d) Modal print



(e) Modal hapus.

Gambar 3.25: Konversi modal pada halaman manajemen perubahan kuliah.

Pada tabel 3.32 menjelaskan kelas-kelas yang digunakan dalam komponen modal.

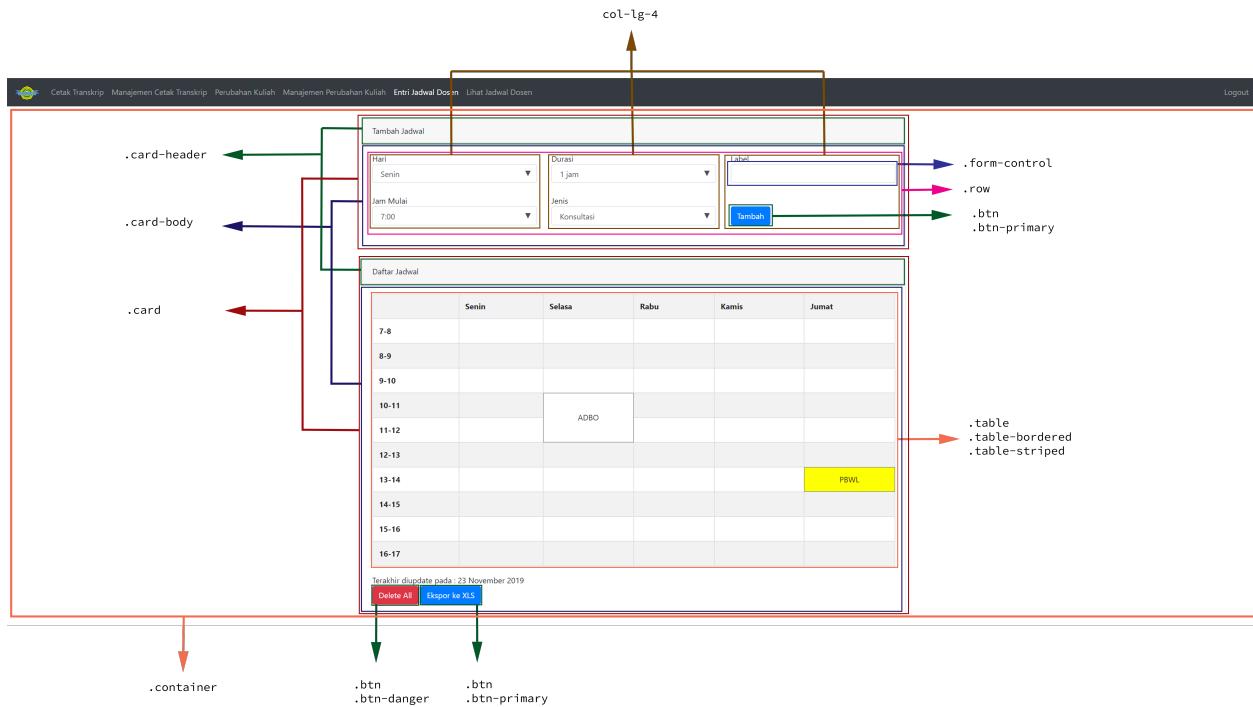
Tabel 3.32: Tabel konversi pada modal manajemen perubahan kuliah.

Jenis Komponen	Foundation 6	Bootstrap 4
Modal	.reveal data-reveal	.modal-fade .modal-dialog .modal-dialog-centered .modal-content
Judul modal	-	.modal-title
Isi modal	-	.modal-body
Tutup modal	.close-button data-close aria-label	.close
Tabel	.stack	.table .table-striped

### 3.3.7 Halaman Entri Jadwal Dosen

#### Halaman Utama

Gambar 3.26 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman entri jadwal dosen.



Gambar 3.26: Konversi halaman entri jadwal dosen.

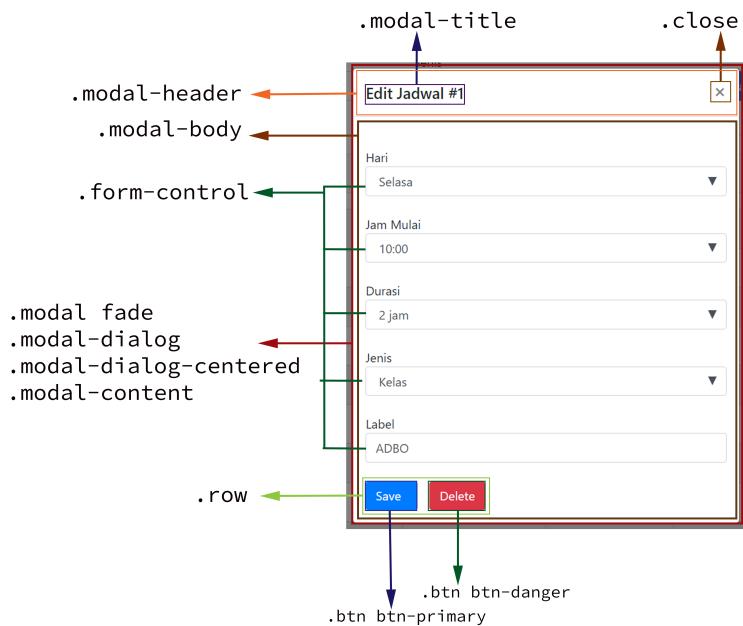
Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada halaman entri jadwal dosen tertera pada tabel 3.33.

Tabel 3.33: Tabel konversi pada halaman entri jadwal dosen.

Jenis Komponen	Foundation 6	Bootstrap 4
Sistem Grid	.row .col	.container .row .col
Border untuk konten	.callout	.card .card-header .card-body
Form	-	.form-control
Link modal sesuai ID	data-open	data-toggle data-target
Tabel	.stack	.table .table-bordered .table table-striped

### Modal: Edit

Gambar 3.27 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada modal entri jadwal dosen.



Gambar 3.27: Konversi modal edit pada halaman entri jadwal dosen.

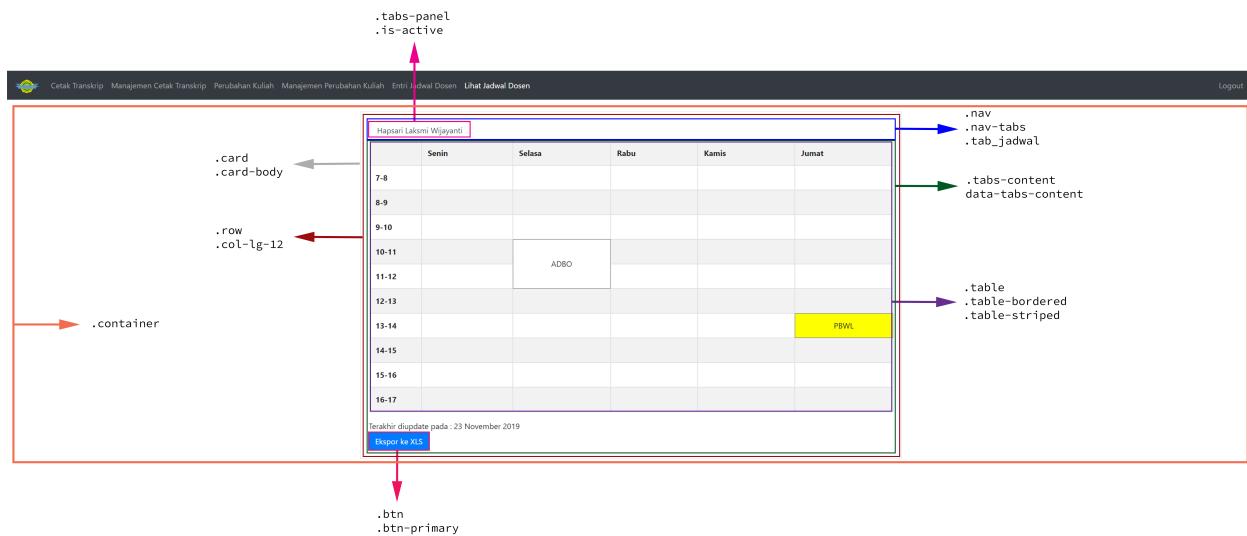
Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada modal edit entri jadwal dosen tertera pada tabel 3.34.

Tabel 3.34: Tabel konversi pada modal edit pada halaman entri jadwal dosen.

Jenis Komponen	Foundation 6	Bootstrap 4
Grid	.row	.row .container
Ukuran Grid	.large-*--columns	.col-lg-*
Modal	.reveal data-reveal	.modal-fade .modal-dialog .modal-dialog-centered .modal-content
Judul modal	-	modal-title
Isi modal	-	modal-body
Form	-	form-control
Tombol berwarna merah	.alert button	.btn btn-danger
Tombol berwarna biru	.button	.btn btn-primary
Tabel	.table-scroll	.table .table-striped

### 3.3.8 Halaman Lihat Jadwal Dosen

Gambar 3.28 menjelaskan komponen dalam website beserta penamaan kelas untuk Bootstrap 4 pada halaman lihat jadwal dosen.



Gambar 3.28: Konversi modal lihat jadwal dosen.

Perbandingan penggunaan kelas pada Foundation 6 dan Bootstrap 4 pada halaman lihat jadwal dosen tertera pada tabel 3.35.

Tabel 3.35: Tabel konversi kelas pada lihat jadwal dosen.

Jenis Komponen	Foundation 6	Bootstrap 4
Sistem Grid	.row .col	.container .row
Judul tabs	.tabs data-tabs	.nav .nav-tabs tab-jadwal
Isi tabs	.tabs-content data-tabs-content	.tabs-content data-tabs-content
Tabs active	.tabs-title .is-active	.tab-panel .is-active
Tabel	.stack	.table .table table-striped .table-bordered
Tombol berwarna biru	.button	.btn btn-primary



## BAB 4

# IMPLEMENTASI WEBSITE BLUETAPE DENGAN BOOTSTRAP

## 4

Bab 4 menjelaskan implementasi website BlueTape dengan Framework Bootstrap 4. Pertama akan dijelaskan file apa saja yang dirubah. Kedua perubahan akses apa saja yang diubah oleh programmer untuk menjalankan website. Terakhir akan disajikan detail perubahan kode yang dijelaskan dalam kode **normal diff**.

### 4.1 Daftar File yang Diubah dan Diganti pada Folder

Bagian [4.1](#) menjelaskan isi file website BlueTape dan status file yang diganti dan diubah.

Listing 4.1: Perubahan isi folder BlueTape

```
.  
|-- nbproject  
|-- vendor  
|-- www  
|   |-- application  
|   |   |-- cache  
|   |   |-- config  
|   |   |   |-- auth.php // Kode diubah  
|   |   |   |-- database.php // Kode diubah  
|   |   |   |-- ..dst  
|   |   |-- controller  
|   |   |   |-- Auth.php  
|   |   |   |-- EntriJadwalDosen.php  
|   |   |   |-- index.html  
|   |   |   |-- LihatJadwalDosen.php  
|   |   |   |-- Migrate.php  
|   |   |   |-- PerubahanKuliahManage.php // Kode diubah  
|   |   |   |-- PerubahanKuliahRequest.php // Kode diubah  
|   |   |   |-- TranskripManage.php // Kode diubah  
|   |   |   |-- TranskripRequest.php // Kode diubah  
|   |   |-- core  
|   |   |-- helper  
|   |-- hooks  
|   |-- languange  
|   |-- libraries  
|   |-- logs  
|   |-- migration  
|   |-- models  
|   |   |--third_party
```

```
|   |   |--views
|   |   |   |-- auth
|   |   |   |   |-- index.html
|   |   |   |   |-- login.php // Kode diubah
|   |   |   |-- EntriJadwalDosen
|   |   |   |   |-- main.php // Kode diubah
|   |   |   |-- errors
|   |   |   |-- LihatJadwalDosen
|   |   |   |   |-- main.php // Kode diubah
|   |   |   |-- PerubahanKuliahManage
|   |   |   |   |-- email.php
|   |   |   |   |-- index.html
|   |   |   |   |-- main.php // Kode diubah
|   |   |   |   |-- printview.php
|   |   |   |-- PerubahanKuliahRequest
|   |   |   |   |-- email.php
|   |   |   |   |-- index.html
|   |   |   |   |-- main.php // Kode diubah
|   |   |   |-- templates
|   |   |   |   |-- flasmessage.php // Kode diubah
|   |   |   |   |-- head_loggedin.php // Kode diubah
|   |   |   |   |-- script.foundation.php // Kode diubah
|   |   |   |   |-- topbar_loggedin.php // Kode diubah
|   |   |-- TranskripManage
|   |   |   |-- email.php
|   |   |   |-- index.html
|   |   |   |   |-- main.php // Kode diubah
|   |   |-- TranskripRequest
|   |   |   |-- email.php
|   |   |   |-- index.html
|   |   |   |   |-- main.php // Kode diubah
|   |   |   |-- index.html
|   |-- .htacces
|   |--index.html
|-- public
  |-- fonts
    |-- OFL.txt
    |-- TitilliumWeb-Black.ttf
    |-- TitilliumWeb-Bold.ttf
    |-- TitilliumWeb-BoldItalic.ttf
    |-- TitilliumWeb-ExtraLight.ttf
    |-- TitilliumWeb-ExtraLightItalic.ttf
    |-- TitilliumWeb-Italic.ttf
    |-- TitilliumWeb-Light.ttf
    |-- TitilliumWeb-LightItalic.ttf
    |-- TitilliumWeb-Regular.ttf
    |-- TitilliumWeb-SemiBold.ttf
    |-- TitilliumWeb-SemiBoldItalic.ttf
  |-- img
    |-- logo.png // File tetap
  |-- lib
```

```
|   |   |   |-- css
|   |   |   |   |-- bootstrap.css // File baru ditambahkan
|   |   |   |   |-- bootstrap-grid.css // File baru ditambahkan
|   |   |   |   |-- bootstrap-reboot.css // File baru ditambahkan
|   |   |   |   |-- app.css
|   |   |   |   |-- foundation.css
|   |   |   |   |-- foundation-datetimepicker.min.css
|   |   |   |   |-- foundation-flex.css
|   |   |   |   |-- foundation-icon.css
|   |   |   |   |-- foundation-icon.eot
|   |   |   |   |-- foundation-icon.svg
|   |   |   |   |-- foundation-icon.ttf
|   |   |   |   |-- foundation-icon.woff
|   |   |   |-- fontawesome
|   |   |   |   |-- fontawesome.css // File baru ditambahkan
|   |   |-- jquery
|   |   |   |--jquery-3.4.1.min.js // File baru ditambahkan
|   |   |-- js
|   |   |   |-- bootstrap.js // File baru ditambahkan
|   |   |   |-- bootstrap.bundle.js // File baru ditambahkan
|   |   |   |-- vendor // Folder dihapus
|   |   |   |-- app.js // Folder dihapus
|   |   |   |-- foundation.js // Folder dihapus
|   |   |-- xdan-datetimepicker
|   |   |   |-- jquery.datetimepicker.full.min.js
|   |   |   |-- jquery.datetimepicker.min.js
|   |   |   |-- jquery.datetimepicker.min.js
|   |   |-- webfonts // Folder ditambahkan, berfungsi untuk simpan ikon.
|   |-- system
|   |-- .htaccess
|   |-- .composer.json
|   |-- .contributig.md
|   |-- .index.php
|   |-- .licence.txt
|   |-- readme.rst
|   |-- web.config
|   .
.
```

## 4.2 Pengaturan Akses bagi *admin*

Sebelumnya menjalankan website, *admin* akan menyunting email admin dan database agar lebih mudah untuk diakses. Hasil perubahan kode ditampilkan dalam berbentuk diff, dimana bagian berwarna merah merupakan kode lama(Foundation 6) dan bagian berwarna hijau merupakan kode terkini (Bootstrap 4).

### 4.2.1 Perubahan Akses URL

Saat admin pertama kali menjalankan aplikasi CodeIgniter maka diperlukan konfigurasi `base_url`. Sehingga URL baru terbentuk dan dapat mengakses *resource* yang ada pada direktori `root`. Penggunaan `base_url` dalam website dijelaskan pada kode [4.2](#)

Listing 4.2: Perubahan file /config/config.php

```

1 diff -r a\www\application\config\config.php b\www\application\config\config.php
2
3 26c26
4 < $config['base_url'] = 'https://bluetape.azurewebsites.net';
5 ---
6 > $config['base_url'] = 'http://127.0.0.1/';

```

#### 4.2.2 Perubahan Akses Email Admin

Kode 4.3 mengarahkan admin ke halaman login setelah aplikasi berjalan, karena website menggunakan *Google API Console* untuk proses autentikasi maka perlu ditambahkan email admin baru pada file `www/application/config/modules.php`. Email ini telah didaftarkan sebelumnya di API tersebut, sehingga admin dapat menggunakan seluruh fitur pada website.

Listing 4.3: Perubahan file /config/modules.php

```

1 diff -r a\www\application\config\modules.php
      b\www\application\config\modules.php
2 25,26c25,26
3 <     'root' => array('pascal@unpar.ac.id', 'shao.wei@unpar.ac.id'),
4 <     'tu.ftis' => array('shao.wei@unpar.ac.id', 'purnomo@unpar.ac.id',
5   'walip@unpar.ac.id'),
6 ---
6 >     'root' => array('pascal@unpar.ac.id', 'shao.wei@unpar.ac.id',
7   'amihapsahapsa@gmail.com'),
7 >     'tu.ftis' => array('shao.wei@unpar.ac.id', 'pranyoto@unpar.ac.id',
8   'walip@unpar.ac.id'),

```

### 4.3 Perubahan Kode pada Tampilan

Bagian ini akan menjelaskan hasil implementasi dalam bentuk *diff*. Diff membedakan kode berdasarkan warna dimana bagian berwarna merah merupakan kode lampau(implementasi dengan Foundation 6) dan bagian berwarna hijau merupakan kode terkini (implementasi dengan Bootstrap 4).

#### 4.3.1 Penggunaan *library* Bootstrap 4 pada Website

Library Bootstrap 4 yang sudah dimasukan pada CodeIgniter akan diintegrasikan pada file `www/application/config/modules.php` dalam bentuk file `.js` dan `head_loggedin.php` dalam bentuk file `.css`. File ini nantinya akan dipakai keseluruh halaman website kecuali halaman login.

Pertama kode 4.4 memanggil file `js` dan `jquery` untuk Bootstrap serta plugin `xdan-datetimepicker` agar dapat diload.

Listing 4.4: Penambahan \views\templates\script\_foundation.php

```

1 diff -r a\www\application\views\templates\script_foundation.php
      b\www\application\views\templates\script_foundation.php
2 3,6c3,4

```

```

3 < ?><script src="/public/js/vendor/jquery.min.js"></script>
4 < <script src="/public/js/vendor/what-input.min.js"></script>
5 < <script src="/public/js/foundation.min.js"></script>
6 < <script src="/public/js/app.js"></script>
7 ---
8 > ?><script src="/public/lib/jquery/jquery-3.4.1.min.js"></script>
9 > <script src="/public/lib/js/bootstrap.js"></script>

```

Kedua file css akan dipanggil, ada tiga macam file yang dipanggil:

- *library* Bootstrap 4.
- Plugin xdan-datetimepicker.
- *library* Font Awesome.

File yang berisi kode 4.5 akan digunakan pada seluruh halaman website kecuali halaman login.

Listing 4.5: Perubahan file \views\templates\head\_loggedin.php

```

1 diff -r a\www\application\views\templates\head_loggedin.php
      b\www\application\views\templates\head_loggedin.php
2 8,11c8,10
3 <     <link rel="stylesheet" href="/public/css/foundation.css" />
4 <     <link rel="stylesheet" href="/public/css/foundation-icons.css" />
5 <     <link rel="stylesheet" href="/public/css/app.css" />
6 <     <link rel="stylesheet"
7         href="/public/lib/xdan-datetimepicker/jquery.datetimepicker.min.css" />
7 ---
8 >     <link rel="stylesheet" href="/public/lib/css/bootstrap.css" />
9 >     <link rel="stylesheet" href="/public/lib/fontawesome/fontawesome.css">
10 >     <link rel="stylesheet"
       href="public/lib/xdan-datetimepicker/jquery.datetimepicker.min.css">

```

### 4.3.2 Halaman Login

Pemanggilan file css dilakukan kembali pada file login.php untuk digunakan pada halaman login website BlueTape. Selain itu file HTML berada di file ini, penjelasan tertera pada kode 4.6

Listing 4.6: Perubahan file \views\auth\login.php

```

1 diff -r a\www\application\views\auth\login.php
      b\www\application\views\auth\login.php
2 10,11c10,13
3 < <link rel="stylesheet" href="/public/css/foundation.css" />
4 < <link rel="stylesheet" href="/public/css/app.css" />
5 ---
6 > <link rel="stylesheet" href="/public/lib/css/bootstrap.css" />
7 > <link rel="stylesheet" href="/public/lib/css/bootstrap-grid.css" />
8 > <link rel="stylesheet" href="/public/lib/css/bootstrap-reboot.css" />
9 > <link rel="stylesheet" href="/public/lib/fontawesome/fontawesome.css" />
10
11 15,16c17,19
12 < <div class="row">
13 <     <div class="large-6 large-centered columns centered">

```

```

14   ---
15   > <div class="container">
16     <div class="row justify-content-center">
17       <div class="col-lg-6">
18
19 21,22c24,25
20   < <a href="= $authURL; ?&gt;" class="button expand"&gt;Login dengan
21     Google&lt;/a&gt;&lt;br/&gt;&lt;br/&gt;
22   &lt; &lt;a class="text-center"
23     href="https://github.com/ftisunpar/BlueTape/wiki/UserGuide"
24     target="_blank"&gt;Petunjuk Penggunaan&lt;/a&gt;
25   ---
26   &gt; &lt;a href="<?= $authURL; ?&gt;" class="btn btn-primary btn-lg"&gt;Login dengan
27     Google&lt;/a&gt;&lt;br/&gt;&lt;br/&gt;
28   &gt; &lt;a href="https://github.com/ftisunpar/BlueTape/wiki/UserGuide"
29     target="_blank"&gt;Petunjuk Penggunaan&lt;/a&gt;
</pre

```

Penjelasan kode diatas tertera pada tabel 4.1:

No Line	Implementasi	Penjelasan
2-9	<b>Pemanggilan file</b>	File css dari Foundation akan diganti dengan file css dari Bootstrap 4.
11-17	<b>Grid konten</b>	Dalam Foundation 6 untuk membuat satu baris konten hanya menggunakan kelas .row, namun di Bootstrap 4 perlu ditambahkan kelas .container. Untuk size sama - sama menggunakan lebar 6 grid yang posisinya diletakkan ditengah website.
19-24	<b>Text</b>	Teks akan diletakkan ditengah, hanya di Foundation 6 saja perlu menginisiasi kelas .text-center pada Bootstrap 4 tidak perlu.

Tabel 4.1: Penjelasan kode yang dipakai pada halaman manajemen perubahan kuliah.

Implementasi notifikasi user terdapat pada kode 4.7.

Listing 4.7: Perubahan file \views\templates\flashmessage.php

```

1 diff -r a\www\application\views\templates\flashmessage.php
2   b\www\application\views\templates\flashmessage.php
3
4 3c3
5   < ?><div class="row">
6   ---
7   > ?>
8
9 5c5
10  < <div class="callout alert"><?= $_SESSION['error'] ?></div>
11  ---
12  > <div class="alert alert-danger" role="alert"><?= $_SESSION['error'] ?></div>
13
14 8c8
15  < <div class="callout primary"><?= $_SESSION['info'] ?></div>
16  ---
17  > <div class="alert alert-primary" role="alert"><?= $_SESSION['info'] ?></div>

```

Beberapa catatan dari kode diatas berada pada tabel 4.2:

No	Line	Implementasi	Penjelasan
8-16		<b>Alert</b>	Labelling menggunakan .callout pada Foundation 6. Bootstrap 4 mrnggunakan kelas .alert dan atribut role.

Tabel 4.2: Penjelasan kode konversi halaman login.

### 4.3.3 Menu Navigasi

Kode 4.8 menjelaskan menu navigasi secara umum terdiri dari kelas untuk judul website, letak menu dan menu yang sedang aktif.

Listing 4.8: Perubahan file \views\templates\topbar\_loggedin.php

```

1 diff -r a\www\application\views\templates\topbar_loggedin.php
2   b\www\application\views\templates\topbar_loggedin.php
3
4 3,11c3,12
5 < ?><div class="title-bar" data-responsive-toggle="navigation-menu"
6   data-hide-for="medium">
7   <button class="menu-icon" type="button" data-toggle></button>
8   <div class="title-bar-title"></div>
10  </div>
11  <
12  <div class="top-bar" id="navigation-menu">
13    <div class="top-bar-left">
14      <ul class="menu" data-responsive-menu="dropdown">
15        <li class="menu-text"></li>
17
18  --->
19  > ?>
20  >
21  > <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
22  >   <a class="navbar-brand" href="#">     width="50"/></a>
24  >   <button class="navbar-toggler" type="button" data-toggle="collapse"
25  >     data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
26  >     aria-expanded="false" aria-label="Toggle navigation">
27  >     <span class="navbar-toggler-icon"></span>
28  >   </button>
29  >
30  >   <div class="collapse navbar-collapse" id="navbarSupportedContent">
31  >     <ul class="navbar-nav mr-auto">
32
33 13c14
34 < <li<?= $module === $currentModule ? ' class="menu-active"' : '' ?>><a
35  href="/<?= $module ?>"><?= $this->config->item('module-names')[$module]
36  ?></a></li>
37
38  --->
```

```

28 > <li <?= $module == $currentModule ? ' class="nav-item active"' : '
29   class="nav-item "' ?>><a class="nav-link" href="/<?= $module ?>"><?=
30   $this->config->item('module-names')[ $module ] ?></a></li>
31
32 16,19c17,20
33 < </div>
34 < <div class="top-bar-right">
35 <   <ul class="menu">
36 <     <li><a href="/auth/logout">Logout</a></li>
37 --->
38 >   <ul class="navbar-nav ml-auto">
39 >     <li class="nav-item">
40 >       <a class="nav-link" href="/auth/logout">Logout</a>
41 >     </li>

```

Catatan untuk kode 4.8 dijabarkan dalam tabel 4.3:

No Line	Implementasi	Penjelasan
3-39	<b>Navbar</b>	Pada foundation 6 ikon dropdown belum berfungsi, sehingga pada Bootstrap 4 menu dibuat <i>responsive</i> sehingga terdapat ikon yang menyimpan daftar menu.

Tabel 4.3: Penjelasan kode konversi navigasi bar.

#### 4.3.4 Halaman Permintaan Cetak Transkrip

Perubahan halaman permintaan cetak transkrip dilampirkan dalam Kode 4.9.

Listing 4.9: Perubahan file \views\TranskripRequest\main.php

```

1 diff -r a\www\application\views\TranskripRequest\main.php
2   b\www\application\views\TranskripRequest\main.php
3
4 9a10
5 > <div class="container">
6 11,14c12,18
7 < <div class="medium-12 column">
8 <   <div class="callout">
9 <     <h5>Permohonan Baru</h5>
10 <     <div class="callout alert">Transkrip akademik sementara dapat diakses via
11       student portal masing-masing.</div>
12 --->
13 >   <div class="col">
14 >     <div class="card">
15 >       <div class="card-header">
16 >         Permohonan Baru
17 >       </div>
18 >       <div class="card-body">
19 >
20 19,22c23,25
21 <   <div class="large-4 column">
22 <     <label>Yang memohon:

```

```
22 <           <input type="email" name="requestByEmail" value="<?= $requestByEmail
23   ?>" readonly="readonly"/>
24 <         </label>
25 ---
26 > <div class="col-lg-4">
27 >     <label class="col-form-label">Yang memohon:</label>
28 >     <input class="form-control" type="email" name="requestByEmail" value="<?=
29   $requestByEmail ?>" readonly/>
30
31 24,27c27,29
32 < <div class="large-4 column">
33 <     <label>NPM:
34   <input type="text" value="<?= $requestByNPM ?>" readonly="readonly"/>
35 <     </label>
36 ---
37 > <div class="col-lg-4">
38 >     <label class="col-form-label">NPM:</label>
39 >     <input class="form-control" type="text" value="<?= $requestByNPM ?>"
40   readonly/>
41
42 29,32c31,34
43 < <div class="large-4 column">
44 <     <label>Nama:
45   <input type="text" name="requestByName" value="<?= $requestByName ?>" 
46   readonly="readonly"/>
47 <     </label>
48 ---
49 > <div class="col-lg-4">
50 >     <label class="col-form-label">Nama:</label>
51 >     <input class="form-control" type="text" name="requestByName" value="<?=
52   $requestByName ?>" readonly/>
53 >
54
55 36,38c38,40
56 < <div class="large-4 column">
57 <     <label>Tipe Transkrip:
58 <     <select name="requestType">
59 ---
60 > <div class="col-lg-4">
61 >     <label class="col-form-label">Tipe Transkrip:</label>
62 >     <select class="form-control" name="requestType">
63
64 45d46
65 <     </label>
66
67 47,50c48,50
68 < <div class="large-8 column">
69 <     <label>Keperluan:
70 <     <input type="text" name="requestUsage" required/>
71 <     </label>
72 ---
```

```
68 > <div class="col-lg-8">
69   >     <label class="col-form-label">Keperluan:</label>
70   >     <input class="form-control" type="text" name="requestUsage" required/>
71 
72 53c53,58
73   <           <button type="button" class="button primary" disabled>Kirim
74     Permohonan</button>
75   ---
76   > <br>
77   > <div class="row">
78   >   <div class="col-lg-12">
79     <input class="btn btn-primary" type="submit" class="button" value="Kirim
80       Permohonan">
81   >   </div>
82   > </div>
83 
84 60,62c66,73
85   < <div class="callout">
86     <h5>Histori Permohonan</h5>
87   <   <table class="stack">
88   ---
89   > </div>
90   > <br>
91   > <div class="card">
92     <div class="card-header">
93       Histori Permohonan
94     </div>
95     <div class="card-body">
96       <table class="table table-striped">
97 
98 65,71c76,82
99   <th>ID</th>
100  <th>Status</th>
101  <th>Tanggal Permohonan</th>
102  <th>Tipe Transkrip</th>
103  <th>Tanggal Jawab/Cetak</th>
104  <th>Keterangan</th>
105  <th>Aksi</th>
106  ---
107  > <th scope="col">ID</th>
108  > <th scope="col">Status</th>
109  > <th scope="col">Tanggal Permohonan</th>
110  > <th scope="col">Tipe Transkrip</th>
111  > <th scope="col">Tanggal Jawab/Cetak</th>
112  > <th scope="col">Keterangan</th>
113  > <th scope="col">Aksi</th>
114 
115 77,78c88,89
116  <td>#<?= $request->id ?></td>
117  <td><span class="<?= $request->labelClass ?> label"><?= $request->status
118    ?></span></td>
```

```
116 ---  
117 > <th>#<?= $request->id ?></th>  
118 > <td><span class="badge badge-<?= $request->labelClass ?>"><?= $request->status  
?></span></td>  
119  
120 84c95,98  
121 < <div class="reveal" id="detail<?= $request->id ?>" data-reveal>  
122 ---  
123 > <!-- Button trigger modal -->  
124 > <a data-toggle="modal" data-target="#lihatModal<?= $request->id ?>"  
125     id="detail<?= $request->id ?>">  
126 > <i class="fas fa-eye"></i>  
127 > </a>  
128  
129 85,86c100,111  
130 <     <h5>Detail Permohonan #<?= $request->id ?></h5>  
131 <     <table class="stack">  
132 ---  
133 > <!-- Modal -->  
134 > <div class="modal fade" id="lihatModal<?= $request->id ?>" tabindex="-1"  
135     role="dialog" aria-labelledby="exampleModalCenterTitle" aria-hidden="true">  
136 >   <div class="modal-dialog modal-dialog-centered" role="document">  
137 >     <div class="modal-content">  
138 >       <div class="modal-header">  
139 >         <h5 class="modal-title" id="exampleModalLongTitle">Detail  
140     Permohonan #<?= $request->id ?></h5>  
141 >         <button type="button" class="close" data-dismiss="modal"  
142     aria-label="Close">  
143 >           <span aria-hidden="true">&times;</span>  
144 >         </button>  
145 >       </div>  
146 >       <div class="modal-body">  
147 >         <table class="table ">  
148  
149 109c134  
150 < <th>Jawaban</th>  
151 ---  
152 > <th scope="col">Jawaban</th>  
153  
154 126,128c151,153  
155 < <button class="close-button" data-close aria-label="Tutup" type="button">  
156 <   <span aria-hidden="true">&times;</span>  
157 < </button>  
158 ---  
159 >       </div>  
160 >  
161 >     </div>  
162  
163 130c155  
164 < <a data-open="detail<?= $request->id ?>"><i class="fi-eye"></i></a>  
165 ---
```

```
162 >           </div>
```

Catatan untuk kode 4.9 dijabarkan dalam tabel 4.4:

Implementasi	Penjelasan
3-16	<b>Konten</b> Dalam Bootstrap 4 pembagian konten menggunakan .card yang terdiri dari .card-header dan card-body sedangkan di Foundation 6 hanya menggunakan kelas .callout. Lalu untuk form, field pada Foundation 6 hanya menggunakan tag <input> sedangkan pada Bootstrap 4 menggunakan kelas .form-control.
82-162	<b>Modal</b> Untuk modal pada Foundation 6 hanya memiliki satu bagian modal yaitu kelas .reveal sedangkan pada Bootstrap 4 menggunakan .modal yang terdiri dari .modal-fade, .modal-content, .modal-header.

Tabel 4.4: Penjelasan kode permintaan cetak transkrip.

### Controller Permintaan Cetak Transkrip

Pemberian label pada kolom 'status' menggunakan logika sederhana yang terletak pada controller, hasil dilampirkan dalam kode 4.10.

Listing 4.10: Perubahan file .php

```
1 diff -r a\www\application\controllers\TranskripRequest.php
      b\www\application\controllers\TranskripRequest.php
2 35c35
3 <   $request->labelClass = 'alert';
4 ---
5 >   $request->labelClass = 'danger';
```

### 4.3.5 Halaman Manajemen Cetak Transkrip

Pada kode 4.11 menjelaskan perubahan kode pada halaman manajemen permintaan transkrip.

### Halaman Utama

Listing 4.11: Perubahan file \views\TranskripManage\main.php

```
1 diff -r a\www\application\views\TranskripManage\main.php
      b\www\application\views\TranskripManage\main.php
2
3 10,12c10,15
4 < <div class="row">
5 <   <div class="callout">
6 <     <h5>Permintaan Transkrip</h5>
7 ---
8 >   <div class="container">
9 >     <div class="card">
10 >       <div class="card-header">
11 >         Permintaan Transkrip
12 >       </div>
13 >     <div class="card-body">
```

```
14
15 15c18,20
16 <   <span class="input-group-label">Cari NPM:</span>
17 ---
18 > <div class="input-group-prepend">
19 >   <span class="input-group-text">Cari NPM:</span>
20 > </div>
21
22 16,18c22,24
23 < <input name="npm" class="input-group-field" type="text"
24   placeholder="2013730013" maxlength="10" minlength="10" <?= $npmQuery === NULL
25   ? '' : " value='$npmQuery'" ?>/>
26 < <div class="input-group-button">
27 <   <input class="button" type="submit" value="Cari"/>
28 ---
29 > <input name="npm" class="form-control" type="text" placeholder="2013730013"
30   maxlength="10" minlength="10" <?= $npmQuery === NULL ? '' : "
31   value='$npmQuery'" ?>/>
32 > <div class="input-group-append">
33 >   <input class="btn btn-primary" type="submit" value="Cari"/>
34
35 22c28,29
36 < <table class="stack">
37 ---
38 > <br>
39 > <table class="table table-striped">
40 25,30c32,37
41 < <th>ID</th>
42 < <th>Status</th>
43 < <th>Tanggal Permohonan</th>
44 < <th>Tipe Transkrip</th>
45 < <th>NPM</th>
46 < <th>Aksi</th>
47 ---
48 > <th scope="col">ID</th>
49 > <th scope="col">Status</th>
50 > <th scope="col">Tanggal Permohonan</th>
51 37c44
52 < <td><span class="<?= $request->labelClass ?> label"><?= $request->status
53   ?></span></td>
54 ---
55 > <td><span class="badge badge-<?= $request->labelClass ?>"><?= $request->status
56   ?></span></td>
57
58 42,44c50,60
59 <   <div class="reveal" id="detail<?= $request->id ?>" data-reveal>
60 <     <h5>Detail Permohonan #<?= $request->id ?></h5>
```

```

59 <           <table class="stack">
60 ---
61 > <div class="modal fade" id="detail<?= $request->id ?>" tabindex="-1"
62   role="dialog" aria-hidden="true">
63 >   <div class="modal-dialog modal-dialog-centered" role="document">
64 >     <div class="modal-content">
65 >       <div class="modal-header">
66 >         <h5 class="modal-title" id="exampleModalLongTitle">Detail
67   Permohonan #<?= $request->id ?></h5>
68 >         <button type="button" class="close" data-dismiss="modal"
69   aria-label="Close">
70 >           <span aria-hidden="true">&times;</span>
71 >         </button>
72 >       </div>
73 >       <div class="modal-body">
74 >         <table class="table table-striped">
75
76 83a100,106
77 >           </div>
78 >         </div>
79 >       </div>
80 >     </div>
81 >   <a data-toggle="modal" data-target="#detail<?= $request->id ?>" id="detailIkon<?= $request->id ?>">
82   <i class="fas fa-eye"></i>
83 > </a>
84
85 84c108,113
86 <           <button class="close-button" data-close aria-label="Tutup"
87   type="button">
88 ---
89 > <div class="modal fade" id="tolak<?= $request->id ?>" tabindex="-1"
90   role="dialog" aria-hidden="true">
91 >   <div class="modal-dialog modal-dialog-centered" role="document">
92 >     <div class="modal-content">
93 >       <div class="modal-header">
94 >         <h5 class="modal-title" id="exampleModalLongTitle">Tolak
95   Permohonan #<?= $request->id ?></h5>
96 >         <button type="button" class="close" data-dismiss="modal"
97   aria-label="Close">
98
99 88,90c117
100 <   <a data-open="detail<?= $request->id ?>"><i class="fi-eye"></i></a>
101 <   <div class="reveal" id="tolak<?= $request->id ?>" data-reveal>
102 <     <h5>Tolak Permohonan</h5>
103 ---
104 >       <div class="modal-body">
105
106 95,100c122,129
107 <   <label>Email penjawab:
108 <     <input type="text" value="<?= $answeredByEmail ?>" readonly="true"/>

```

```
102 <      </label>
103 <      <label>Alasan penolakan:
104 <          <input name="answeredMessage" class="input-group-field" type="text"
105 required>
106 </label>
107 --->
108 <      <div class="form-group">
109 <          <label>Email penjawab:</label>
110 <          <input class="form-control" type="text" value=<?= $answeredByEmail
111 ?>" readonly="true"/>
112 </div>
113 <      <div class="form-group">
114 <          <label>Alasan penolakan:</label>
115 <          <input class="form-control" name="answeredMessage" type="text"
116 required/>
117 </div>
118 102c131,133
119 <      <input type="submit" class="alert button" value="Tolak"/>
120 --->
121 <      <div class="form-group">
122 <          <input type="submit" class="btn btn-danger" value="Tolak"/>
123 </div>
124 103a135,141
125 <          </div>
126 <      </div>
127 </div>
128 > <a data-toggle="modal" data-target="#tolak<?= $request->id ?>" id="detailIkon<?= $request->id ?>">
129 >     <i class="fas fa-thumbs-down"></i>
130 > </a>
131
132 104c143,148
133 <          <button class="close-button" data-close aria-label="Tutup"
134 type="button">
135 --->
136 < <div class="modal fade" id="cetak<?= $request->id ?>" tabindex="-1"
137 role="dialog" aria-hidden="true">
138 <     <div class="modal-dialog modal-dialog-centered" role="document">
139 <         <div class="modal-content">
140 <             <div class="modal-header">
141 <                 <h5 class="modal-title" id="exampleModalLongTitle">Cetak
142 Permohonan #<?= $request->id ?></h5>
143 >                 <button type="button" class="close" data-dismiss="modal"
144 aria-label="Close">
145
146 108,110c152
147 <     <a data-open="tolak<?= $request->id ?>"><i class="fi-dislike"></i></a>
148 <     <div class="reveal" id="cetak<?= $request->id ?>" data-reveal>
```

```

145 <           <h5>Cetak Permohonan</h5>
146 ---
147 >           <div class="modal-body">
148
149 121,126c163,170
150 <           <label>Email penjawab:
151 <             <input type="text" value=<?= $answeredByEmail ?>" readonly="true"/>
152 <           </label>
153 <           <label>Keterangan tambahan:
154 <             <input name="answeredMessage" class="input-group-field" type="text"
155   required/>
156 <           </label>
157 ---
158 >           <div class="form-group">
159 >             <label class="col-form-label">Email penjawab:</label>
160 >             <input class="form-control" type="text" value=<?= $answeredByEmail
161   ?>" readonly="true"/>
162 >           </div>
163 >           <div class="form-group">
164 >             <label class="col-form-label">Keterangan tambahan:</label>
165 >             <input class="form-control" name="answeredMessage" type="text"
166   required/>
167 >           </div>
168
169 128c172,174
170 <           <input type="submit" class="button" value="Sudah dicetak"/>
171 ---
172 >           <div class="form-group">
173 >             <input class="btn btn-primary" type="submit" class="button"
174   value="Sudah dicetak"/>
175 >           </div>
176
177 129a176,182
178 >           </div>
179 >           </div>
180 >           </div>
181 >           <a data-toggle="modal" data-target="#cetak<?= $request->id ?>" id="detailIkon<?= $request->id ?>">
182 >             <i class="fas fa-print"></i>
183 >           </a>
184
185 130c184,189
186 <           <button class="close-button" data-close aria-label="Tutup"
187   type="button">
188 ---
189 >           <div class="modal fade" id="hapus<?= $request->id ?>" tabindex="-1"
190   role="dialog" aria-hidden="true">
191 >             <div class="modal-dialog modal-dialog-centered" role="document">
192 >               <div class="modal-content">
193 >                 <div class="modal-header">

```

```

189 >           <h5 class="modal-title" id="exampleModalLongTitle">Hapus
190   Permohonan #<?= $request->id ?></h5>
191 >           <button type="button" class="close" data-dismiss="modal"
192   aria-label="Close">
193 134,136c193
194 <     <a data-open="cetak<?= $request->id ?>"><i class="fi-print"></i></a>
195 <     <div class="reveal" id="hapus<?= $request->id ?>" data-reveal>
196   <h5>Hapus Permohonan</h5>
197 -----
198 >           <div class="modal-body">
199 143c200
200 <     <input type="submit" class="alert button" value="Hapus"/>
201 -----
202 >     <input class="btn btn-danger" type="submit" value="Hapus"/>
203
204 145,147c202,204
205 < <button class="close-button" data-close aria-label="Tutup" type="button">
206   <span aria-hidden="true">&times;</span>
207 < </button>
208 -----
209 >           </div>
210 >     </div>
211 > </div>
212
213 149c206,208
214 <     <a data-open="hapus<?= $request->id ?>"><i class="fi-trash"></i></a>
215 -----
216 >     <a data-toggle="modal" data-target="#hapus<?= $request->id ?>" id="detailIkon<?= $request->id ?>">
217   <i class="fas fa-trash"></i>
218 >     </a>

```

Catatan untuk kode 4.8 dijabarkan dalam tabel 4.5:

No	Implementasi	Penjelasan
3-13	<b>Border untuk konten</b>	Dalam Bootstrap 4 pembagian konten menggunakan .card yang terdiri dari .card-header dan card-body sedangkan di Foundation 6 hanya menggunakan kelas .callout. Lalu untuk form, field pada Foundation 6 hanya menggunakan tag <input> sedangkan pada Bootstrap 4 menggunakan kelas .form-control.
51-54	<b>Label</b>	Pemanggilan label untuk jenis aksi yang dilakukan menggunakan controller dari manajemen cetak transkrip (Penjelasan ada pada kode 4.12).
56-218	<b>Modal</b>	Lalu terdapat <i>input group field</i> , dimana pada Bootstrap 4 setiap tag span diikuti kelas .input-group-prepend

Tabel 4.5: Penjelasan kode konversi manajemen cetak transkrip.

### Controller Manajemen Cetak Transkrip

Pemberian label pada kolom 'status' menggunakan logika sederhana yang terletak pada controller, hasil dilampirkan dalam kode 4.12.

Listing 4.12: Controller Manajemen Cetak Transkrip

```

1  diff -r
2  a\www\application\controllers\TranskripManage.php
   b\www\application\controllers\TranskripManage.php
3  49c49
4  < $request->labelClass = 'alert';
5  ---
6  > $request->labelClass = 'danger';

```

### 4.3.6 Halaman Permintaan Perubahan Kuliah

Catatan untuk kode 4.13 dijabarkan dalam tabel ??:

Listing 4.13: Perubahan file \views\PerubahanKuliahRequest\main.php

```

1  diff -r a\www\application\views\PerubahanKuliahRequest\main.php
   b\www\application\views\PerubahanKuliahRequest\main.php
2
3  10,15c11,17
4  < <div class="row">
5  <     <div class="large-12 column">
6  <         <div class="callout">
7  <             <h5>Permohonan Baru</h5>
8  <             <div class="callout alert">Untuk perubahan jadwal kuliah, silakan
9                 berkoordinasi langsung dengan peserta kuliah.</div>
9  <             <form method="POST" action="/PerubahanKuliahRequest/add">
10    ---
11 > <div class="container">
12 >     <div class="card">
13 >         <div class="card-header">
14 >             Permohonan Baru
15 >         </div>
16 >         <div class="card-body">
17 >             <form class="p-3" method="POST"
18 >               action="/PerubahanKuliahRequest/add">
19
19 17,21c19,22
20 < <div class="row">
21 <     <div class="large-4 column">
22 <         <label>Pemohon:
23 <             <input type="email" name="requestByEmail" value=<?=
24 <               $requestByEmail ?>" readonly="readonly"/>
24 <         </label>
25    ---
26 > <div class="form-group row">
27 >     <div class="col-sm-6">
28 >         <label class="col-form-label">Pemohon:</label>

```

```
29 >           <input class="form-control" type="email" name="requestByEmail"
   | value="<?= $requestByEmail ?>" readonly/>
30
31 23,26c24,26
32 <     <div class="large-8 column">
33 <       <label>Nama:
34 <         <input type="text" name="requestByName" value="<?= $requestByName
   | ?>" readonly="readonly"/>
35 <       </label>
36 ---
37 >     <div class="col-sm-6">
38 >       <label class="col-form-label">Nama:</label>
39 >       <input class="form-control" type="text" name="requestByName"
   | value="<?= $requestByName ?>" readonly="readonly"/>
40
41 29,33c30,33
42 < <div class="row">
43 <   <div class="large-2 column">
44 <     <label>Kode MK:
45 <       <input type="text" name="mataKuliahCode" required maxlength="9"
   | pattern="[A-Z]{3}[0-9]{3}([0-9]{3})?" title="Kode MK dalam format XYZ123"/>
46 <     </label>
47 ---
48 > <div class="form-group row">
49 >   <div class="col-sm-2">
50 >     <label class="col-form-label">Kode MK:</label>
51 >     <input class="form-control" type="text" name="mataKuliahCode" required
   | maxlength="9" pattern="[A-Z]{3}[0-9]{3}([0-9]{3})?" title="Kode MK dalam
   | format XYZ123"/>
52
53 35,38c35,37
54 <   <div class="large-5 column">
55 <     <label>Nama Mata Kuliah:
56 <       <input type="text" name="mataKuliahName" required/>
57 <     </label>
58 ---
59 >   <div class="col-sm-5">
60 >     <label class="col-form-label">Nama Mata Kuliah:</label>
61 >     <input class="form-control" type="text" name="mataKuliahName"
   | required/>
62
63 40,43c39,41
64 <   <div class="large-1 column">
65 <     <label>Kelas:
66 <       <input type="text" name="class" maxlength="1"/>
67 <     </label>
68 ---
69 >   <div class="col-sm-1">
70 >     <label class="col-form-label">Kelas:</label>
71 >     <input class="form-control" type="text" name="class" maxlength="1"/>
72
```

```
73 45,47c43,45
74 <      <div class="large-4 column">
75 <          <label>Jenis Perubahan:
76 <              <select name="changeType">
77 ---
78 >      <div class="col-sm-4">
79 >          <label class="col-form-label">Jenis Perubahan:</label>
80 >          <select name="changeType" class="form-control">
81
82 52d49
83 <          </label>
84
85 55,59c52,55
86 < <div class="row">
87 <     <div class="large-3 column">
88 <         <label>Dari Hari & Jam:
89 <             <input class="disableable" type="text" name="fromDateTime"
90   id="fromDateTime"/>
91 <         </label>
92 ---
93 > <div class="form-group row">
94 >     <div class="col-sm-3">
95 >         <label class="col-form-label">Dari Hari & Jam:</label>
96 >         <input id="datetimepicker" class="form-control disableable"
97   type="text" name="fromDateTime">
98
99 61,64c57,59
100 <     <div class="large-3 column">
101 <         <label>Dari Ruang:
102 <             <input class="disableable" type="text" name="fromRoom"/>
103 <         </label>
104 ---
105 >     <div class="col-sm-3">
106 >         <label class="col-form-label">Dari Ruang:</label>
107 >         <input class="form-control disableable" type="text" name="fromRoom"/>
108
109 66,69c61,63
110 <     <div class="large-6 column">
111 <         <label>Keterangan Tambahan:
112 <             <input class="disableable" type="text" name="remarks"/>
113 <         </label>
114 ---
115 >     <div class="col-sm-6">
116 >         <label class="col-form-label">Keterangan Tambahan:</label>
117 >         <input class="form-control disableable" type="text" name="remarks"/>
118
119 72,76c66,69
120 < <div class="row toFields">
121 <     <div class="large-3 column">
122 <         <label>Menjadi Hari & Jam:
123 <             <input class="disableable toDateTime" type="text"
```

```
    name="toDateTime[]"/>
122 <      </label>
123 ---
124 > <div class="form-group row toFields">
125 >     <div class="col-sm-3">
126 >         <label class="col-form-label">Menjadi Hari & Jam:</label>
127 >         <input id="datetimetypepicker" class="form-control disableable toDateTime"
128 type="text" name="toDateTime[]"/>
129 78,81c71,73
130 <     <div class="large-3 column">
131 <         <label>Menjadi Ruang:
132 <             <input class="disableable toRoom" type="text" name="toRoom[]"/>
133 <         </label>
134 ---
135 >     <div class="col-sm-3">
136 >         <label class="col-form-label">Menjadi Ruang:</label>
137 >         <input class="form-control disableable toRoom" type="text"
138 name="toRoom[]"/>
139 83,85c75,77
140 <     <div class="large-6 column">
141 <         <br/>
142 <         <a href="#" class="eraseButton button secondary">Hapus</a>
143 ---
144 >     <div class="col-sm-2">
145 >         <br/><br>
146 >         <a href="#" class="eraseButton btn btn-secondary">Hapus</a>
147
148 88,91c80,83
149 < <div class="row" id="sendDiv">
150 <     <div class="large-12 column">
151 <         <button type="button" class="button primary" disabled>Kirim
152 <             Permohonan</button>
153 <         <a href="#" id="addToButton" class="button secondary">Tambah Pertemuan
154 <             Ekstra</a>
155 ---
156 > <div class="form-group row" id="sendDiv">
157 >     <div class="col-sm-12">
158 >         <input type="submit" class="btn btn-primary" value="Kirim Permohonan">
159 >         <a href="#" id="addToButton" class="btn btn-secondary">Tambah
160 <             Pertemuan Ekstra</a>
161
162 96,98c88,95
163 < <div class="callout">
164 <     <h5>Histori Permohonan</h5>
165 <     <table class="stack">
166 ---> </div>
167 > <br>
168 > <div class="card">
```

```

167 >     <div class="card-header">
168 >         Histori Permohonan
169 >     </div>
170 >     <div class="card-body">
171 >         <table class="table table-striped table-responsive">
172
173 101,108c98,105
174 < <th>ID</th>
175 < <th>Status</th>
176 < <th>Tanggal Permohonan</th>
177 < <th>Kode MK</th>
178 < <th>Perubahan</th>
179 < <th>Tanggal Jawab</th>
180 < <th>Keterangan</th>
181 < <th>Aksi</th>
182 ---
183 > <th scope="col">ID</th>
184 > <th scope="col">Status</th>
185 > <th scope="col">Tanggal Permohonan</th>
186 > <th scope="col">Kode MK</th>
187 > <th scope="col">Perubahan</th>
188 > <th scope="col">Tanggal Jawab</th>
189 > <th scope="col">Keterangan</th>
190 > <th scope="col">Aksi</th>
191
192 115c112
193 < <td><span class=<?=$request->labelClass ?> label"><?=$request->status
194   ?></span></td>
195 ---
196 > <td><span class=" badge badge-<?=$request->labelClass ?>"><?=$
197   $request->status ?></span></td>
198
199 122c119,123
200 < <a data-open="detail<?=$request->id ?>"><i class="fi-eye"></i></a>
201 ---
202 > <a data-toggle="modal" data-target="#detail<?=$request->id ?>""
203   id="detailIkon<?=$request->id ?>">
204 > <span style="font-size: 18px; color: Dodgerblue;">
205 >   <i class="fas fa-eye"></i>
206 > </span>
207 > </a>
208
209 128a130
210 > <h5></h5>
211
212 132,135c135,145
213 <
214 < <div class="reveal" id="detail<?=$request->id ?>" data-reveal>
215 <   <h5>Detail Permohonan #<?=$request->id ?></h5>
216 <   <table class="stack">
217 ---
```

```

215 > <div class="modal fade" id="detail<?= $request->id ?>" tabindex="-1"
216   role="dialog" aria-hidden="true">
217 >   <div class="modal-dialog modal-dialog-centered" role="document">
218 >     <div class="modal-content">
219 >       <div class="modal-header">
220 >         <h5 class="modal-title" id="exampleModalLongTitle">Detail
221 >           Permohonan #<?= $request->id ?></h5>
222 >           <button type="button" class="close" data-dismiss="modal"
223 >             aria-label="Close">
224 >             <span aria-hidden="true">&times;;</span>
225 >           </button>
226 >         </div>
227 >         <div class="modal-body">
228 >           <table class="table table-striped">
229 >             201,203c211,213
230 <           <button class="close-button" data-close aria-label="Tutup" type="button">
231 <             <span aria-hidden="true">&times;;</span>
232 <           </button>
233 >           <!--
234 >             </div>
235 >           </div>
236 >         </div>
237 >         214a225
238 >         > jQuery('#datetimetypepicker').datetimepicker();

```

No	Implementasi	Penjelasan
3-17	<b>Card</b>	Pembagian konten menggunakan .card yang terdiri dari .card-header dan card-body sedangkan di Foundation 6 hanya menggunakan kelas .callout..
51-54	<b>Form</b>	Field pada Foundation 6 hanya menggunakan tag <input> sedangkan pada Bootstrap 4 menggunakan kelas .form-control, dimana pada Bootstrap 4 setiap tag span diikuti kelas .input-group-prepend.
56-218	<b>Modal</b>	Pada Foundation 6 hanya memiliki satu bagian modal yaitu kelas .reveal sedangkan pada Bootstrap 4 menggunakan .modal yang terdiri dari .modal-fade, modal-content, modal-header.
173-183	<b>Tabel</b>	Pada foundation 6 pada tag <th> tidak diikuti kelas, namun pada Bootstrap 4 perlu menggunakan atribut scope=col.

Tabel 4.6: Penjelasan kode permintaan perubahan kuliah.

### Controller Permintaan Perubahan Kuliah

Pemberian label pada kolom 'status' menggunakan logika sederhana yang terletak pada controller, hasil dilampirkan dalam kode 4.14.

Listing 4.14: Controller Request Perubahan Kuliah

```

1 diff -r a\www\application\controllers\PerubahanKuliahRequest.php
2   b\www\application\controllers\PerubahanKuliahRequest.php

```

```

3 |34c34
4 |< $request->labelClass = 'alert';
5 |---
6 |> $request->labelClass = 'danger';

```

#### 4.3.7 Halaman Manajemen Perubahan Kuliah

Kode 4.15 menjabarkan perubahan halaman manajemen perubahan kuliah:

Listing 4.15: Perubahan file \views\PerubahanKuliahManage\main.php

```

1 diff -r a\www\application\views\PerubahanKuliahManage\main.php
2   b\www\application\views\PerubahanKuliahManage\main.php
3
4 10,13c10,17
5 < <div class="row">
6 <   <div class="callout">
7 <     <h5>Permohonan Perubahan Kuliah</h5>
8 <     <table class="stack">
9 --- 
10 > <div class="container">
11 >   <div class="card">
12 >     <div class="card-header">
13 >       Permohonan Perubahan Kuliah
14 >     </div>
15 >     <br>
16 >     <div class="card-body">
17 >       <table class="table table-striped">
18
19 16,21c20,25
20 < <th>ID</th>
21 < <th>Status</th>
22 < <th>Tanggal Permohonan</th>
23 < <th>Kode MK</th>
24 < <th>Perubahan</th>
25 < <th>Aksi</th>
26 --- 
27 > <th scope="col">ID</th>
28 > <th scope="col">Status</th>
29 > <th scope="col">Tanggal Permohonan</th>
30 > <th scope="col">Kode MK</th>
31 > <th scope="col">Perubahan</th>
32 > <th scope="col">Aksi</th>
33
34 28c32
35 < <td><span class="<?= $request->labelClass ?> label"><?= $request->status
36 ?></span></td>
37 --- 
38 > <td><span class="badge badge-<?= $request->labelClass ?>"><?= $request->status
39 ?></span></td>
40
41 33,37c37,41

```

```
39 < <a data-open="detail<?= $request->id ?>"><i class="fi-eye"></i></a>
40 < <a target="_blank" href="/PerubahanKuliahManage/printview/<?= $request->id
    ?>"><i class="fi-print"></i></a>
41 < <a data-open="konfirmasi<?= $request->id ?>"><i class="fi-like"></i></a>
42 < <a data-open="tolak<?= $request->id ?>"><i class="fi-dislike"></i></a>
43 < <a data-open="hapus<?= $request->id ?>"><i class="fi-trash"></i></a>
44 ---
45 > <a data-toggle="modal" data-target="#detail<?= $request->id ?>" id="detailIkon<?= $request->id ?>"><i class="fas fa-eye blueiconcolor"></i></a>
46 > <a target="_blank" href="/PerubahanKuliahManage/printview/<?= $request->id
    ?>"><i class="fas fa-print"></i></a>
47 > <a data-toggle="modal" data-target="#konfirmasi<?= $request->id ?>"><i
    class="fas fa-thumbs-up"></i></a>
48 > <a data-toggle="modal" data-target="#tolak<?= $request->id ?>"><i class="fas
    fa-thumbs-down"></i></a>
49 > <a data-toggle="modal" data-target="#hapus<?= $request->id ?>"><i class="fas
    fa-trash"></i></a>
50
51 57,60c61,71
52 <
53 <     <div class="reveal" id="detail<?= $request->id ?>" data-reveal>
54 <         <h5>Detail Permohonan #<?= $request->id ?></h5>
55 <         <table class="stack">
56 ---
57 > <div class="modal fade" id="detail<?= $request->id ?>" tabindex="-1"
    role="dialog" aria-hidden="true">
58 >   <div class="modal-dialog modal-dialog-centered" role="document">
59 >     <div class="modal-content">
60 >       <div class="modal-header">
61 >         <h5 class="modal-title" id="exampleModalLongTitle">Detail
62 >           Permohonan #<?= $request->id ?></h5>
63 >         <button type="button" class="close" data-dismiss="modal"
64 >           aria-label="Close">
65 >           <span aria-hidden="true">&times;</span>
66 >         </button>
67 >       </div>
68 >       <div class="modal-body">
69 >         <table class="table table-striped">
70 >           126c137,146
71 <             <button class="close-button" data-close aria-label="Tutup"
72 >               type="button">
73 >             </div>
74 >           </div>
75 >         </div>
76 > <div class="modal fade" id="konfirmasi<?= $request->id ?>" tabindex="-1"
    role="dialog" aria-hidden="true">
77 >   <div class="modal-dialog modal-dialog-centered" role="document">
```

```
78 >      <div class="modal-content">
79 >          <div class="modal-header">
80 >              <h5 class="modal-title" id="exampleModalLongTitle">Konfirmasi
81 >                  Permohonan #<?= $request->id ?></h5>
82 >                  <button type="button" class="close" data-dismiss="modal"
83 >                      aria-label="Close">
84
85 130,131c150
86 <      <div class="reveal" id="konfirmasi<?= $request->id ?>" data-reveal>
87 <          <h5>Konfirmasi Permohonan</h5>
88 ---
89 >          <div class="modal-body">
90
91 136,141c155,162
92 <          <label>Email penjawab:
93 <              <input type="text" value=<?= $answeredByEmail ?>" readonly="true"/>
94 <          </label>
95 <          <label>Keterangan:
96 <              <input name="answeredMessage" class="input-group-field" type="text"/>
97 <          </label>
98 ---
99 >          <div class="form-group">
100 >              <label>Email penjawab:</label>
101 >              <input class="form-control" type="text" value=<?= $answeredByEmail ?>" readonly="true"/>
102 >          </div>
103 >          <div class="form-group">
104 >              <label>Keterangan:</label>
105 >              <input class="form-control" name="answeredMessage"
106 >                  class="input-group-field" type="text"/>
107 >          </div>
108 ---
109 >          <input type="submit" class="success button" value="Konfirmasi"/>
110 ---
111 >          <div class="form-group">
112 >              <input type="submit" class="btn btn-success" value="Konfirmasi"/>
113 >          </div>
114
115 143c164,166
116 <          <button class="close-button" data-close aria-label="Tutup"
117 >              type="button">
118 ---
119 >          </div>
120 >      </div>
121 >      </div>
122 >      <div class="modal fade" id="tolak<?= $request->id ?>" tabindex="-1"
123 >          role="dialog" aria-hidden="true">
124 >          <div class="modal-dialog modal-dialog-centered" role="document">
125 >              <div class="modal-content">
```

```
123 >         <div class="modal-header">
124 >             <h5 class="modal-title" id="exampleModalLongTitle">Tolak
125 >                 Permohonan #<?= $request->id ?></h5>
126 >                 <button type="button" class="close" data-dismiss="modal"
127 >                     aria-label="Close">
128 149,150c181
129 <         <div class="reveal" id="tolak<?= $request->id ?>" data-reveal>
130 <             <h5>Tolak Permohonan</h5>
131 --->
132 >         <div class="modal-body">
133 155,160c186,193
134 <     <label>Email penjawab:
135 <     <input type="text" value="<?= $answeredByEmail ?>" readonly="true"/>
136 <   </label>
137 <     <label>Alasan penolakan:
138 <     <input name="answeredMessage" class="input-group-field" type="text"
139     required/>
140 <   </label>
141 --->
142 >     <div class="form-group">
143 >         <label>Email penjawab:</label>
144 >         <input class="form-control" type="text" value="<?= $answeredByEmail ?>"
145 >             readonly="true"/>
146 >     </div>
147 >     <div class="form-group">
148 >         <label>Alasan penolakan:</label>
149 >         <input class="form-control" name="answeredMessage"
150 >             class="input-group-field" type="text" required/>
151 >     </div>
152 162c195,197
153 <             <input type="submit" class="alert button" value="Tolak"/>
154 --->
155 >     <div class="form-group">
156 >         <input type="submit" class="btn btn-danger" value="Tolak"/>
157 >     </div>
158 164c200,209
159 <             <button class="close-button" data-close aria-label="Tutup"
160 >                 type="button">
161 --->
162 >     </div>
163 >     </div>
164 >     <div class="modal fade" id="hapus<?= $request->id ?>" tabindex="-1"
165 >         role="dialog" aria-hidden="true">
166 >         <div class="modal-dialog modal-dialog-centered" role="document">
167 >             <div class="modal-content">
```

```

167 >           <div class="modal-header">
168 >             <h5 class="modal-title" id="exampleModalLongTitle">Hapus
169 >               Permohonan #<?= $request->id ?></h5>
170 >             <button type="button" class="close" data-dismiss="modal"
171 >               aria-label="Close">
172
173 168,169c213
174 < <div class="reveal" id="hapus<?= $request->id ?>" data-reveal>
175 < <h5>Hapus Permohonan</h5>
176 ---
177 >           <div class="modal-body">
178
179 176c220
180 <     <input type="submit" class="alert button" value="Hapus"/>
181 ---
182 >           <input type="submit" class="btn btn-danger" value="Hapus"/>
183
184 178,180c222,224
185 < <button class="close-button" data-close aria-label="Tutup" type="button">
186 <   <span aria-hidden="true">&times;</span>
187 < </button>
188 ---
189 >         </div>
190 >       </div>
191 >     </div>
192 >   </div>
193 > </div>

```

Keterangan darikode diatas tertera pada tabel 4.7

No	Implementasi	Penjelasan
3-17	<b>Card</b>	Pembagian konten menggunakan .card yang terdiri dari .card-header dan card-body sedangkan di Foundation 6 hanya menggunakan kelas .callout.
51-54	<b>Form</b>	Field pada Foundation 6 hanya menggunakan tag <input> sedangkan pada Bootstrap 4 menggunakan kelas .form-control, dimana pada Bootstrap 4 setiap tag span diikuti kelas .input-group-prepend.
56-218	<b>Modal</b>	Pada Foundation 6 hanya memiliki satu bagian modal yaitu kelas .reveal sedangkan pada Bootstrap 4 menggunakan .modal yang terdiri dari .modal-fade, .modal-content, .modal-header.
173-183	<b>Tabel</b>	Pada foundation 6 pada tag <th> tidak diikuti kelas, namun pada Bootstrap 4 perlu menggunakan atribut scope=col.

Tabel 4.7: Penjelasan kode manajemen perubahan kuliah.

#### 4.3.8 Halaman Entri Jadwal Dosen

Kode 4.17 menjabarkan perubahan halaman entri jadwal dosen:

Listing 4.16: Kode untuk Halaman Entri Jadwal Dosen

```
1 diff -r a\www\application\views\EntriJadwalDosen\main.php
2   b\www\application\views\EntriJadwalDosen\main.php
3
4 11a12,17
5 > <div class="container">
6 >   <div class="card">
7 >     <div class="card-header">
8 >       Tambah Jadwal
9 >     </div>
10 >   <div class="card-body">
11
12 14,16c19
13 < <div class="large-12 column callout">
14 <   <h5>Tambah Jadwal</h5>
15 < <div class="large-4 columns">
16 <   <div class="col-lg-4">
17
18 20c23
19 < <select name="hari">
20 <   <option value="1">Senin</option>
21 > <select class="form-control" name="hari">
22
23 33c36
24 < <select name="jam_mulai">
25 <   <option value="0600">06.00</option>
26 > <select class="form-control" name="jam_mulai">
27
28 39c42
29 < <div class=" large-4 columns">
30 <   <div class="col-lg-4">
31 > <div class="col-lg-4">
32
33 41c44
34 <   <select name="durasi">
35 <   <option value="0030">30 menit</option>
36 > <select class="form-control" name="durasi">
37
38 47c50
39 < <select name="jenis_jadwal">
40 <   <option value="1">Reguler</option>
41 > <select class="form-control" name="jenis_jadwal">
42
43 51c54
44 <   </select>
45 <   <br>
46 > </select><br>
47
48 53,56c56,59
49 < <div class="large-4 columns">
50 <   <Label <input type="text" name="label_jadwal"><br>
```

```

51 <      <input type="submit" class="button" value="Tambah">
52 <    </form>
53 ---
54 > <div class="col-lg-4">
55 >   Label <input class="form-control" type="text" name="label_jadwal"><br><br>
56 >   <input class="btn btn-primary" type="submit" value="Tambah">
57 > </form><br>
58
59 60,64c65,72
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 ---
66 > <br>
67 > <div class="card">
68 > <div class="card-header">
69 >   Daftar Jadwal
70 > </div>
71 > <div class="card-body">
72 >   <div id="jadwal_table">
73 >     <table class="table table-bordered table-striped">
74
75 66c74
76 < <td style='width:10%'></td>
77 ---
78 > <th></th>
79
80 69c77
81 < echo "<td style='width:18%'> $namaHari[$i] </td>"; //Membuat Header Tabel yang
   berisi daftar hari
82 ---
83 > echo "<th> $namaHari[$i] </th>"; //Membuat Header Tabel yang berisi daftar hari
84
85 77c85
86 < echo "<tr><td>" . $i . "-" . ($i + 1);
87 ---
88 > echo "<tr><th>" . $i . "-" . ($i + 1);
89
90 91a100
91 > $border = "border border-secondary align-middle";
92
93 106a116
94 > $($cellLocation).addClass('<?php echo $border; ?>');
95
96 117c127
97 < $($menuName).foundation('open');
98 ---
99 > $($menuName).modal();
100

```

```
101 | 149,150c159,160
102 | < a href="/EntriJadwalDosen/deleteAll/export/" class="alert button"
103 |   onClick="return konfirmasi();">Delete All</a>
104 | < a href="/EntriJadwalDosen/export/" class="button">Ekspor ke XLS</a>
105 | ---
106 | > <a href="/EntriJadwalDosen/deleteAll/export/" class="btn btn-danger"
107 |   onClick="return konfirmasi();">Delete All</a>
108 | > <a href="/EntriJadwalDosen/export/" class="btn btn-primary">Ekspor ke XLS</a>
109 |
110 |
111 |
112 | 169,170c179,184
113 | < <div id="edit_menu<?php echo $dataHariIni->id ?>" class="reveal" data-reveal >
114 | <     <button class="close-button" data-close aria-label="Close modal"
115 |       type="button">
116 |     ---
117 | > <div class="modal fade" id="edit_menu<?php echo $dataHariIni->id ?>" tabindex="-1" role="dialog" aria-hidden="true">
118 |   <div class="modal-dialog modal-dialog-centered" role="document">
119 |     <div class="modal-content">
120 |       <div class="modal-header">
121 |         <h5 class="modal-title" id="exampleModalLongTitle">Edit Jadwal
122 | #<?= $dataHariIni->id ?></h5>
123 |       <button type="button" class="close" data-dismiss="modal"
124 |         aria-label="Close">
125 |
126 |
127 | 179c193
128 | < <select name="hari">
129 |   ---
130 | > <select class="form-control" name="hari">
131 |
132 |
133 | 197c211
134 | < <select name="jam_mulai">
135 |   ---
136 | > <select class="form-control" name="jam_mulai">
137 |
138 |
139 | 213c227
140 | < <select name="durasi">
141 |   ---
142 | > <select class="form-control" name="durasi">
143 |
144 |
145 | 229c243
146 | < <select name="jenis_jadwal">
147 |   ---
148 | > <select class="form-control" name="jenis_jadwal">
149 |
150 |
151 | 233,238c247,252
152 | < </select>
153 | < Label <input type="text" name="label_jadwal" value="<?php echo
154 |   $dataHariIni->label; ?>"><br>
155 | < <div class="row large-4 column">
156 | <   <div class="large-2 column">
157 | <     <input type="submit" name="submitId<?php echo $dataHariIni->id ?>"
```

```

    class="button" value="Save  ">
145 <  </form>
146 ---
147 > </select><br>
148 > Label
149 > <input class="form-control" type="text" name="label_jadwal" value=<?php echo
150   $dataHariIni->label; ?>"><br>
151 > <div class="row">
152   > <div class="col-lg-2">
153     <input class="btn btn-primary" type="submit" name="submitId<?php echo
154       $dataHariIni->id ?>" class="button" value="Save  ">
155
156 240c254
157 < <div class="large-2 column ">
158 ---
159 > <div class="col-lg-2">
160
161 243,244c257,258
162 <   <input type="submit" id="deletebtn<?php echo $dataHariIni->id ?>" 
163   name="deletebtn<?php echo $dataHariIni->id ?>" class="alert button"
164   value="Delete">
165 < </form><div>
166 ---
167 >   <input class="btn btn-danger" type="submit" id="deletebtn<?php echo
168   $dataHariIni->id ?>" name="deletebtn<?php echo $dataHariIni->id ?>" 
169   value="Delete">
170 > </form>

```

Beberapa catatan dari kode diatas tertera pada tabel 4.8:

No	Implementasi	Penjelasan
80-94	<b>Edit background tabel</b>	Pada Bootstrap 4 diatur lebar tabel dan garis yang mengelilingi <i>cell</i> yang berisi jadwal kuliah.
96-99	<b>Modal</b>	Memanggil jQuery saat jadwal yang terisi dipilih user.

Tabel 4.8: Penjelasan kode permintaan perubahan kuliah.

#### 4.3.9 Halaman Lihat Jadwal Dosen

Kode 4.17 menjabarkan perubahan halaman lihat jadwal dosen:

Listing 4.17: Kode untuk Halaman Lihat Jadwal Dosen

```

1 diff -r a\www\application\views\LihatJadwalDosen\main.php
2   b\www\application\views\LihatJadwalDosen\main.php
3
4 12c12,13
5 <
6 ---
7 > <div class="container">
8   >   <div class="card card-body">
9
10 15,16c15,16

```

```
10 < <div class="large-12 column">
11 < <ul class="tabs" data-tabs id="tab_jadwal">
12 ---
13 > <div class="col-lg-12">
14 > <ul class="nav nav-tabs" data-tabs id="tab_jadwal">
15
16 22c22
17 < <li class="tabs-title is-active"><a href="#hal<?php echo $idx; ?>" aria-selected="true"><?php foreach ($currRow as $data) {
18 ---
19 > <li class="nav-item"><a class="nav-link active" href="#hal<?php echo $idx; ?>" aria-selected="true"><?php foreach ($currRow as $data) {
20
21 29c29
22 < <li class="tabs-title"><a a href="#hal<?php echo $idx; ?>"><?php foreach ($currRow as $data) {
23 ---
24 > <li class="nav-item"><a class="nav-link" href="#hal<?php echo $idx; ?>"><?php foreach ($currRow as $data) {
25
26 47c47
27 < <div class="table-scroll" id="jadwal_table<?php echo $idx; ?>">
28 ---
29 > <div id="jadwal_table<?php echo $idx; ?>">
30
31 59c59
32 < <table id="tabel<?php echo $idx; ?>" border=1 style="border-color:black ; border-collapse:separate">
33 ---
34 > <table class="table table-bordered table-striped" id="tabel<?php echo $idx; ?>">
35
36 66c66
37 < echo "<td style='width:18%'>" . $namaHari[$i] . "</td>";
38 ---
39 > echo "<th>" . $namaHari[$i] . "</th>";
40
41 73c73
42 < echo "<tr><td>" . $i . "-" . ($i + 1);
43 ---
44 > echo "<tr><th>" . $i . "-" . ($i + 1);
45
46 89a90
47 > $border = "border border-secondary align-middle";
48
49 104a106
50 > $($cellLocation).addClass('<?php echo $border; ?>');
51
52 140c141
53 < <a href="/LihatJadwalDosen/export/" class="button">Ekspor ke XLS</a>
54 ---
```

55 > <a class="btn btn-primary" href="/LihatJadwalDosen/export/"  
class="button">Ekspor ke XLS</a>

Keterangan dari kode diatas tertera pada tabel 4.9:

No	Implementasi	Penjelasan
9-24	<b>Tabs</b>	Tabs pada Foundation 6 dan Bootstrap 4 hanya berbeda penamaan nya saja.
31-44	<b>Edit background tabel</b>	Pada Bootstrap 4 diatur lebar tabel dan garis yang mengeliminasi <i>cell</i> yang berisi jadwal kuliah.

Tabel 4.9: Penjelasan kode permintaan perubahan kuliah.

## **DAFTAR REFERENSI**

- [1] Foundation 6 (2017) *Foundation v6.3.1 Documentation*. Zurb Inc. 100 W Rincon Ave, Campbell, CA 95008, USA.
- [2] Bootstrap 4 (2019) *Bootstrap v4.3.1 Documentation*. Bootstrap Core Team. 77 Massachusetts Avenue Cambridge, MA 02139, USA.
- [3] BlueTape (2019) *UserGuide BlueTape*. Fakultas Teknologi Informasi dan Sains (FTIS) Universitas Katolik Parahyangan. Bandung, Jawa Barat.
- [4] Codeigniter 3 (2017) *Codeigniter 3.1.4 Documentation*. British Columbia Institute of Technology. 5112 Bobby Hicks Hwy, Gray, TN 37615, USA.
- [5] jQuery (2020) *jQuery*. The jQuery Team. Brooklyn, NY.