

# SKRIPSI

KONVERSI ZURB FOUNDATION KE BOOTSTRAP 4



Hapsari Laksmi W

NPM: 2015730037

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN  
«tahun»



# **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**

**«tahun»**



# **LEMBAR PENGESAHAN**

**KONVERSI ZURB FOUNDATION KE BOOTSTRAP 4**

**Hapsari Laksmi W**

**NPM: 2015730037**

**Bandung, «tanggal» «bulan» «tahun»**

**Menyetujui,**

**Pembimbing Utama**

**Pembimbing Pendamping**

**«pembimbing utama/1»**

**«pembimbing pendamping/2»**

**Ketua Tim Penguji**

**Anggota Tim Penguji**

**«penguji 1»**

**«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» «bulan» «tahun»

Meterai  
Rp. 6000

Hapsari Laksmi W  
NPM: 2015730037



## **ABSTRAK**

Proses administrasi suatu universitas sebelum menggunakan teknologi memiliki beberapa hambatan. Di Fakultas Teknologi Informasi dan Sains misalnya, mahasiswa kesulitan untuk mengajukan permohonan cetak transkrip apabila berada jauh di daerah Universitas atau dosen tidak bisa merubah jadwal kuliah secara online apabila suatu kelas ditiadakan. Sehingga secara tidak langsung menghambat proses administrasi dalam FTIS selain itu karena harus dilakukan secara manual, kertas akan lebih banyak digunakan untuk proses pencatatan data sehingga tidak ramah lingkungan selain itu proses manual juga memiliki presentase kesalahan yang lebih banyak dibanding dengan menggunakan proses digital.

Sehingga Fakultas Teknologi Informasi dan Sains di Universitas Katolik Parahyangan mengembangkan aplikasi BlueTape yang dikembangkan dari tahun 2015 dengan FrameWork CodeIgniter dan Zurb Foundation sebagai salah satu cara untuk mengurangi pekerjaan paper-based menjadi paperless. Aplikasi ini dapat digunakan untuk melakukan beberapa proses administrasi seperti manajemen cetak transkrip, manajemen jadwal dosen dan manajemen perubahan kuliah.

Skripsi ini ditunjukan untuk migrasi framework front-end sebelumnya yaitu Zurb Foundation menuju ke Bootstrap.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

**Kata-kata kunci:** «paperless, BlueTape, Bootstrap, Foundation, CodeIgniter, mahasiswa, dosen, »



## **ABSTRACT**

«Tuliskan abstrak anda di sini, dalam bahasa Inggris»

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

**Keywords:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Inggris»



*«kepada siapa anda mempersembahkan skripsi ini...?»*



## KATA PENGANTAR

«Tuliskan kata pengantar dari anda di sini ... »

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Bandung, «bulan» «tahun»

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 . . . . .	5
2.2.2 CodeIgniter URLs . . . . .	6
2.2.3 Model . . . . .	6
2.2.4 View . . . . .	7
2.2.5 Controller . . . . .	9
2.3 jQuery . . . . .	10
2.3.1 Penggunaan Dasar jQuery . . . . .	10
2.3.2 jQuery Selectors . . . . .	11
2.3.3 jQuery Event . . . . .	11
2.3.4 jQuery Traversing . . . . .	12
2.3.5 jQuery HTML . . . . .	13
2.4 Foundation 6 . . . . .	14
2.4.1 Struktur File . . . . .	14
2.4.2 Sistem Grid pada Foundation . . . . .	14
2.4.3 Navigation dan Media Attributes . . . . .	15
2.4.4 Komponen . . . . .	17
2.5 Bootstrap 4 . . . . .	21
2.5.1 Sistem Grid Bootstrap . . . . .	21
2.5.2 Konten . . . . .	22

2.5.3	Komponen . . . . .	25
2.6	Plugin . . . . .	32
<b>3</b>	<b>ANALISIS TAMPILAN WEBSITE BLUETAPE</b>	<b>35</b>
3.1	Analisis Konversi Tampilan dengan Foundation . . . . .	35
3.1.1	Menjalankan Framework Foundation . . . . .	35
3.1.2	Folder untuk Menyimpan File Foundation 6 dan Plugin . . . . .	35
3.1.3	Folder untuk Implementasi Framework Foundation 6 dan Plugin . . . . .	35
3.1.4	Import File Foundation . . . . .	36
3.1.5	Grid System Foundation 6 dalam Website BlueTape . . . . .	37
3.1.6	Navigation Bar . . . . .	37
3.1.7	Halaman Login . . . . .	38
3.1.8	Halaman Cetak Transkrip . . . . .	39
3.1.9	Halaman Manajemen Cetak Transkrip . . . . .	41
3.1.10	Halaman Perubahan Kuliah . . . . .	44
3.1.11	Halaman Manajemen Perubahan Kuliah . . . . .	46
3.1.12	Halaman Entri Jadwal Dosen . . . . .	50
3.1.13	Halaman Lihat Jadwal Dosen . . . . .	52
3.2	Analisis Perubahan Kelas yang digunakan dari Foundation 6 ke Bootstrap 4 pada BlueTape . . . . .	52
<b>4</b>	<b>IMPLEMENTASI WEBSITE BLUETAPE DENGAN BOOTSTRAP 4</b>	<b>57</b>
4.1	Konversi Tampilan dengan Bootstrap 4 . . . . .	57
4.1.1	Menjalankan Framework Bootstrap 4 . . . . .	57
4.1.2	Folder untuk Menyimpan File Bootstrap 4 dan Plugin . . . . .	57
4.1.3	Folder untuk Implementasi Framework Bootstrap 4 dan Plugin . . . . .	57
4.1.4	Import File Bootstrap 4 . . . . .	58
4.1.5	Grid System Bootstrap 4 dalam Website BlueTape . . . . .	58
4.1.6	Navigation Bar . . . . .	59
4.1.7	Halaman Login . . . . .	61
4.1.8	Halaman Cetak Transkrip . . . . .	62
4.1.9	Halaman Manajemen Cetak Transkrip . . . . .	64
4.1.10	Halaman Perubahan Kuliah . . . . .	68
4.1.11	Halaman Manajemen Perubahan Kuliah . . . . .	70
4.1.12	Halaman Entri Jadwal Dosen . . . . .	74
4.1.13	Halaman Lihat Jadwal Dosen . . . . .	75
<b>5</b>	<b>IMPLEMENTASI BOOTSTRAP 6 KE DALAM WEBSITE BLUETAPE</b>	<b>77</b>
5.1	Implementasi Tampilan Login . . . . .	78
<b>DAFTAR REFERENSI</b>		<b>79</b>
<b>A KODE PROGRAM</b>		<b>81</b>
<b>B HASIL EKSPERIMEN</b>		<b>83</b>

## DAFTAR GAMBAR

2.1	Flow Chart Aplikasi CodeIgniter . . . . .	6
2.2	Struktur File Zurb Foundation . . . . .	14
2.3	Grid pada Zurb Foundation . . . . .	15
2.4	<i>Basic Navigation Menu</i> pada Foundation . . . . .	16
2.5	Menu <i>align to right</i> dalam Foundation . . . . .	16
2.6	Menu <i>align to center</i> dalam Foundation . . . . .	16
2.7	Menu <i>active state</i> dalam Foundation . . . . .	16
2.8	<i>Menu text</i> dalam Foundation . . . . .	17
2.9	Basic Button pada Foundation . . . . .	17
2.10	Coloring Button pada Foundation . . . . .	18
2.11	Basic Table pada Foundation . . . . .	18
2.12	Text Input pada Foundation . . . . .	19
2.13	Grid pada Zurb Foundation . . . . .	20
2.14	Grid pada Bootstrap . . . . .	21
2.15	Pilihan kelas grid pada Bootstrap . . . . .	22
2.16	Tabel default pada Bootstrap . . . . .	22
2.17	Tabel default pada Bootstrap . . . . .	23
2.18	Tabel default pada Bootstrap . . . . .	24
2.19	Menyelaraskan gambar ke kanan dan kiri pada bootstrap . . . . .	25
2.20	<i>Forms Basic</i> pada Bootstrap . . . . .	26
2.21	<i>Forms Basic</i> pada Bootstrap . . . . .	26
2.22	<i>Disabled Basic</i> pada Bootstrap . . . . .	26
2.23	<i>Button</i> pada Bootstrap . . . . .	27
2.24	Tombol <i>dropdown</i> pada Bootstrap . . . . .	28
2.25	Badge pada Bootstrap . . . . .	28
2.26	Card pada Bootstrap . . . . .	29
2.27	Navigation Bar pada Bootstrap . . . . .	29
2.28	Modal pada Bootstrap . . . . .	31
2.29	Ikon <i>Coffee</i> pada Font Awesome . . . . .	31
2.30	Alert pada Bootstrap . . . . .	32
2.31	Datetimepicker pada Bootstrap . . . . .	33
3.1	Analisis Tampilan Login . . . . .	37
3.2	Analisis Tampilan Login pada layar Medium dan Small . . . . .	38
3.3	Analisis Tampilan Login . . . . .	38
3.4	Analisis Halaman Cetak Transkrip . . . . .	39
3.5	Analisis Modal Lihat . . . . .	40
3.6	Analisis Manajemen Cetak Transkrip . . . . .	41
3.7	Analisis Modal Lihat . . . . .	42
3.8	Analisis Modal Print . . . . .	42
3.9	Analisis Modal Tolak . . . . .	43
3.10	Analisis Modal Hapus . . . . .	43

3.11	Analisis Tampilan Perubahan Kuliah . . . . .	44
3.12	Analisis Modal Lihat . . . . .	45
3.13	Analisis Halaman Manajemen Perubahan Kuliah . . . . .	46
3.14	Analisis Modal Lihat . . . . .	47
3.15	Analisis Modal Setuju . . . . .	48
3.16	Analisis Modal Tolak . . . . .	48
3.17	Analisis Modal Hapus . . . . .	49
3.18	Analisis Halaman Entri Jadwal Dosen . . . . .	50
3.19	Analisis Modal Edit Jadwal . . . . .	51
3.20	Analisis Modal Edit Jadwal . . . . .	52
4.1	Konversi Tampilan Login . . . . .	59
4.2	Analisis Tampilan Login pada layar Medium dan Small . . . . .	60
4.3	Konversi Tampilan Login . . . . .	61
4.4	Konversi Halaman Cetak Transkrip . . . . .	62
4.5	Konversi Modal Lihat . . . . .	63
4.6	Konversi Manajemen Cetak Transkrip . . . . .	64
4.7	Konversi Modal Lihat . . . . .	65
4.8	Konversi Modal Print . . . . .	65
4.9	Konversi Modal Tolak . . . . .	66
4.10	Konversi Modal Hapus . . . . .	67
4.11	Konversi Perubahan Kuliah . . . . .	68
4.12	Konversi Modal Lihat . . . . .	69
4.13	Konversi Halaman Manajemen Perubahan Kuliah . . . . .	70
4.14	Konversi Modal Lihat . . . . .	71
4.15	Analisis Modal Setuju . . . . .	72
4.16	Konversi Modal Tolak . . . . .	72
4.17	Konversi Modal Hapus . . . . .	73
4.18	Konversi Halaman Entri Jadwal Dosen . . . . .	74
4.19	Analisis Modal Edit Jadwal . . . . .	75
5.1	Konversi Tampilan Login Bootstrap . . . . .	78
B.1	Hasil 1 . . . . .	83
B.2	Hasil 2 . . . . .	83
B.3	Hasil 3 . . . . .	83
B.4	Hasil 4 . . . . .	83

## **DAFTAR TABEL**



# 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** dan **Foundation**.

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* (bila diperlukan).

### 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 **Zurb Foundation** ke **Bootstrap 4**
2. Bagaimana mengimplementasikan *plugin* yang tersedia di dalam *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. Mengimplementasikan *plugin* yang tersedia dalam *library* *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 *Zurb Foundation* 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* Zurb Foundation 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* Zurb Foundation, *framework* Bootstrap 4.

Bab Analisis

Bab 3 berisi analisis antarmuka yang sudah ada.

Bab Perancangan antarmuka

Bab 4 berisi mengenai isi program dan perancangan kelas - kelas program.

Bab Implementasi

Bab 5 membahas mengenai pembuatan template utama aplikasi BlueTape yaitu , pembuatan menu aplikasi dan hasil eksekusi tampilan aplikasi.

Bab Kesimpulan dan saran.

Bab 6 berisi kesimpulan setelah mengerjakan skripsi ini dan saran yang diberikan.

## 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 <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

#### 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 cetak transkrip. Terdapat sebuah tabel yang menampilkan daftar pemohonan dengan tanggal yang terurut. Staf Tata Usaha dapat mencari daftar permintaan berdasarkan NPM pemohon.

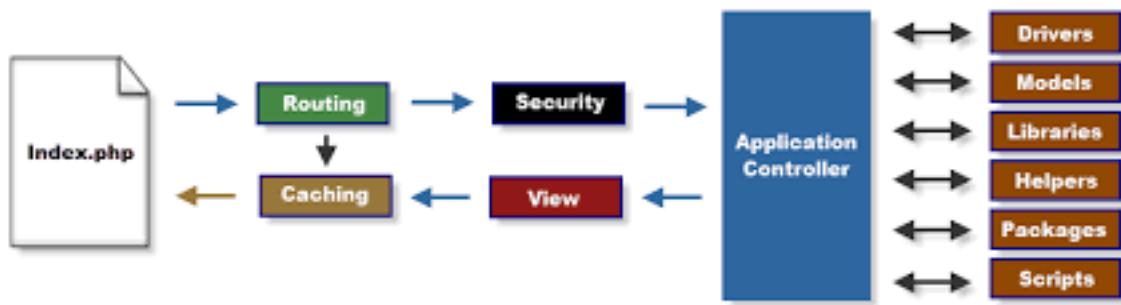
Beberapa tombol yang tersedia untuk setiap permohonan :

- 🔍 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

### 2.2.1 Application Flow Chart

Gambar berikut mengilustrasikan bagaimana alur data pada sistem :



Gambar 2.1: Flow Chart Aplikasi 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

CodeIgniter menggunakan pendekatan berbasis segmen :

example .com / class / function / ID

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.

#### Anatomi Model

Kelas model akan disimpan dalam direktori `application/models/directory`. Prototipe dasar dari sebuah model kelas :

```
<?php
class Model_name extends CI_Model {
}
```

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

### Loading a Model

Model akan dimuat dan dipanggil didalam metode *controller*. Untuk memuat sebuah model maka dapat digunakan metode berikut:

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

### Koneksi ke Database

Apabila model sudah dimuat, model tersebut tidak terhubung secara langsung ke database. Dengan cara secara manual mengatur koneksi database melalui parameter ketiga:

```
$config[ 'hostname' ] = 'localhost';
$config[ 'username' ] = 'myusername';
$config[ 'password' ] = 'mypassword';
$config[ 'database' ] = 'mydatabase';
$config[ 'dbdriver' ] = 'mysqli';
$config[ 'dbprefix' ] = '';
$config[ 'pconnect' ] = FALSE;
$config[ 'db_debug' ] = TRUE;

$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.

#### Membuat sebuah View

CodeIgniter memuat view dengan memanggil sebuah file php, misalkan `blogview.php`, dan *developer* dapat mengisinya dengan kode HTML sebagai berikut:

```
<html>
<head>
    <title>My Blog</title>
</head>
<body>
    <h1>Welcome to my Blog !</h1>
</body>
</html>
```

File tersebut akan disimpan di direktori `application/views/`.

## Loading sebuah View

View dapat dimuat dengan membuat file *view* dengan syntax berikut:

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

Dimana **name** adalah nama dari file view.

## Memuat Beberapa View

CodeIgniter dapat menangani beberapa panggilan dari dalam *controller* dengan menggunakan syntax :

```
$this->load->view()
```

Apabila ada lebih dari satu panggilan yang terjadi, maka *views* akan dilampirkan secara bersamaan. Berikut ini kode yang digunakan jika *developer* ingin mempunyai sebuah halaman yang terdiri dari **header**, **menu**, **content** dan **footer**.

```
<?php

class Page extends CI_Controller {

    public function index()
    {
        $data[ 'page_title' ] = 'Your title';
        $this->load->view( 'header' );
        $this->load->view( 'menu' );
        $this->load->view( 'content' , $data );
        $this->load->view( 'footer' );
    }

}
```

## Menyimpan Views didalam Sub Direktori

Untuk menyimpan didalam sub direktori maka dapat menyertakan nama direktori yang memuat *view*.

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

## Menambahkan data dinamis ke View

Data yang dikirim dari controller menuju view berbentuk array atau objek, sehingga akan dilampirkan dalam parameter kedua dalam metode loading view. Berikut ini penggunaan dengan array:

```
$data = array(
    'title' => 'My Title',
    'heading' => 'My Heading',
    'message' => 'My Message'
);

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

Kemudian, penggunaan dengan objek:

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

Sehingga apabila dimasukan ke controller, kode yang ditambahkan adalah:

```
<?php
class Blog extends CI_Controller {

    public function index()
    {
        $data['title'] = "My Real Title";
        $data['heading'] = "My Real Heading";

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

Untuk mengaksesnya dalam file HTML maka *developer* dapat menggunakan syntax php :

```
<html>
<head>
    <title><?php echo $title;?></title>
</head>
<body>
    <h1><?php echo $heading;?></h1>
</body>
</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. Sebuah *controller* secara sederhana merupakan sebuah file yang dinamakan dengan aturan tertentu sehingga dapat dihubungkan dengan sebuah URL. Misalnya untuk URL ini:

```
<?php
example.com/index.php/blog/
```

Dalam contoh diatas, *CodeIgniter* berusaha menemukan *controller* bernama Blog.php dan lalu memuatnya. Ketika sebuah nama *controller* sesuai dengan segmen pertama dari sebuah URL, maka URL akan memuatnya. Kode berikut merupakan contoh dari *controller* sederhana.

```
<?php
class Blog extends CI_Controller {

    public function index()
    {
        echo 'Hello World';
    }
}
```

## Method

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

```
<?php
class Blog extends CI_Controller {

    public function index()
    {
        echo 'HelloWorld!';
    }

    public function comments()
    {
        echo 'Lookatthis!';
    }
}
```

Pemanggilan method `index` dapat secara otomatis dilakukan apabila segmen kedua kosong. Namun ada cara lain untuk menampilkan pesan "Hello World" yang dapat dilakukan dengan:

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

Kemudian untuk memuat method `comment()` dapat dituliskan sebagai berikut:

```
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 *user* panggil hanya dengan satu baris kode.

### 2.3.1 Penggunaan Dasar jQuery

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

Syntax dasar dalam jQuery sebagai berikut :

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

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

Contoh penggunaan syntax dalam jQuery sebagai berikut:

- `$(this).hide()`: menyembunyikan elemen saat ini.
- `$("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: `$( )`. Contoh : Perintah user untuk memilih sebuah tombol sehingga semua elemen `<p>` akan tersembunyi.

```
$( document ).ready( function () {
    $("button").click( function () {
        $("p").hide();
    });
});
```

### 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 event` :

```
$( document ).ready( function () {
    // jQuery methods go here ...
});
```

Syntax ini 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. Contoh : Memilih elemen `<p>` untuk memberikan teks berupa alert

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

**`change()`**

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

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

Contoh penggunaan metode ini: Memberikan teks peringatan ketika sebuah field <input> berubah.

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

### **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 syntax yang dipakai :

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

Contoh penggunaan metode : Memberikan sebuah peringatan ketika sebuah teks dipilih didalam sebuah *text field*.

```
$( "input" ).select(function(){
    alert( "Text marked!" );
});
```

### **preventDefault()**

Metode preventDefault() membatalkan event yang *user* kehendaki. 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.

```
document.getElementById( "myAnchor" ).addEventListener( "click" ,
function(event){
    event.preventDefault()
});
```

### **2.3.4 jQuery Traversing**

Berfungsi untuk menemukan elemen HTML berdasarkan relasinya dengan elemen lain. Metode yang dipakai untuk fungsi ini adalah:

#### **closest()**

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

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

Fungsi diatas akan mengembalikan *ancestor* dari <span>, yaitu sebuah elemen <ul>. Hasil :

**find()**

Metode **find()** mengembalikan elemen turunan/ancestor dari elemen yang dipilih oleh *user*. Contoh: Mengembalikan semua elemen **<span>** yang merupakan turunan dari elemen **<ul>**.

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

Hasil dari kode diatas:

**2.3.5 jQuery HTML****remove()**

**remove()** - berfungsi untuk menghapus elemen HTML yang ada. Contoh penggunaannya seperti: Menghapus elemen dengan **id="div1"**.

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

**\$removeAttr()**

Metode **removeAttr()** menghapus satu atau banyak atribut dari elemen terpilih. Syntax dari metode ini :

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

Contoh pemakaian metode ini: Menghapus atribut style dari semua elemen **<p>**:

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

**\$clone()**

Metode **clone()** membuat salinan dari elemen terpilih, termasuk child nodes, teks atau atribut. Syntax dari metode ini :

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

Contoh pemakaian metode ini: Clone semua elemen **<p>** dan masukan semua elemen pada akhir elemen **<body>**.

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

**insertBefore**

Metode **insertBefore()** akan memasukan elemen HTML sebelum elemen terpilih. Syntax dari metode ini :

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

Dimana :

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

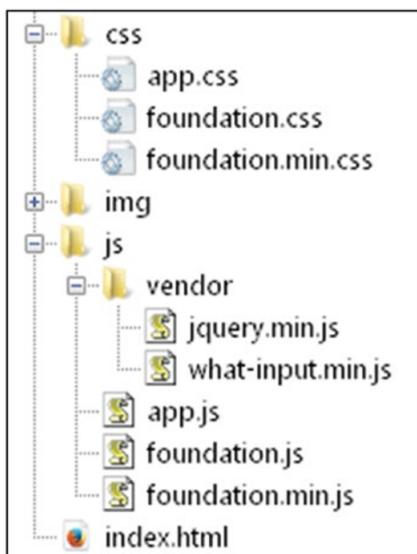
- **selector** : Required. Menjelaskan dimana konten akan dimasukan.

Contoh pemakaian metode ini: Memasukkan sebuah elemen `<span>` sebelum `<p>`.

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

## 2.4 Foundation 6

### 2.4.1 Struktur File



Gambar 2.2: Struktur File Zurb Foundation

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.

### 2.4.2 Sistem Grid pada Foundation

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 ditentunkan kelasnya dengan tiga pilihan `.small-#`, `.medium-#` dan `.large-#`.

```
<div class="row">
  <div class="columns small-2 large-4"><!-- ... --></div>
  <div class="columns small-4 large-4"><!-- ... --></div>
  <div class="columns small-6 large-4"><!-- ... --></div>
</div>
```

```
<div class="row">
  <div class="columns large-3">!-- ... -->/div>
  <div class="columns large-6">!-- ... -->/div>
  <div class="columns large-3">!-- ... -->/div>
</div>
<div class="row">
  <div class="columns small-6 large-2">!-- ... -->/div>
  <div class="columns small-6 large-8">!-- ... -->/div>
  <div class="columns small-12 large-2">!-- ... -->/div>
</div>
<div class="row">
  <div class="columns small-3">!-- ... -->/div>
  <div class="columns small-9">!-- ... -->/div>
</div>
<div class="row">
  <div class="columns large-4">!-- ... -->/div>
  <div class="columns large-8">!-- ... -->/div>
</div>
<div class="row">
  <div class="columns small-6 large-5">!-- ... -->/div>
  <div class="columns small-6 large-7">!-- ... -->/div>
</div>
<div class="row">
  <div class="columns large-6">!-- ... -->/div>
  <div class="columns large-6">!-- ... -->/div>
</div>
```



Gambar 2.3: Grid pada Zurb Foundation

### 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:

```
<ul class="menu">
  <li><a href="#">One</a></li>
  <li><a href="#">Two</a></li>
```

```
<li><a href="#">Three</a></li>
<li><a href="#">Four</a></li>
</ul>
```

One    Two    Three    Four

Gambar 2.4: *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` atau kearah tengah dengan menambahkan kelas `.align-center` didalam kelas `.menu`.

```
<ul class="menu align-right">
  <li><a href="#">One</a></li>
  <li><a href="#">Two</a></li>
  <li><a href="#">Three</a></li>
  <li><a href="#">Four</a></li>
</ul>
```

One    Two    Three    Four

Gambar 2.5: Menu *align to right* dalam Foundation

```
<ul class="menu align-center">
  <li><a href="#">One</a></li>
  <li><a href="#">Two</a></li>
  <li><a href="#">Three</a></li>
  <li><a href="#">Four</a></li>
</ul>
```

One    Two    Three    Four

Gambar 2.6: Menu *align to center* dalam Foundation

### Active State

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

```
<ul class="menu">
  <li class="is-active"><a>Home</a></li>
  <li><a>About</a></li>
  <li><a>Nachos</a></li>
</ul>
```

Home    About    Nachos

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.

```
<ul class="menu">
<li class="menu-text">Site Title</li>
<li><a href="#">One</a></li>
<li><a href="#">Two</a></li>
<li><a href="#">Three</a></li>
</ul>
```

Site Title    One    Two    Three

Gambar 2.8: *Menu text* dalam Foundation

### 2.4.4 Komponen

#### 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.

```
<!-- Anchors (links) -->
<a href="about.html" class="button">Learn More</a>
<a href="#features" class="button">View All Features</a>

<!-- Buttons (actions) -->
<button type="button" class="success-button">Save</button>
<button type="button" class="alert-button">Delete</button>
```

Learn More    View All Features    Save    Delete

Gambar 2.9: Basic Button pada Foundation

Warna pada button dapat diterapkan untuk memperlihatkan fungsi yang sesuai dengan aksi yang digunakan.

```
<a class="button-primary" href="#">Primary</a>
<a class="button-secondary" href="#">Secondary</a>
<a class="button-success" href="#">Success</a>
<a class="button-alert" href="#">Alert</a>
<a class="button-warning" href="#">Warning</a>
```



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*.

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

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

## Hover State

Hover State diaplikasikan menggunakan kelas `.hover` untuk sedikit membedakan baris terpilih dalam tabel dengan baris-baris lainnya dengan cara menggelapkan baris terpilih.

```
<table class="hover">
</table>
```

## Striped

Secara default, tabel akan memiliki baris yang bergaris. Ada beberapa pilihan kelas untuk mengubah desain tabelnya.

- Kelas `.unstriped` dapat digunakan untuk menghapus garis-garis atau dengan mengubah `$table-is-striped` ke `false` untuk menghapus semua strip pada seluruh tabel.
- Kelas `.striped` untuk menambahkan strip pada tabel.

## 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`.

```
<form>
<div class="grid-container">
  <div class="grid-x grid-padding-x">
    <div class="medium-6 cell">
      <label>Input Label
        <input type="text" placeholder=".medium-6.cell">
      </label>
    </div>
    <div class="medium-6 cell">
      <label>Input Label
        <input type="text" placeholder=".medium-6.cell">
      </label>
    </div>
  </div>
</div>
</form>
```

Input Label

.medium-6.cell

Input Label

.medium-6.cell

Gambar 2.12: Text Input pada Foundation

**Select Menus** Penggunaan `select` menus digunakan apabila *developer* menginginkan banyak pilihan data dalam satu menu.

```
<label>Select Menu
<select>
  <option value="husker">Husker</option>
```

```

<option value="starbuck">Starbuck</option>
<option value="hotdog">Hot Dog</option>
<option value="apollo">Apollo</option>
</select>
</label>

```

## Tabs

Tab semakin banyak digunakan dalam desain web karena *developer* dapat menyajikan konten secara seragam. Ini memungkinkan *developer* untuk menyimpan banyak dokumen dalam satu *window*. *developer* dapat menggunakan tab sebagai widget navigasi untuk beralih antar konten sehingga menghasilkan tata letak yang sistematis dan bersih. Komponen Tab dari Foundation membantu *developer* melakukan hal itu hanya dengan menambahkan beberapa baris kode.

```

<ul class="tabs" data-tabs id="tab_component">
  <li class="tabs-title"><a href="#pub1">Section 1</a></li>
  <li class="tabs-title is-active"><a href="#pub2">Section 2</a></li>
  <li class="tabs-title"><a href="#pub3">Section 3</a></li>
  <li class="tabs-title"><a href="#pub4">Section 4</a></li>
</ul>
<div class="tabs-content" data-tabs-content="tab_component">
  <div class="tabs-panel" id="pub1">
    <p>Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts.</p>
  </div>
  <div class="tabs-panel is-active" id="pub2">
    <p> Separated they live in Bookmarksgrove right at the coast of the Semantics , a large language ocean . </p>
  </div>
  <div class="tabs-panel" id="pub3">
    <p>A small river named Duden flows by their place and supplies it with the necessary regelialia .</p>
  </div>
  <div class="tabs-panel" id="pub4">
    <p>It is a paradisematic country , in which roasted parts of sentences fly into your mouth. </p>
  </div>
</div>

```



Gambar 2.13: Grid pada Zurb Foundation

## Dropdown Menu

Berfungsi untuk mengubah menu dasar menjadi menu dropdown yang dapat di-expand dengan plugin Menu Dropdown. Menu dropdown dibangun berdasarkan sintaks komponen Menu. Tambahkan kelas `.dropdown` dan atribut `data-dropdown-menu` ke wadah menu untuk mengatur dropdown.

```

<ul class="dropdown-menu" data-dropdown-menu>
  <li><a href="#">Item 1</a></li>
  <li><a href="#">Item 2</a></li>
  <li><a href="#">Item 3</a></li>
  <li><a href="#">Item 4</a></li>

```

```
</ul>
```

## 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*.

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

## 2.5 Bootstrap 4

### 2.5.1 Sistem Grid Bootstrap

Sistem grid Bootstrap menggunakan *container*, *rows*, dan *columns* untuk tata letak dan penyelarasan konten. Selain itu sistem ini dibangun dengan *flexbox* dan seluruhnya *responsive*. [2]



Gambar 2.14: Grid pada Bootstrap

```
<div class="container">
  <div class="row">
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
  </div>
</div>
```

Dalam contoh diatas 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.

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 `.com-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 yang menjelaskan penggunaan kelas grid dalam berbagai perangkat :

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

Gambar 2.15: Pilihan kelas grid pada Bootstrap

### 2.5.2 Konten

#### Tabel

**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.

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

Gambar 2.16: Tabel default pada Bootstrap

```
<table class="table">
<thead>
<tr>
  <th scope="col" style="text-align: center;>#</th>
  <th scope="col" style="text-align: center;>First</th>
  <th scope="col" style="text-align: center;>Last</th>
  <th scope="col" style="text-align: center;>Handle</th>
</tr>
</thead>
<tbody>
<tr>
  <th scope="row" style="text-align: center;>1</th>
  <td>Mark</td>
  <td>Otto</td>
```

```

<td>@mdo</td>
</tr>
<tr>
  <th scope="row">2</th>
  <td>Jacob</td>
  <td>Thornton</td>
  <td>@fat</td>
</tr>
<tr>
  <th scope="row">3</th>
  <td>Larry</td>
  <td>the Bird</td>
  <td>@twitter</td>
</tr>
</tbody>
</table>

```

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

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

Gambar 2.17: Tabel default pada Bootstrap

```


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


```

```

        <td>@twitter</td>
    </tr>
</tbody>
</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>`.

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

Gambar 2.18: Tabel default pada Bootstrap

```


| scope="col"># | First          | Last     | Handle   |
|---------------|----------------|----------|----------|
| row 1         | Mark           | Otto     | @mdo     |
| row 2         | Jacob          | Thornton | @fat     |
| row 3         | Larry the Bird |          | @twitter |


```

## 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.

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



Gambar 2.19: Menyelaraskan gambar ke kanan dan kiri pada bootstrap

```
>
>
```

### 2.5.3 Komponen

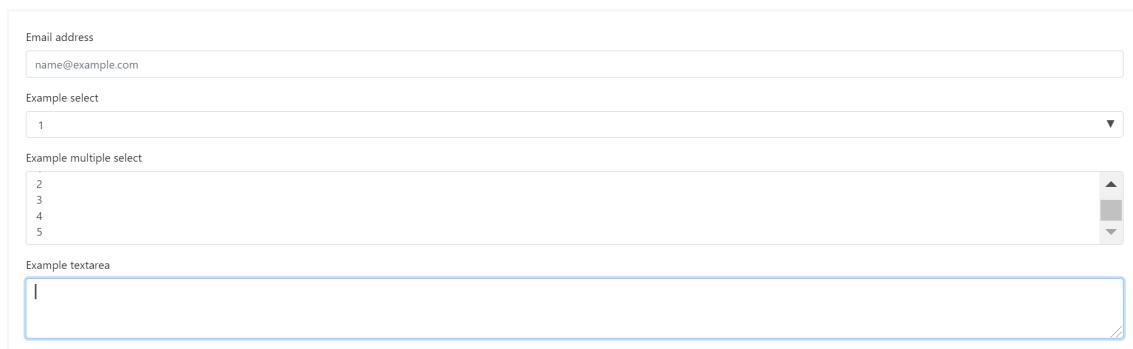
#### Formulir

*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

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

```
<form>
  <div class="form-group">
    <label for="exampleFormControlInput1">Email address</label>
    <input type="email" class="form-control" id="exampleFormControlInput1" placeholder="Email address" required="required">
  </div>
  <div class="form-group">
    <label for="exampleFormControlSelect1">Example select</label>
    <select class="form-control" id="exampleFormControlSelect1">
      <option>1</option>
      <option>2</option>
      <option>3</option>
      <option>4</option>
      <option>5</option>
    </select>
  </div>
  <div class="form-group">
    <label for="exampleFormControlSelect2">Example multiple select</label>
    <select multiple class="form-control" id="exampleFormControlSelect2">
      <option>1</option>
      <option>2</option>
      <option>3</option>
      <option>4</option>
      <option>5</option>
    </select>
  </div>
  <div class="form-group">
    <label for="exampleFormControlTextarea1">Example textarea</label>
    <textarea class="form-control" id="exampleFormControlTextarea1" rows="3"></textarea>
  </div>
</form>
```

Gambar 2.20: *Forms Basic* pada Bootstrap

### Column Sizing

Bootstrap memungkinkan *developer* 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.

Gambar 2.21: *Forms Basic* pada Bootstrap

```
<form>
  <div class="form-row">
    <div class="col-7">
      <input type="text" class="form-control" placeholder="City">
    </div>
    <div class="col">
      <input type="text" class="form-control" placeholder="State">
    </div>
    <div class="col">
      <input type="text" class="form-control" placeholder="Zip">
    </div>
  </div>
</form>
```

### 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>`.

Gambar 2.22: *Disabled Basic* pada Bootstrap

```
<form>
  <fieldset disabled>
    <div class="form-group">
      <label for="disabledTextInput">Disabled input</label>
      <input type="text" id="disabledTextInput" class="form-control" placeholder="Disabled input">
    </div>
    <div class="form-group">
      <label for="disabledSelect">Disabled select menu</label>
      <select id="disabledSelect" class="form-control">
        <option>Disabled select</option>
      </select>
    </div>
    <div class="form-check">
      <input class="form-check-input" type="checkbox" id="disabledFieldsetCheck" disabled>
      <label class="form-check-label" for="disabledFieldsetCheck">
        Can't check this
      </label>
    </div>
    <button type="submit" class="btn btn-primary">Submit</button>
  </fieldset>
</form>
```

## Button

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



Gambar 2.23: *Button* pada Bootstrap

```
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>

<button type="button" class="btn btn-link">Link</button>
```

## Button with Dropdowns

```
<div class="input-group mb-3">
  <div class="input-group-prepend">
    <button class="btn btn-outline-secondary dropdown-toggle" type="button" data-toggle="button"
           aria-haspopup="true" aria-expanded="false">Dropdown</button>
  <div class="dropdown-menu">
    <a class="dropdown-item" href="#">Action</a>
    <a class="dropdown-item" href="#">Another action</a>
    <a class="dropdown-item" href="#">Something else here</a>
```

```

<div role="separator" class="dropdown-divider"></div>
<a class="dropdown-item" href="#">Separated link</a>
</div>
</div>
<input type="text" class="form-control" aria-label="Text input with dropdown button">
</div>

<div class="input-group">
<input type="text" class="form-control" aria-label="Text input with dropdown button">
<div class="input-group-append">
<button class="btn btn-outline-secondary dropdown-toggle" type="button" data-toggle="dropdown"
aria-haspopup="true" aria-expanded="false">Dropdown</button>
<div class="dropdown-menu">
<a class="dropdown-item" href="#">Action</a>
<a class="dropdown-item" href="#">Another action</a>
<a class="dropdown-item" href="#">Something else here</a>
<div role="separator" class="dropdown-divider"></div>
<a class="dropdown-item" href="#">Separated link</a>
</div>
</div>
</div>

```



Gambar 2.24: Tombol *dropdown* pada Bootstrap

## Badge

Kelas badge dan .badge-\* di dalam sebuah <a> akan memberikan badge yang dapat diberi atribut *hover* dan *focus*.

```

<a href="#" class="badge badge-primary">Primary</a>
<a href="#" class="badge badge-secondary">Secondary</a>
<a href="#" class="badge badge-success">Success</a>
<a href="#" class="badge badge-danger">Danger</a>
<a href="#" class="badge badge-warning">Warning</a>
<a href="#" class="badge badge-info">Info</a>
<a href="#" class="badge badge-light">Light</a>
<a href="#" class="badge badge-dark">Dark</a>

```

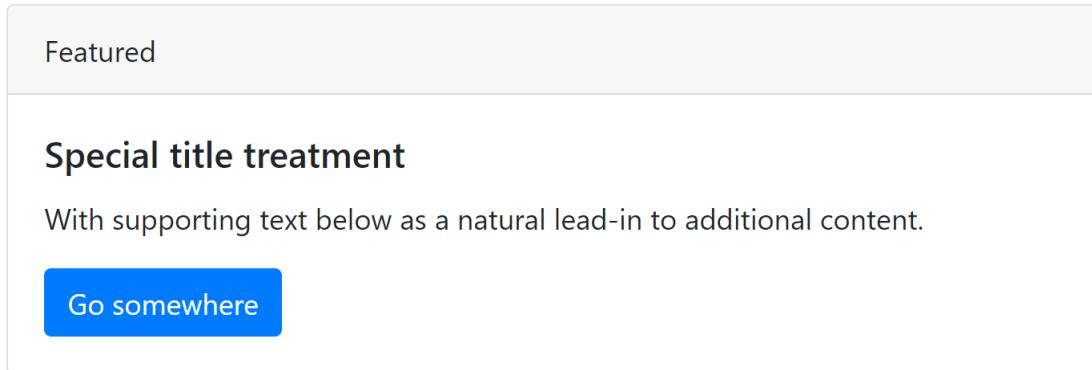


Gambar 2.25: Badge pada Bootstrap

## Card

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

```
<div class="card">
    <div class="card-header">
        Featured
    </div>
    <div class="card-body">
        <h5 class="card-title">Special title treatment</h5>
        <p class="card-text">With supporting text below as
        a natural lead-in to additional content.</p>
        <a href="#" class="btn btn-primary" card>Go somewhere</a>
    </div>
</div>
```



Gambar 2.26: 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 merupakan semua sub-komponen yang termasuk dalam navigation bar, navbar mengimplementasikan tema `light-themed` yang secara otomatis menyembunyikan menu pada breakpoint `lg`

Navbar Home Link Dropdown ▾ Disabled

Search

Gambar 2.27: Navigation Bar pada Bootstrap

```

<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="#">Navbar</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
      <li class="nav-item active">
        <a class="nav-link" href="#">Home <span class="sr-only">( current )</span></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Link</a>
      </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
          Dropdown
        </a>
        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
          <a class="dropdown-item" href="#">Action</a>
          <a class="dropdown-item" href="#">Another action</a>
          <div class="dropdown-divider"></div>
          <a class="dropdown-item" href="#">Something else here</a>
        </div>
      </li>
      <li class="nav-item">
        <a class="nav-link disabled" href="#">Disabled</a>
      </li>
    </ul>
    <form class="form-inline my-2 my-lg-0">
      <input class="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search" aria-describedby="SearchFormInput">
      <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>
    </form>
  </div>
</nav>

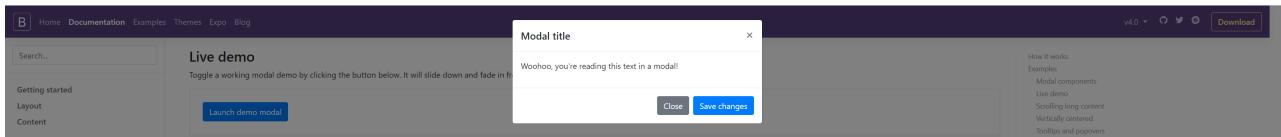
```

## 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).



Gambar 2.28: Modal pada Bootstrap

Listing 2.1: contoh aja

```
<!-- Button trigger modal -->
<button type="button" class="btn btn-primary" data-toggle="modal" data-target="#myModal">
  Launch demo modal
</button>

<!-- Modal -->
<div class="modal-fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        ...
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>
```

## Ikon

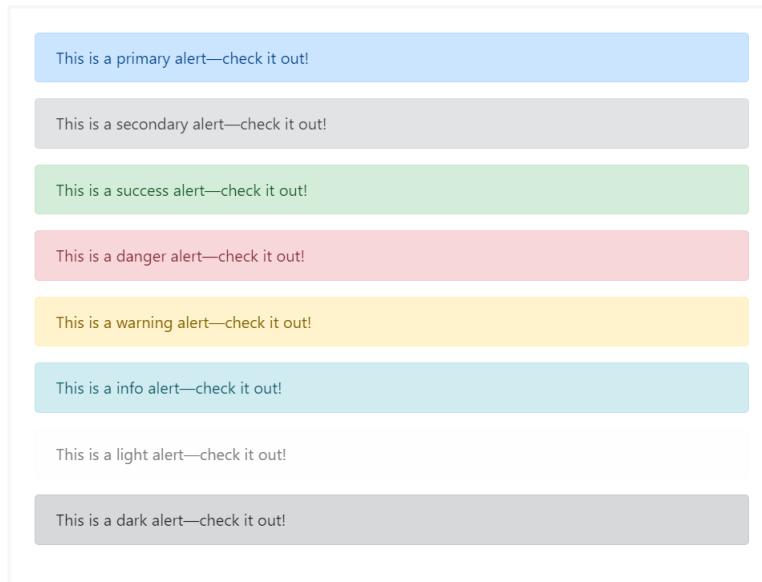
Bootstrap tidak memiliki *library* ikon secara *default*, sehingga ikon yang digunakan diambil dari **Font Awesome**. Penggunaan ikon dengan menggunakan tag `<i>` yang disertai dengan kelas `fa` (font-awesome).

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

Gambar 2.29: 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.



Gambar 2.30: Alert pada Bootstrap

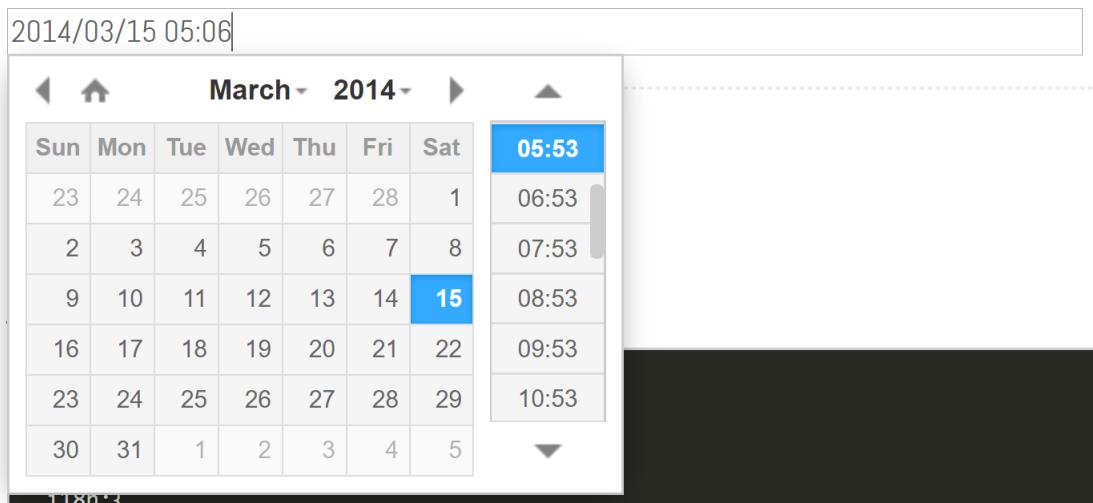
```
<div class="alert alert-primary" role="alert">
  This is a primary alert check it out!
</div>
<div class="alert alert-secondary" role="alert">
  This is a secondary alert check it out!
</div>
<div class="alert alert-success" role="alert">
  This is a success alert check it out!
</div>
<div class="alert alert-danger" role="alert">
  This is a danger alert check it out!
</div>
<div class="alert alert-warning" role="alert">
  This is a warning alert check it out!
</div>
<div class="alert alert-info" role="alert">
  This is a info alert check it out!
</div>
<div class="alert alert-light" role="alert">
  This is a light alert check it out!
</div>
<div class="alert alert-dark" role="alert">
  This is a dark alert check it out!
</div>
```

## 2.6 Plugin

**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.



Gambar 2.31: Datetimepicker pada Bootstrap

Penggunaan nya dalam kode HTML sebagai berikut :

```
<input id="datetimepicker" type="text" >
```

Penggunaan dalam kode Javascript sebagai berikut :

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



## BAB 3

### ANALISIS TAMPILAN WEBSITE BLUETAPE

Bab 3 menjelaskan komponen yang digunakan untuk membangun seluruh tampilan website dengan framework Foundation 6 dan kelas dalam Bootstrap 4 yang akan digunakan untuk website BlueTape.

#### **3.1 Analisis Konversi Tampilan dengan Foundation**

Pada bagian ini akan dijabarkan keseluruhan kelas yang dipakai dalam website BlueTape. Pertama akan dijabarkan file framework yang digunakan, kemudian akan dijelaskan komponen yang digunakan dalam website BluTape dan terakhir penggunaan kelas framework Foundation pada website.

##### **3.1.1 Menjalankan Framework Foundation**

Website BlueTape menggunakan sebuah framework PHP yaitu CodeIgniter. Hal pertama yang dilakukan saat ingin menjalankan tampilan adalah cek terlebih dimana file yang berkaitan dengan Framework Foundation 6 disimpan, lalu dilihat file tersebut dijalankan ketika membangun sebuah tampilan.

##### **3.1.2 Folder untuk Menyimpan File Foundation 6 dan Plugin**

Seluruh file terkait dengan Framework Foundation 6 diletakkan dalam folder `/www/public`. Dimana terdiri dari folder:

1. css: Terdiri dari file - file css dari beserta file ikon bawaan dari framework Foundation 6.
2. img: Menyimpan sebuah gambar logo BlueTape.
3. js: Menyimpan file javascript bawaan dari Foundation 6 dan file jquery yang tersimpan dalam folder vendor.
4. lib: Menyimpan plugin xdan-datetimepicker.

##### **3.1.3 Folder untuk Implementasi Framework Foundation 6 dan Plugin**

Untuk melihat code tampilan website, maka *user* akan mengakses file - file dalam folder `/www/application/views`. Folder views terdiri dari beberapa sub-folder. Secara garis besar, file terdiri dari dua fungsi yaitu menyimpan kode template yang digunakan untuk keseluruhan tampilan dan implementasi komponen - komponen Foundation 6 untuk membangun sebuah website.

Pertama untuk sub-folder template terdiri dari:

1. auth : Berisi file terkait dengan framework Foundation akan diimport disini, kemudian akan disertakan sebuah tampilan login dan notifikasi - notifikasi terkait dengan status login. File tersebut tercatat di `login.php`.

2. error : Berisi folder untuk mengatasi error ketika user gagal untuk *load* tampilan. Pesan akan ditampilkan pada *user* ketika error terjadi. Folder ini sudah ada sejak pertama mengunduh framework CodeIgniter.
3. templates : Folder ini mencangkup kode template yang digunakan untuk seluruh bagian website BlueTape. Terdiri dari:
  - flashmessage Menyimpan notifikasi terkait dengan autentikasi ketika login ke dalam website.
  - head\_loggedin Another descriptive item
  - script.foundation And another descriptive item
  - head\_loggedin And another descriptive item

Lalu sub-folder template yang berfungsi untuk menyimpan kode tampilan website terdiri 6 folder. Masing - masing folder memiliki sebuah file **main.php**. Sub-folder tersebut yaitu:

1. EntriJadwalDosen
2. LihatJadwalDisen
3. PerubahanKuliahManage
4. PerubahanKuliahRequest
5. TranskripManage
6. TranskripRequest

### 3.1.4 Import File Foundation

File **head\_loggedin.php** akan memanggil semua file css terkait dengan file Foundation 6.

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

Kemudian untuk file jquery dan javascript yang digunakan dalam Foundation 6 akan dipanggil dalam file **script.foundation.php**

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

### 3.1.5 Grid System Foundation 6 dalam Website BlueTape

Keseluruhan tampilan pada website menggunakan kelas `float grid` dimana sebelum menggunakan komponen, elemen akan mengimplementasikan kelas:

- `.row`: Halaman website BlueTape yang memiliki satu konten akan diletakkan dalam kelas ini.
  - Halaman Transkrip Manage
  - Halaman Perubahan Kuliah Manage
- `.row column`: Untuk halaman yang memiliki lebih dari satu konten akan dimasukan dalam kombinasi kelas dari `.row` dan `.column` sehingga konten berbaris secara vertikal. Halaman yang menggunakan kelas ini adalah:
  - Halaman Perubahan Kuliah Request
  - Halaman Transkrip Request
  - Halaman Lihat Jadwal Dosen
  - Halaman Entri Jadwal Dosen

### 3.1.6 Navigation Bar



Gambar 3.1: Analisis Tampilan Login

Navigation bar diaplikasikan untuk keseluruhan tampilan website, pada layar medium dan small daftar menu akan berubah menjadi ikon menu. Kelas yang digunakan adalah sebagai berikut.

- `.top-bar` : Menu akan terletak pada bagian atas dari halaman.
- `.menu` : Kelas merupakan fondasi untuk membangun komponen dalam sebuah navigasi seperti daftar menu, judul dan letak menu.
- `.menu-active`: Kelas untuk menandakan menu yang dipilih user.
- `.menu-text`: Kelas untuk menyelaraskan nama menu berbentuk teks agar sejajar dengan *navigation bar*.
- `.top-bar-left`: Kelas yang mengatur daftar menu disebelah kiri.
- `.top-bar-right`: Kelas yang mengatur daftar menu disebelah kanan.

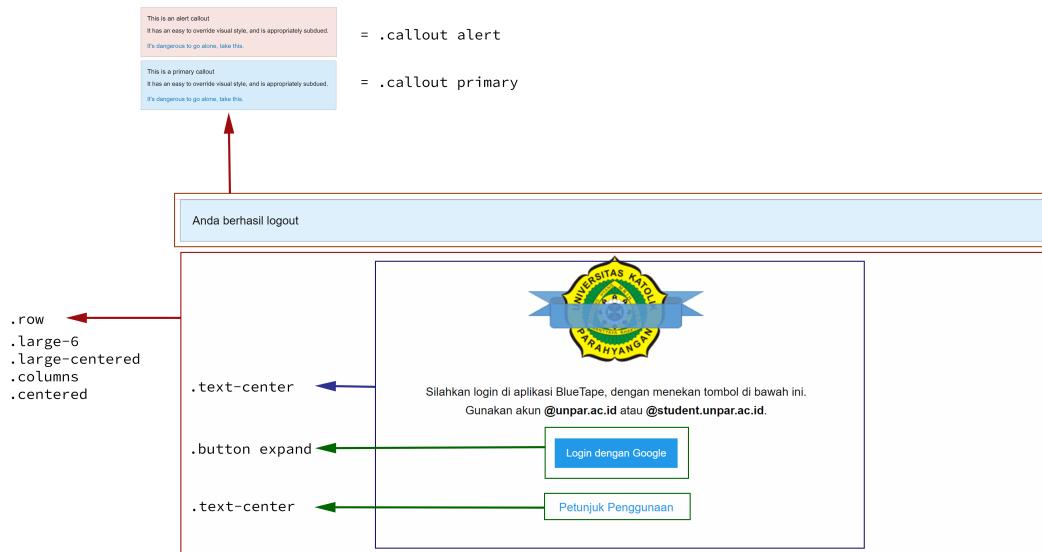


Gambar 3.2: Analisis Tampilan Login pada layar Medium dan Small

Kemudian pada layar *mobile* digunakan komponen *Advanced Layout* dimana daftar menu akan dalam mode *hide*.

- **.title-bar data-responsive-toggle:** Inisiasi untuk membuat navigasi menu yang responsif.
- **.menu-icon:** Kelas untuk membuat icon menu.
- **.title-bar-title :** Logo digunakan untuk judul website BlueTape, sehingga akan terletak pada bagian kanan dari *navigation bar*.
- **data-toggle:** Atribut ini akan memanggil data yang disimpan dalam **data-toggle**.
- **data-hide-for:** Atribut yang mengatur kapan menu navigasi akan responsif.

### 3.1.7 Halaman Login



Gambar 3.3: Analisis Tampilan Login

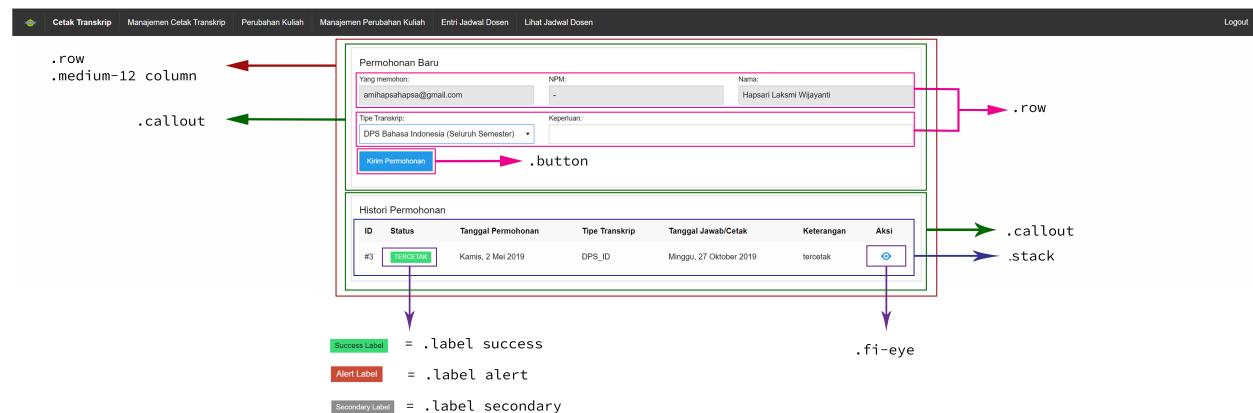
Kelas yang digunakan dalam halaman ini sebagai berikut:

- **.row, .large-6, .column:** Konten akan terletak sejajar secara horizontal.
- **.text-center:** Kalimat login terletak ditengah container.

- **.button expnad:** Tombol akan memiliki panjang yang menyesuaikan lebar konten.
- **Callout]:** Terdapat dua jenis kelas yang dipakai
  - **.callout alert :** Notifikasi bahwa user harus login untuk mengakses website.
  - **.callout primary :** Notifikasi bahwa user berhasil logout.

### 3.1.8 Halaman Cetak Transkrip

#### Halaman Utama



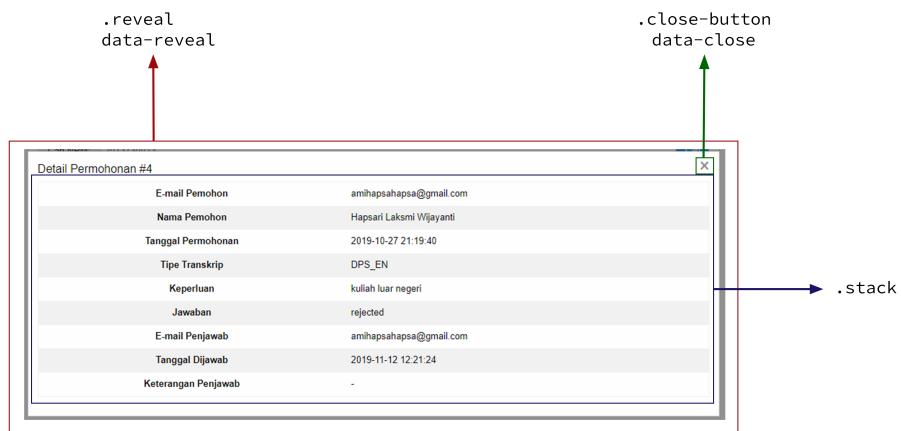
Gambar 3.4: Analisis Halaman Cetak Transkrip

- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* menjadi satu baris.
- **.medium-12 column:** Mengatur agar pada layar medium, semua konten akan selebar 12 grid.
- **.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.
- **Kode Program-eye:** Ikon Font Awesome yang digunakan untuk link ke modal lihat.
- **.button:** Jenis kelas yang digunakan pada tombol ‘Kirim Permohonan’
- **.stack**

- Label: Terdiri dari tiga jenis kelas:
  - `.label success`: Label untuk transkrip yang telah tercetak
  - `.label alert`: Label untuk transkrip yang gagal tercetak
  - `.label secondary`: Label untuk transkrip yang belum tercetak

## Modal

Ikon eye akan menampilkan sebuah modal yang menampilkan Detail Permohonan berdasarkan ID yang tercatat.



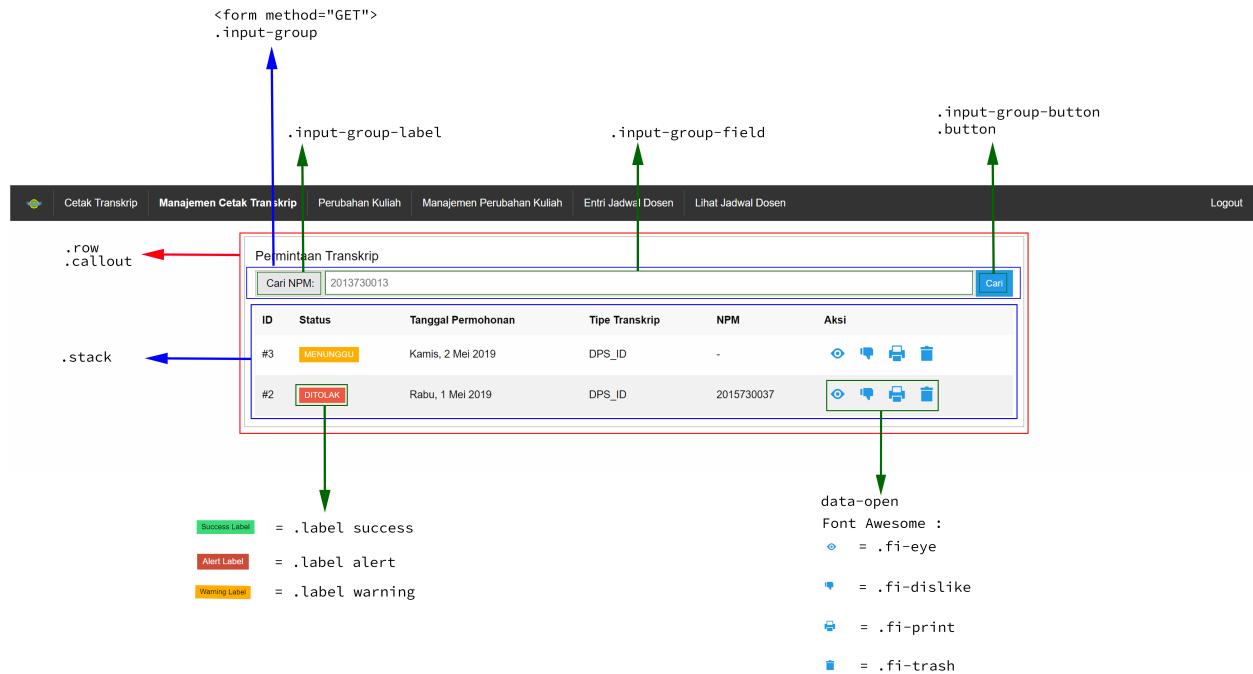
Gambar 3.5: Analisis Modal Lihat

Komponen Modal terdiri dari:

- `.reveal data-reveal`: Membuat modal yang menampung tabel detail permohonan.
- `.close-button data-close`: Menutup modal yang telah terbuka.
- `.stack`: Membuat tabel detail permohonan.

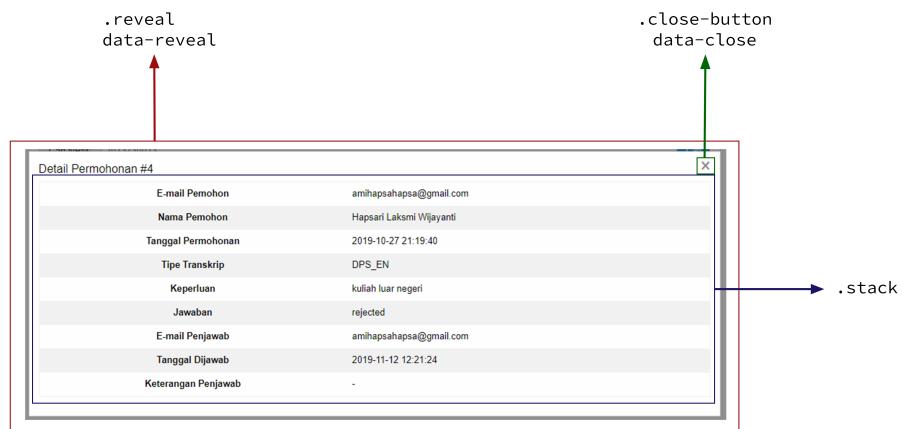
### 3.1.9 Halaman Manajemen Cetak Transkrip

#### Halaman Utama



Gambar 3.6: Analisis Manajemen Cetak Transkrip

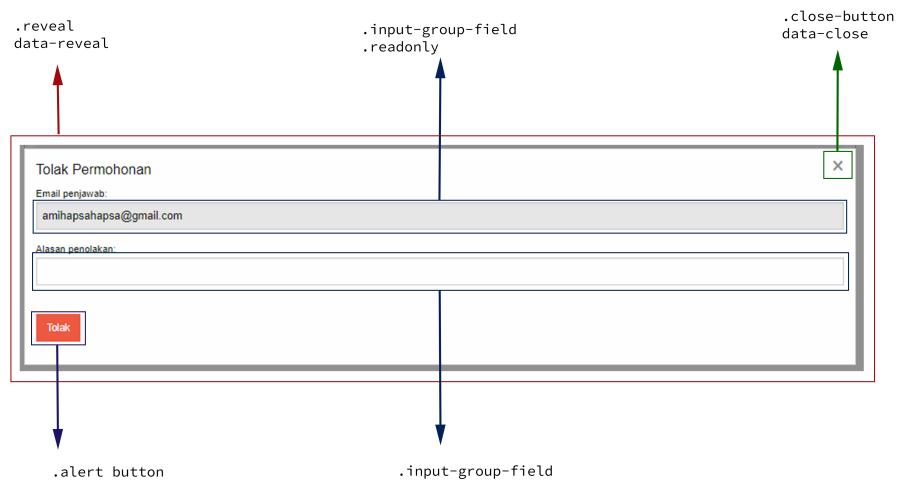
- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* menjadi satu baris.
- **.callout:** Untuk membuat border yang memisahkan konten permohonan baru dan histori permohonan.
- **.stack:** Jenis tabel yang digunakan tabel histori permohonan, sehingga pada layar medium tabel akan tersusun secara bertumpuk.
- Ikon Font Awesome yang terdiri dari
  - **Kode Program-eye:** Ikon untuk menuju modal lihat transkrip.
  - **Kode Program-dislike:** Ikon untuk menuju modal tolak cetak transkrip.
  - **Kode Program-print:** Ikon untuk menuju modal cetak transkrip.
  - **Kode Program-trash:** Ikon untuk menuju hapus permitaan transkrip.
- Label: Terdiri dari tiga jenis kelas:
  - **.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



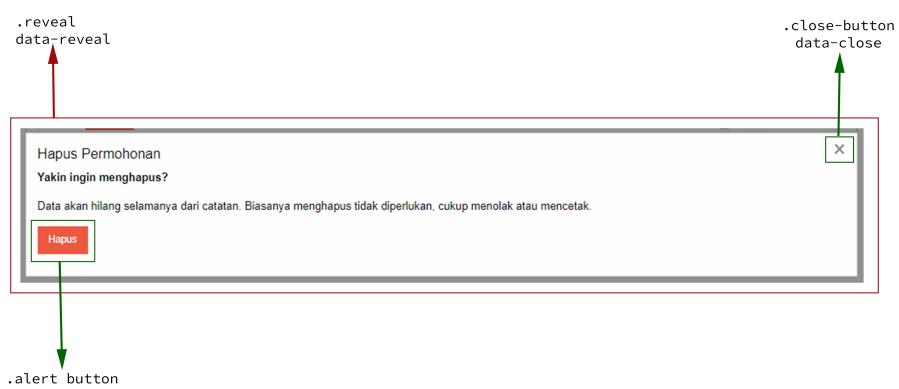
Gambar 3.7: Analisis Modal Lihat



Gambar 3.8: Analisis Modal Print



Gambar 3.9: Analisis Modal Tolak



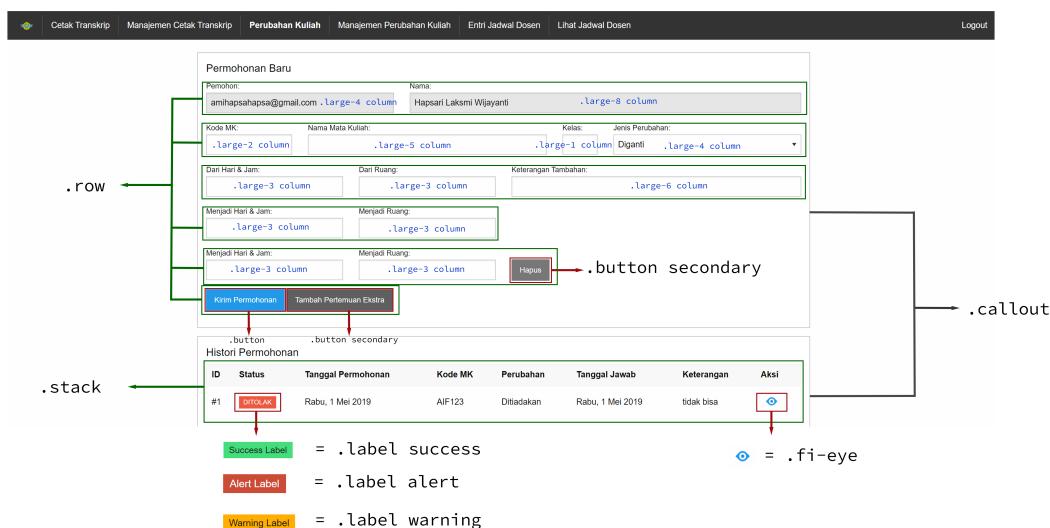
Gambar 3.10: Analisis Modal Hapus

Komponen Modal terdiri dari:

- **.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.

### 3.1.10 Halaman Perubahan Kuliah

#### Halaman Utama

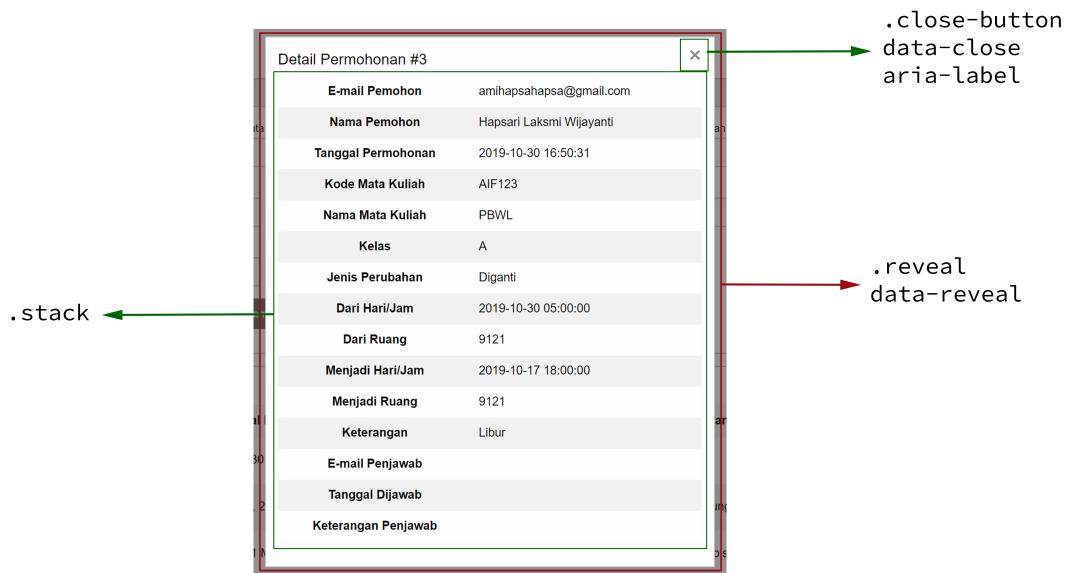


Gambar 3.11: Analisis Tampilan Perubahan Kuliah

- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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.
- Ikon Font Awesome yang terdiri dari
  - Kode Program-eye data-open: Ikon untuk menuju modal lihat detail permohonan berdasarkan ID.
- Label: Terdiri dari tiga jenis kelas:

- `.label success`: Label untuk perubahan kuliah berhasil.
- `.label alert`: Label untuk perubahan kuliah gagal.
- `.label warning`: Label untuk perubahan kuliah diproses.

## Modal



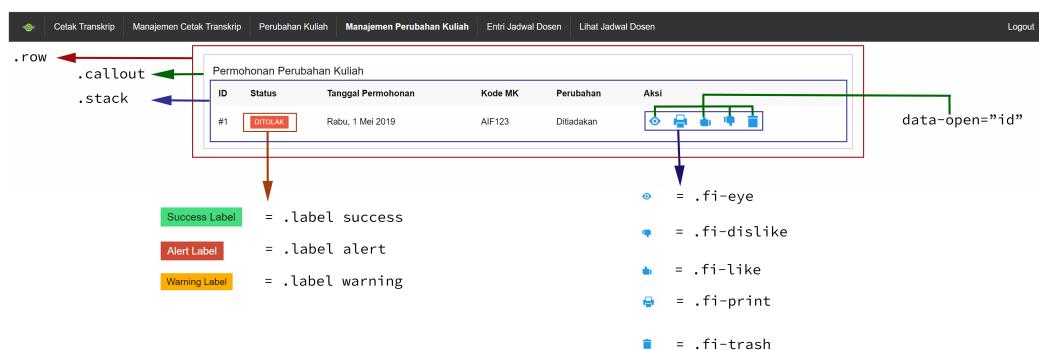
Gambar 3.12: Analisis Modal Lihat

Komponen Modal terdiri dari:

- `.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.

### 3.1.11 Halaman Manajemen Perubahan Kuliah

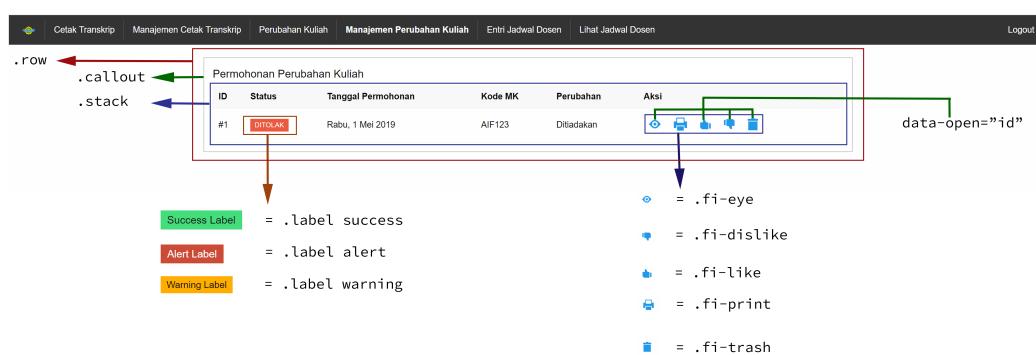
#### Halaman Utama



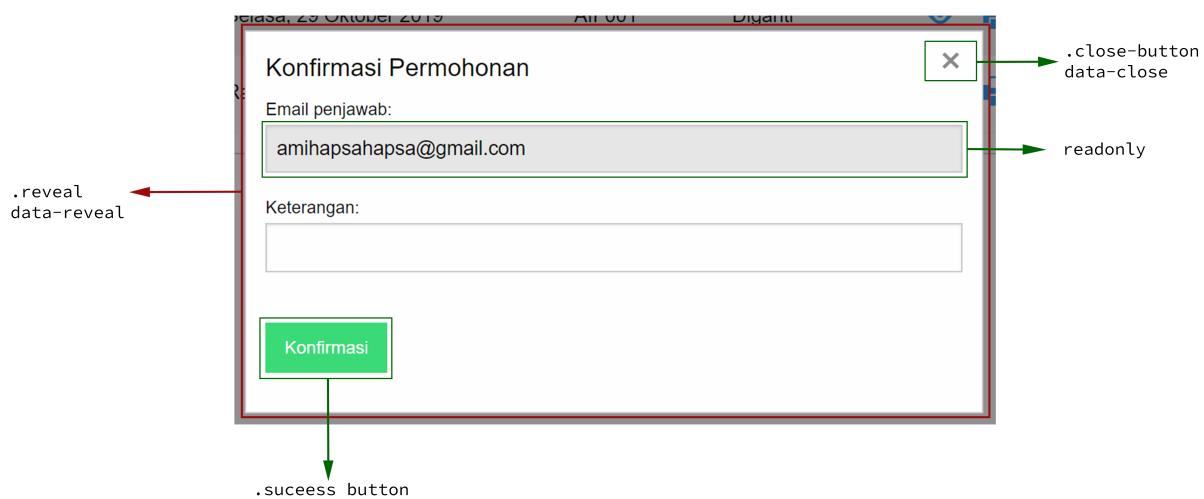
Gambar 3.13: Analisis Halaman Manajemen Perubahan Kuliah

- `.row`: Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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.
- Ikon Font Awesome yang terdiri dari
  - **Kode Program-eye**: Ikon menuju modal lihat permohonan perubahan kuliah.
  - **Kode Program-like**: Ikon menuju modal persetujuan permohonan perubahan kuliah.
  - **Kode Program-dislike**: Ikon menuju modal persetujuan penolakan perubahan kuliah.
  - **Kode Program-print**: Ikon untuk menuju halaman cetak jadwal perubahan kuliah.
  - **Kode Program-trash**: Ikon untuk menghapus permitaan perubahan kuliah.
- Label: Terdiri dari tiga jenis kelas:
  - `.label success`: Label untuk permitaan perubahan kuliah telah disetujui.
  - `.label alert`: Label untuk permitaan perubahan kuliah telah ditolak.
  - `.label warning`: Label untuk permintaan perubahan kuliah sedang diproses.

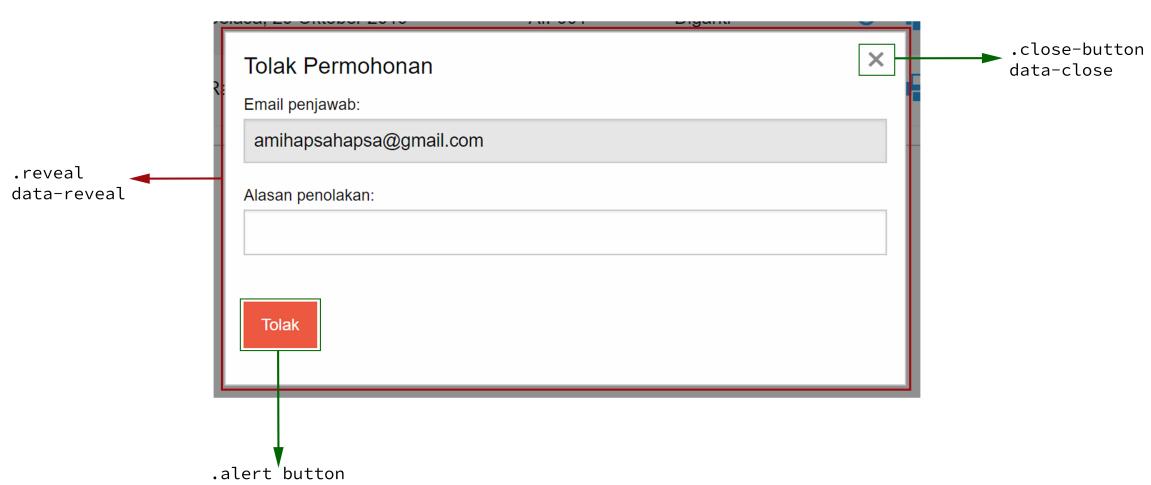
## Modal



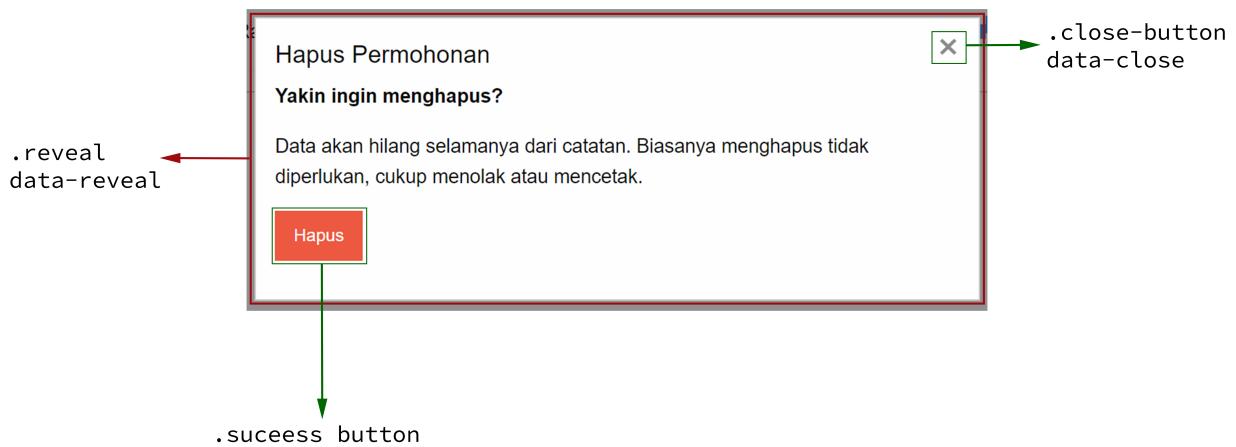
Gambar 3.14: Analisis Modal Lihat



Gambar 3.15: Analisis Modal Setuju



Gambar 3.16: Analisis Modal Tolak



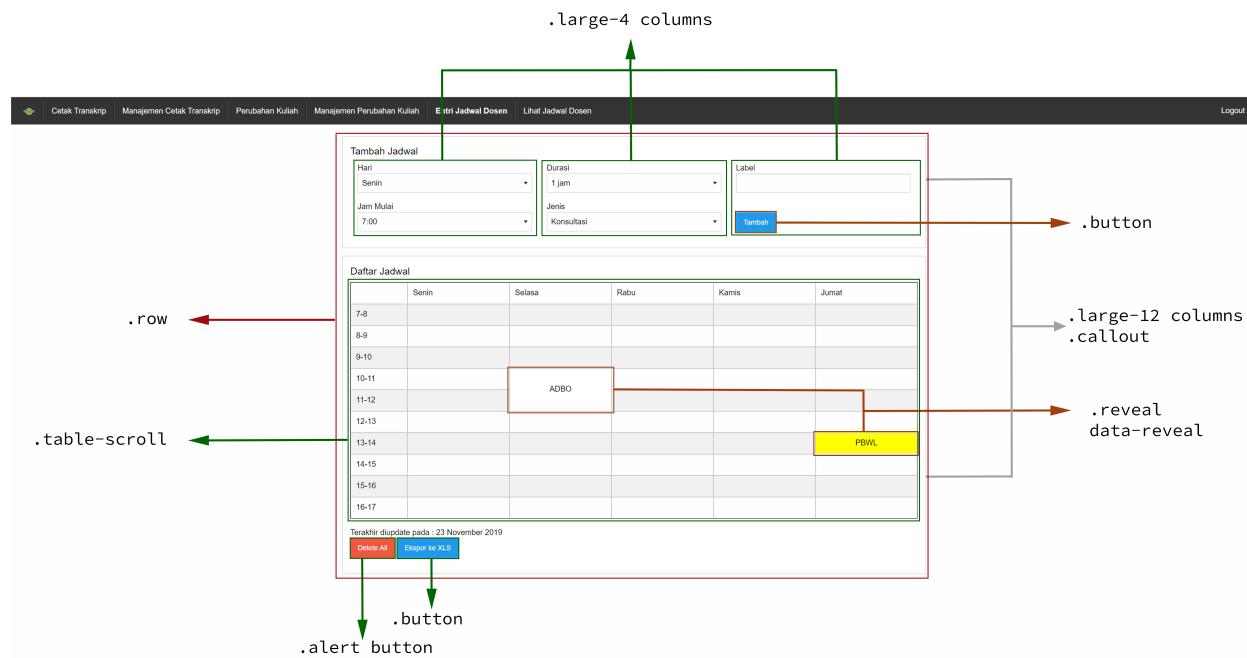
Gambar 3.17: Analisis Modal Hapus

Komponen Modal terdiri dari:

- **.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.
- **.alert button:** Membuat button pada tombol 'tolak' dan 'hapus'.

### 3.1.12 Halaman Entri Jadwal Dosen

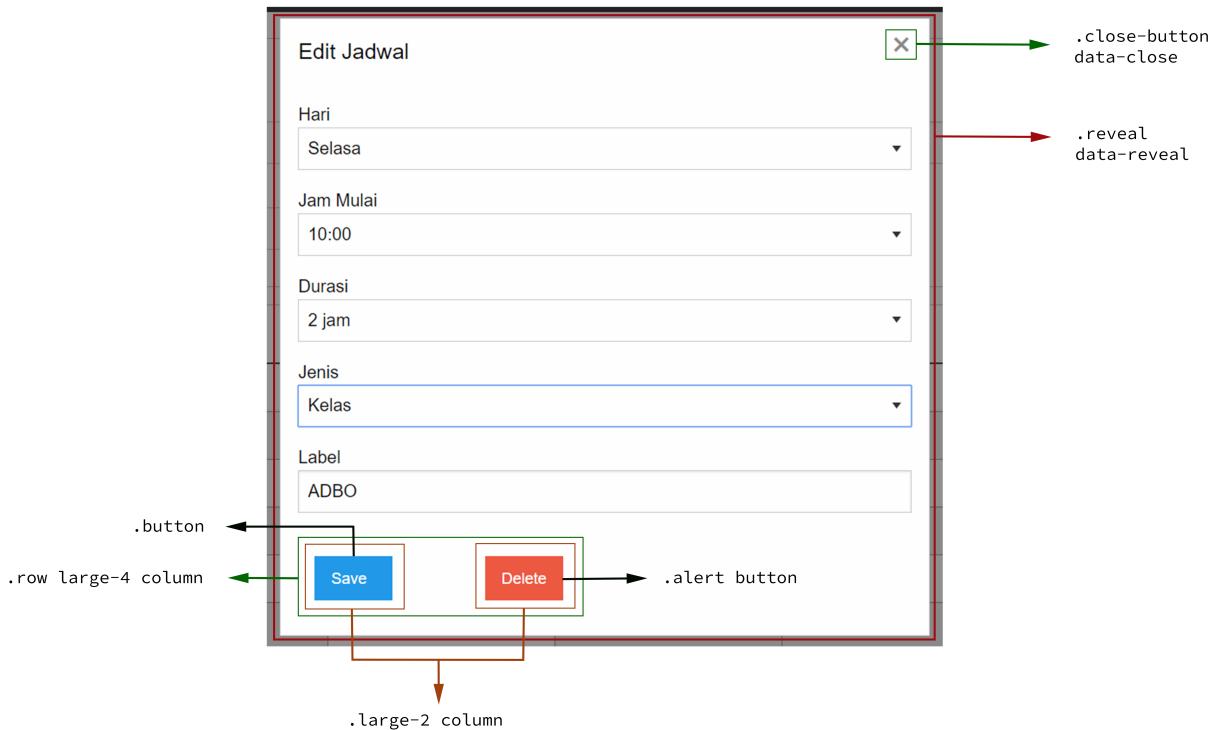
#### Halaman Utama



Gambar 3.18: Analisis Halaman Entri Jadwal Dosen

- **.row**: Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* menjadi satu baris.
- **.large-4 column**: Setiap field akan pada *form* Tambah Jadwal akan memiliki lebar masing-masing 4 grid pada layar *large*.
- **.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 'Ekspor ke XLS' untuk konten Daftar Jadwal dan 'Tambah' pada konten Tambah Jadwal.
- **.alert button**: Membuat button pada tombol 'Delete All'.

## Edit Jadwal

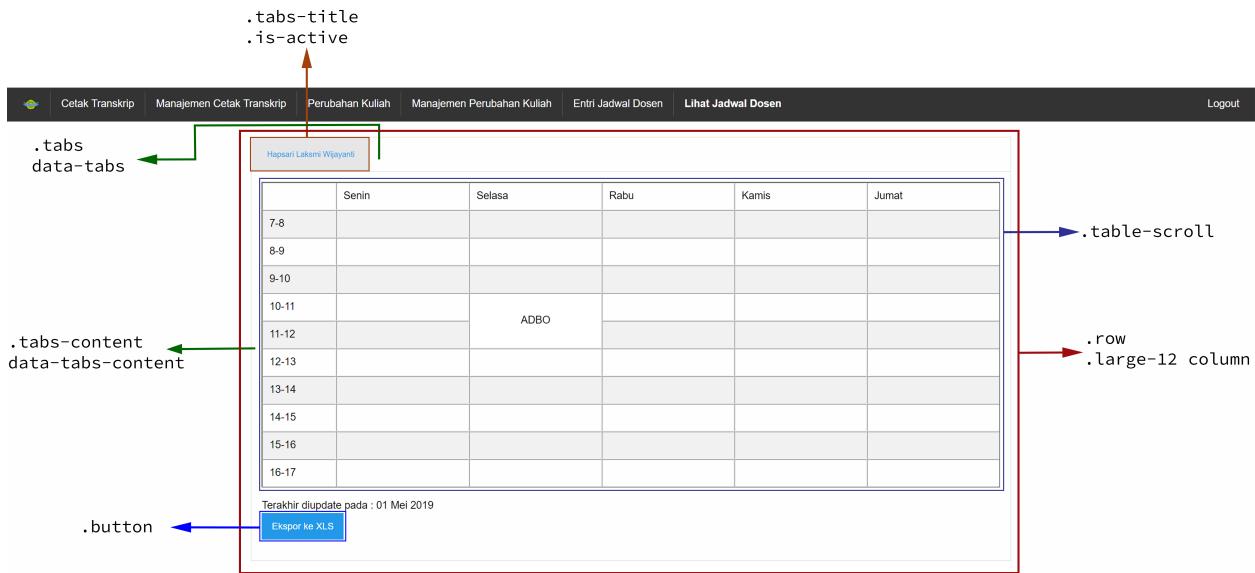


Gambar 3.19: Analisis Modal Edit Jadwal

Komponen Modal terdiri dari:

- **.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'.

### 3.1.13 Halaman Lihat Jadwal Dosen



Gambar 3.20: Analisis Modal Edit Jadwal

- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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 digerakan secara horizontal.
- **button:** Membuat button pada tombol ‘Eksport 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 judul tabs nama dosen.
- **.is-active:** Menunjukkan tabs nama dosen yang sedang dilihat

## 3.2 Analisis Perubahan Kelas yang digunakan dari Foundation 6 ke Bootstrap 4 pada BlueTape

Bagian ini menjelaskan detail penggunaan kelas-kelas saat mengaplikasikan framework Foundation 6 dan detail kelas-kelas pada Bootstrap 4 yang nantinya akan digunakan dalam website BlueTape terbaru.

### Navigasi Menu

Foundation 6	Bootstrap 4
.title-bar-title	.navbar-brand
.menu	.top-bar
.menu-active	.active
.menu-text	.nav-item, .nav-link
.top-bar-left	.ml-auto
.top-bar-right	.mr-auto
.navbar-dark, .bg-dark	-

Navigasi menu memiliki kelas sebagai berikut:

### Halaman Login

Foundation 6	Bootstrap 4
.row, .column	.container
.row	.row
.large-centered	.justify-content-center
.large-6	.col-lg-6
.text-center	.text-center
.button	.btn
-	.btn-primary
.expand	.btn-lg
.callout alert	-
.callout primary	-
-	.btn-lg

### Halaman Cetak Transkrip

Foundation 6	Bootstrap 4
.row	.container
.callout	.card, .card-header, .card-body
.medium-12	.col-md-12
.larger-4	.col-lg-4
.large-8	.col-lg-8
.column	.col
.button	.btn btn-primary
.label success	.badge badge-success
.label alert	.badge badge-danger
.label secondary	.badge badge-secondary
.fi-eye	.fas fa-eye
.stack	.table table-striped
-	.col-form-label
-	.form-control

### Halaman Manajemen Cetak Transkrip

Foundation 6	Bootstrap 4
.row	.container
.callout	.card, .card-header, .card-body
.medium-12	.col-md-12
.larger-4	.col-lg-4
.large-8	.col-lg-8
.column	.col
.button	.btn btn-primary
.label success	.badge badge-success
.label alert	.badge badge-danger
.label warning	.badge badge-warning
.fi-eye	.fas fa-eye
.fi-dislike	.fas fa-thumbs-down
.fi-print	.fas fa-print
.fi-trash	.fas fa-trash
.stack	.table table-striped
.input-group	.input-group
.input-group-label	.input-group-prepend
.input-group-field	.form-control
.button, .input-group-button	.btn btn-outline-primary

### Halaman Perubahan Kuliah

Foundation 6	Bootstrap 4
.row	.container
.callout	.card, .card-header, .card-body
.medium-* column	.col-md-*
.large-* column	.col-lg-*
.column	.col
.button	.btn btn-primary
.label success	.badge badge-success
.label alert	.badge badge-danger
.label secondary	.badge badge-secondary
.fi-eye	.fas fa-eye
.stack	.table table-striped
.input-group	.input-group
.input-group-label	.input-group-prepend
.input-group-field	.form-control
.button	.btn btn-primary
.button secondary	.btn btn-secondary

### Halaman Manajemen Perubahan Kuliah

Foundation 6	Bootstrap 4
.row	.container
.callout	.card, .card-header, .card-body
.label success	.badge badge-success
.label alert	.badge badge-danger
.label warning	.badge badge-warning
.fi-eye	.fas fa-eye
.fi-like	.fas fa-thumbs-up
.fi-dislike	.fas fa-thumbs-down
.fi-print	.fas fa-print
.fi-trash	.fas fa-trash
.stack	.table table-striped
.input-group	.input-group
.input-group-label	.input-group-prepend
.input-group-field	.form-control
data-open	data-toggle, data-target
.button, .input-group-button	.btn btn-outline-primary

### Halaman Entri Jadwal Dosen

Foundation 6	Bootstrap 4
.row	.row
-	.container
.callout	.card, .card-header, .card-body
.medium-4 column	.col-md-4
.label success	.badge badge-success
.label alert	.badge badge-danger
.label warning	.badge badge-warning
.fi-eye	.fas fa-eye
.fi-like	.fas fa-thumbs-up
.fi-dislike	.fas fa-thumbs-down
.fi-print	.fas fa-print
.fi-trash	.fas fa-trash
.stack	.table table-striped
.input-group	.input-group
.input-group-label	.input-group-prepend
.input-group-field	.form-control
.table-scroll	.table, .table-bordered, .table-striped
.button	.btn btn-primary
.alert button	.btn btn-danger

### Halaman Lihat Jadwal Dosen

Foundation 6	Bootstrap 4
-	.container
.row	.row
.callout	.card, .card-header, .card-body
.large-* column	.col-lg-*
.button	.btn btn-primary
.label success	.badge badge-success
.button	.btn btn-primary
.tabs, data-tabs	.nav, .nav-tabs
.tabs-title, .is-active	tabs-panel, .is-active
.tabs-content, data-tabs-content	.tabs-content, data-tabs-content .
.table-scroll	.table, .table-bordered, .table-striped

### Modal

## BAB 4

# IMPLEMENTASI WEBSITE BLUETAPE DENGAN BOOTSTRAP 4

Bab 4 menjelaskan implementasi untuk mengubah seluruh tampilan website dengan framework Bootstrap 4 pada website BlueTape.

### 4.1 Konversi Tampilan dengan Bootstrap 4

Pada bagian ini akan dijabarkan keseluruhan kelas yang dipakai dalam website BlueTape. Pertama akan dijabarkan file framework yang digunakan, kemudian akan dijelaskan komponen yang digunakan dalam website BluTape dan terakhir penggunaan kelas framework Bootstrap 4 pada website.

#### 4.1.1 Menjalankan Framework Bootstrap 4

Website BlueTape menggunakan sebuah framework PHP yaitu CodeIgniter. Hal pertama yang dilakukan saat ingin menjalankan tampilan adalah cek terlebih dimana file yang berkaitan dengan Framework Bootstrap 4 disimpan, lalu dilihat file tersebut dijalankan ketika membangun sebuah tampilan.

#### 4.1.2 Folder untuk Menyimpan File Bootstrap 4 dan Plugin

Seluruh file terkait dengan Framework Bootstrap 4 diletakkan dalam folder `/www/public`. Dimana terdiri dari folder:

1. css: Terdiri dari file - file css dari beserta file ikon bawaan dari framework Bootstrap 4.
2. img: Menyimpan sebuah gambar logo BlueTape.
3. js: Menyimpan file javascript bawaan dari Bootstrap 4 dan file jquery yang tersimpan dalam folder vendor.
4. lib: Menyimpan plugin xdan-datetimepicker.

#### 4.1.3 Folder untuk Implementasi Framework Bootstrap 4 dan Plugin

Untuk melihat code tampilan website, maka *user* akan mengakses file - file dalam folder `/www/application/views`. Folder views terdiri dari beberapa sub-folder. Secara garis besar, file terdiri dari dua fungsi yaitu menyimpan kode template yang digunakan untuk keseluruhan tampilan dan implementasi komponen - komponen Bootstrap 4 untuk membangun sebuah website.

Pertama untuk sub-folder template terdiri dari:

1. auth : Berisi file terkait dengan framework Bootstrap 4 akan diimport disini, kemudian akan disertakan sebuah tampilan login dan notifikasi - notifikasi terkait dengan status login. File tersebut tercatat di `login.php`.

2. error : Berisi folder untuk mengatasi error ketika user gagal untuk *load* tampilan. Pesan akan ditampilkan pada *user* ketika error terjadi. Folder ini sudah ada sejak pertama mengunduh framework CodeIgniter.
3. templates : Folder ini mencangkup kode template yang digunakan untuk seluruh bagian website BlueTape.

Lalu sub-folder template yang berfungsi untuk menyimpan kode tampilan website terdiri 6 folder. Masing - masing folder memiliki sebuah file **main.php**. Sub-folder tersebut yaitu:

1. EntriJadwalDosen
2. LihatJadwalDisen
3. PerubahanKuliahManage
4. PerubahanKuliahRequest
5. TranskripManage
6. TranskripRequest

#### 4.1.4 Import File Bootstrap 4

File **head\_loggedin.php** akan memanggil semua file css terkait dengan file Bootstrap 4.

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

Kemudian untuk file jquery dan javascript yang digunakan dalam Bootstrap 4 akan dipanggil dalam file **script\_Bootstrap 4.php**

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

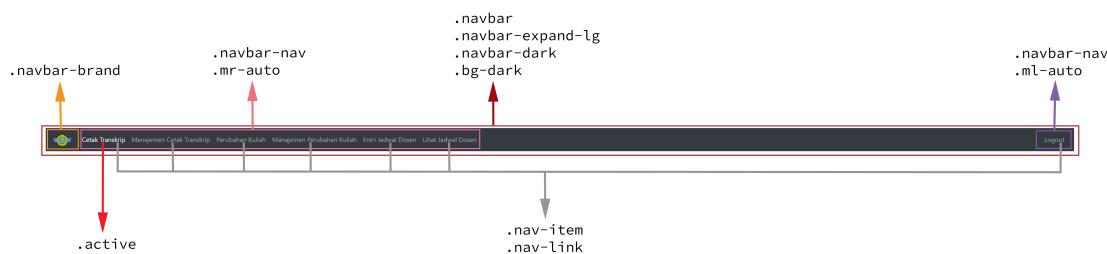
#### 4.1.5 Grid System Bootstrap 4 dalam Website BlueTape

Keseluruhan tampilan pada website menggunakan kelas **float grid** dimana sebelum menggunakan komponen, elemen akan mengimplementasikan kelas:

- **.container**: Halaman website BlueTape yang memiliki satu konten akan diletakkan dalam kelas ini.

- Halaman Transkrip Manage
- Halaman Perubahan Kuliah Manage
- **.container column:** Untuk halaman yang memiliki lebih dari satu konten akan dimasukan dalam kombinasi kelas dari **.row** dan **.column** sehingga konten berbaris secara vertikal. Halaman yang menggunakan kelas ini adalah:
  - Halaman Perubahan Kuliah Request
  - Halaman Transkrip Request
  - Halaman Lihat Jadwal Dosen
  - Halaman Entri Jadwal Dosen

#### 4.1.6 Navigation Bar

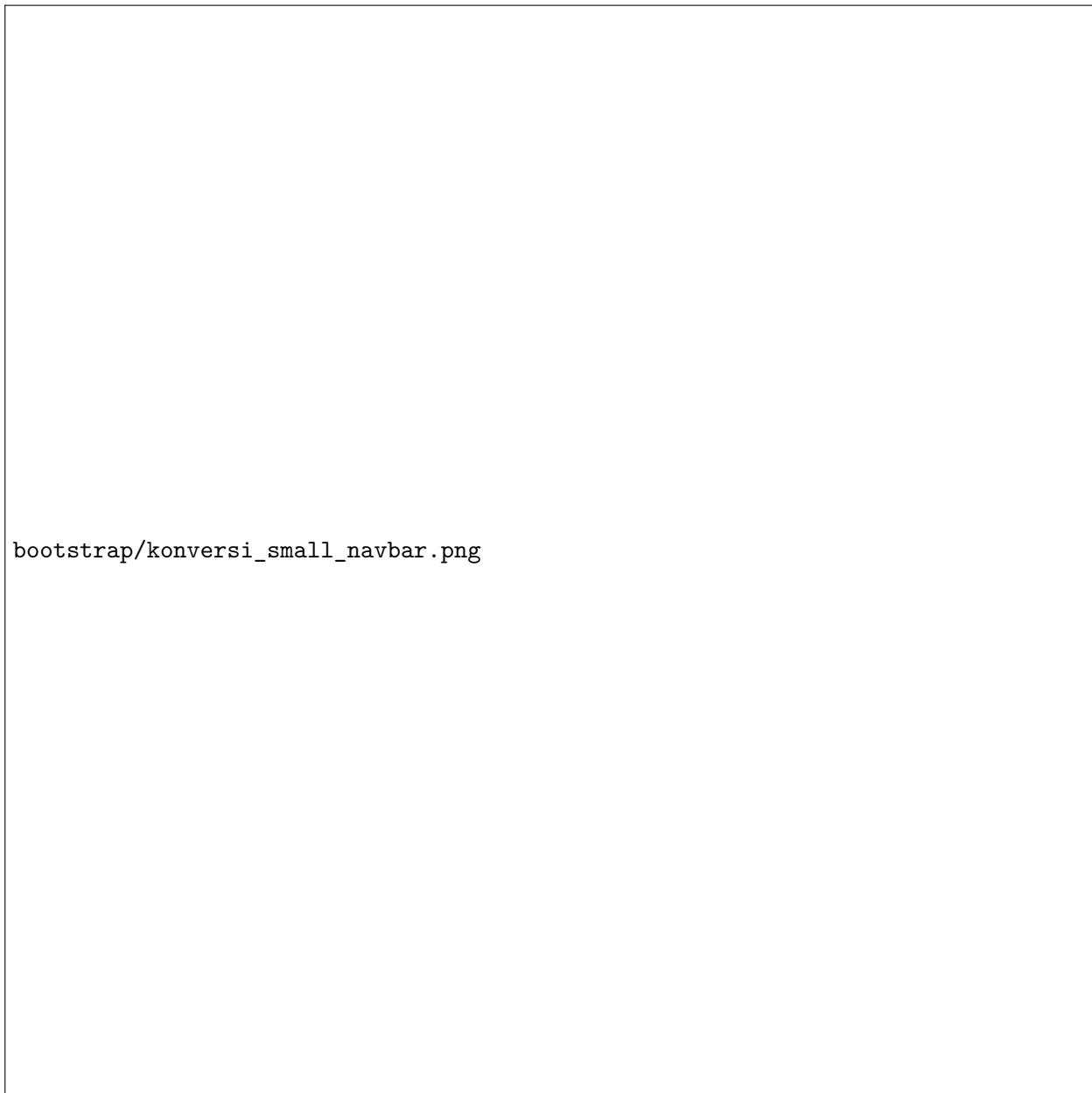


Gambar 4.1: Konversi Tampilan Login

Navigation bar diaplikasikan untuk keseluruhan tampilan website, pada layar medium dan small daftar menu akan berubah menjadi ikon menu. Kelas yang digunakan adalah sebagai berikut.

- **.top-bar :** Menu akan terletak pada bagian atas dari halaman.
- **.menu :** Kelas merupakan fondasi untuk membangun komponen dalam sebuah navigasi seperti daftar menu, judul dan letak menu.
- **.menu-active:** Kelas untuk menandakan menu yang dipilih user.
- **.menu-text:** Kelas untuk menyelaraskan nama menu berbentuk teks agar sejajar dengan *navigation bar*.

- **.top-bar-left:** Kelas yang mengatur daftar menu disebelah kiri.
- **.top-bar-right:** Kelas yang mengatur daftar menu disebelah kanan.



bootstrap/konversi\_small\_navbar.png

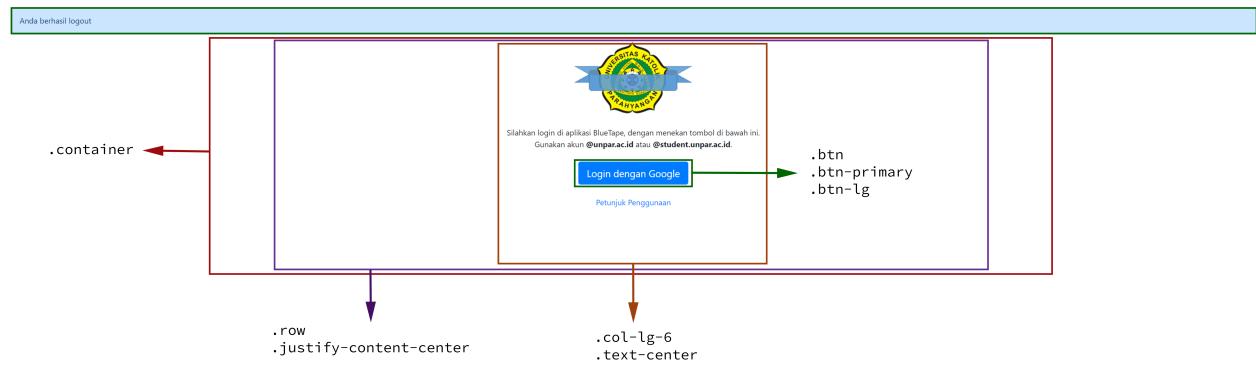
Gambar 4.2: Analisis Tampilan Login pada layar Medium dan Small

Kemudian pada layar *mobile* digunakan komponen *Advanced Layout* dimana daftar menu akan dalam mode *hide*.

- **.title-bar data-responsive-toggle:** Inisiasi untuk membuat navigasi menu yang responsif.
- **.menu-icon:** Kelas untuk membuat icon menu.
- **.title-bar-title :** Logo digunakan untuk judul website BlueTape, sehingga akan terletak pada bagian kanan dari *navigation bar*.

- **data-toggle:** Atribut ini akan memanggil data yang disimpan dalam **data-toggle**.
- **data-hide-for:** Atribut yang mengatur kapan menu navigasi akan responsif.

#### 4.1.7 Halaman Login



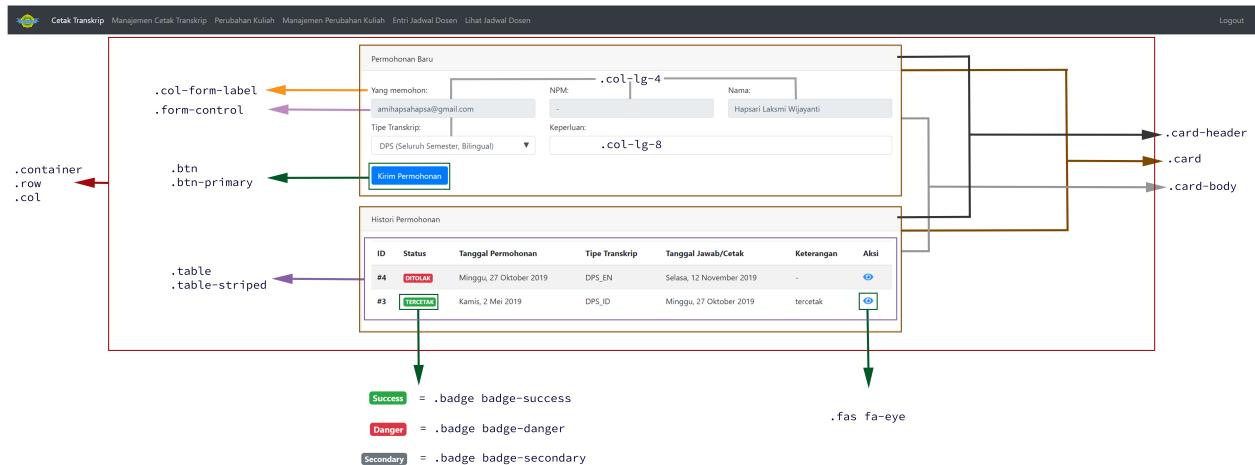
Gambar 4.3: Konversi Tampilan Login

Kelas yang digunakan dalam halaman ini sebagai berikut:

- **.row, .large-6, .column:** Konten akan terletak sejajar secara horizontal.
- **.text-center:** Kalimat login terletak ditengah container.
- **.button expand:** Tombol akan memiliki panjang yang menyesuaikan lebar konten.
- **Callout]:** Terdapat dua jenis kelas yang dipakai
  - **.callout alert** : Notifikasi bahwa user harus login untuk mengakses website.
  - **.callout primary** : Notifikasi bahwa user berhasil logout.

#### 4.1.8 Halaman Cetak Transkrip

##### Halaman Utama

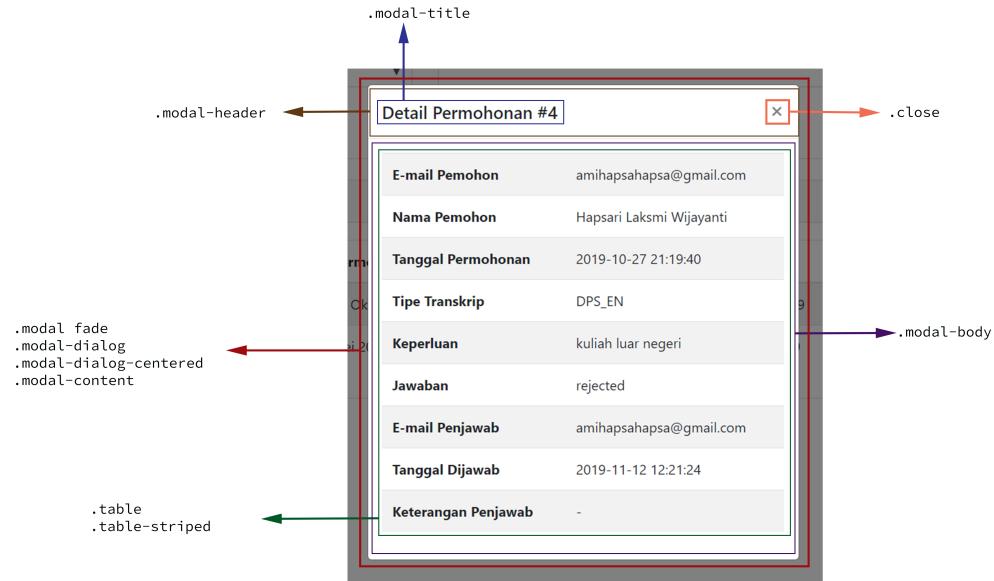


Gambar 4.4: Konversi Halaman Cetak Transkrip

- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* menjadi satu baris.
- **.medium-12 column:** Mengatur agar pada layar medium, semua konten akan selebar 12 grid.
- **.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.
- **.fas fa-eye:** Ikon Font Awesome yang digunakan untuk link ke modal lihat.
- **.button:** Jenis kelas yang digunakan pada tombol 'Kirim Permohonan'
- **.stack**
- Label: Terdiri dari tiga jenis kelas:
  - **.label success:** Label untuk transkrip yang telah tercetak
  - **.label alert:** Label untuk transkrip yang gagal tercetak
  - **.label secondary:** Label untuk transkrip yang belum tercetak

## Modal

Ikon eye akan menampilkan sebuah modal yang menampilkan Detail Permohonan berdasarkan ID yang tercatat.



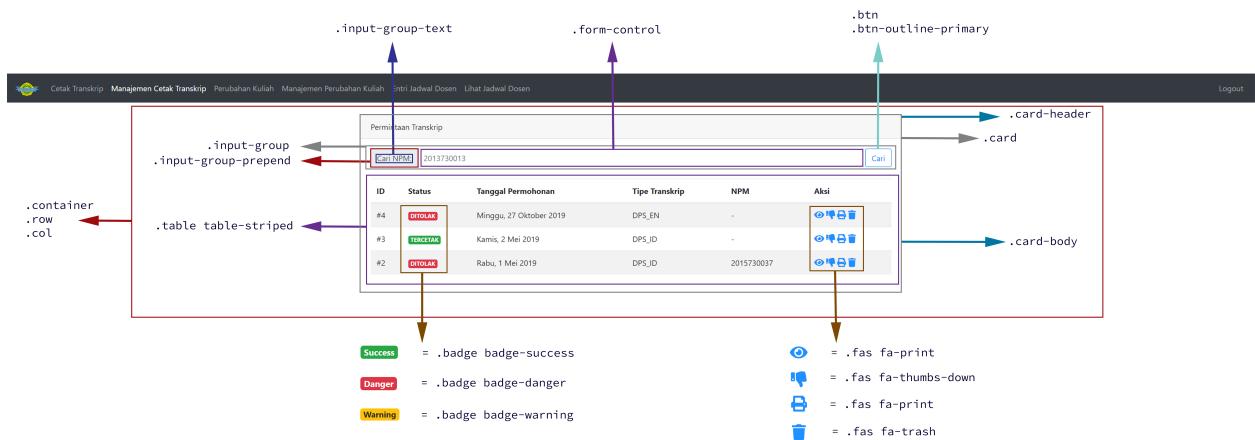
Gambar 4.5: Konversi Modal Lihat

Komponen Modal terdiri dari:

- **.reveal data-reveal**: Membuat modal yang menampung tabel detail permohonan.
- **.close-button data-close**: Menutup modal yang telah terbuka.
- **.stack**: Membuat tabel detail permohonan.

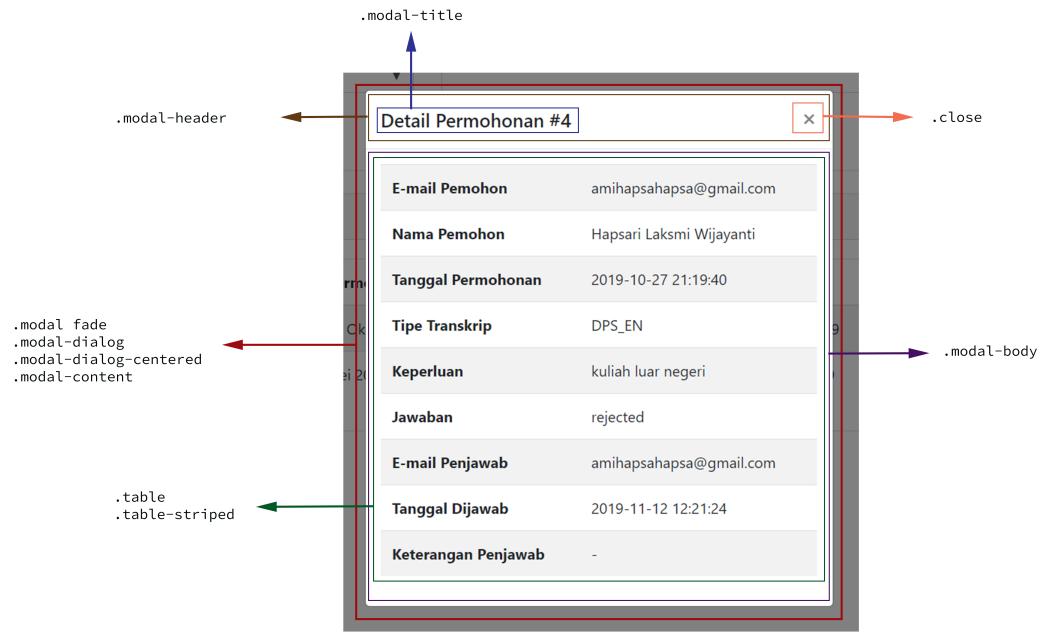
#### 4.1.9 Halaman Manajemen Cetak Transkrip

##### Halaman Utama

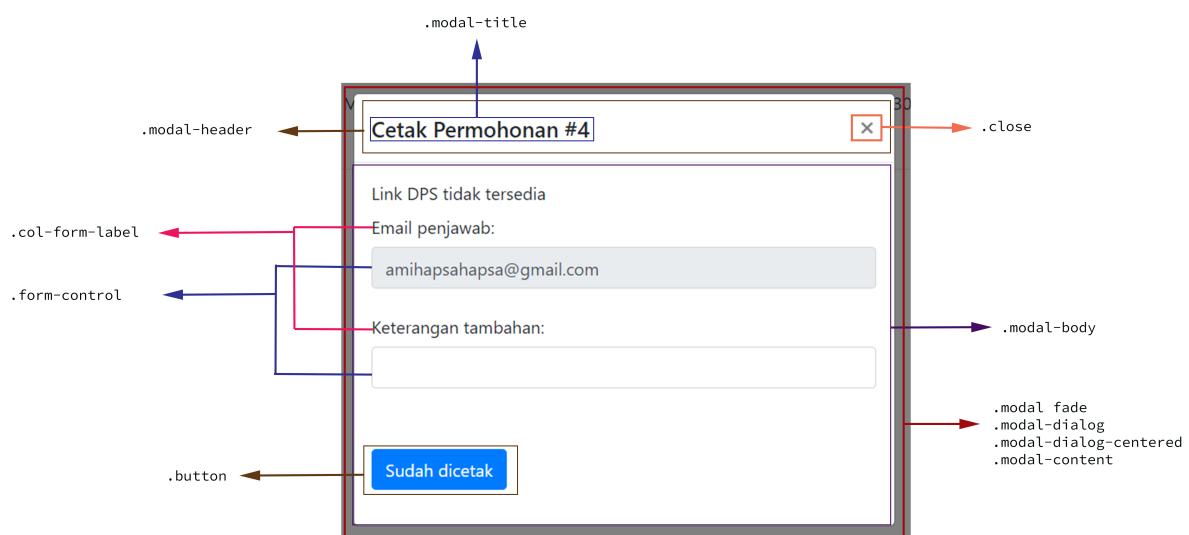


Gambar 4.6: Konversi Manajemen Cetak Transkrip

- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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.
- Ikon Font Awesome yang terdiri dari
  - **.fas fa-eye:** Ikon untuk menuju modal lihat transkrip.
  - **.fas fa-dislike:** Ikon untuk menuju modal tolak cetak transkrip.
  - **.fas fa-print:** Ikon untuk menuju modal cetak transkrip.
  - **.fas fa-trash:** Ikon untuk menuju hapus permitaan transkrip.
- Label: Terdiri dari tiga jenis kelas:
  - **.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



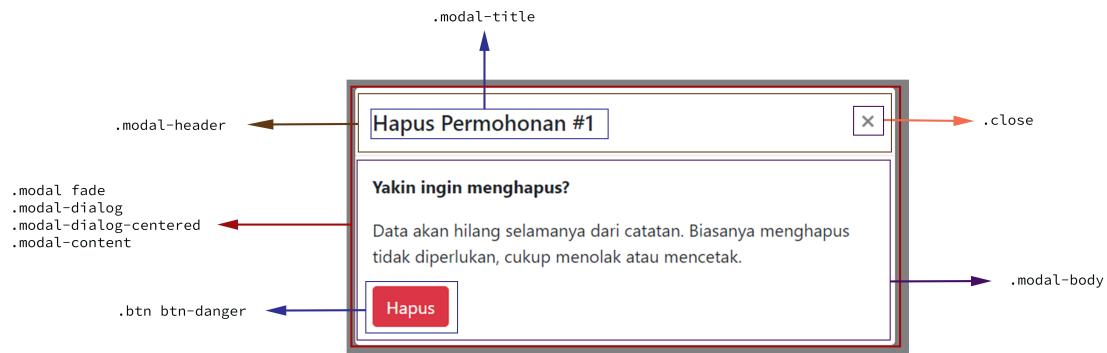
Gambar 4.7: Konversi Modal Lihat



Gambar 4.8: Konversi Modal Print

bootstrap/konversi\_modal\_dislike\_manajemen\_cetak\_transkrip.png

Gambar 4.9: Konversi Modal Tolak



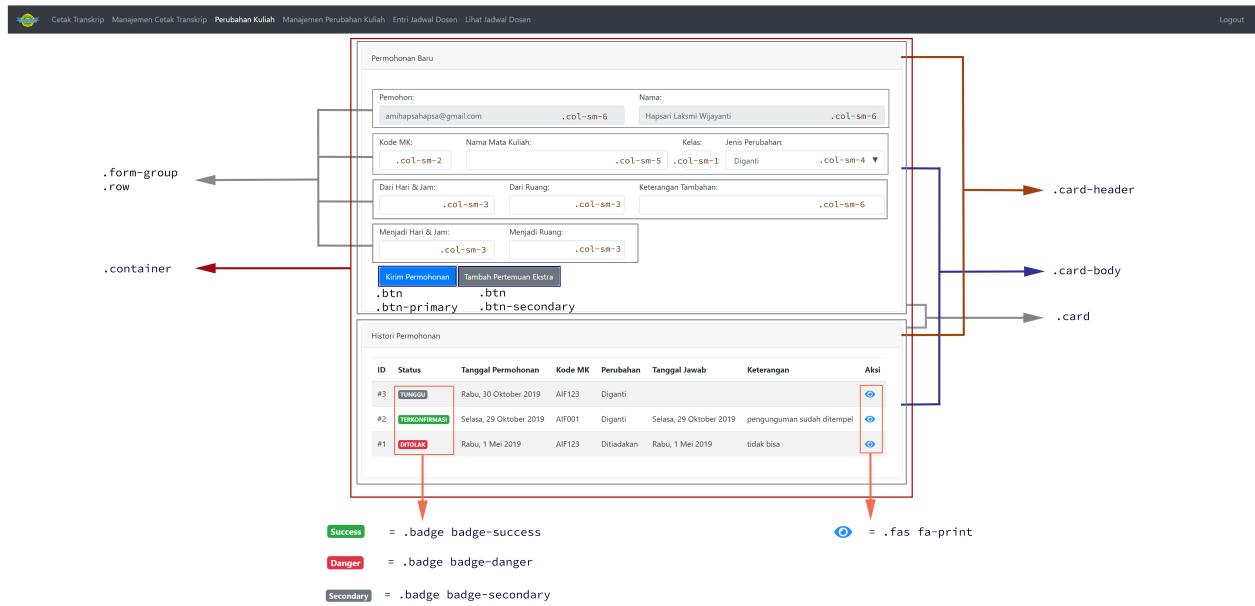
Gambar 4.10: Konversi Modal Hapus

Komponen Modal terdiri dari:

- **.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.

#### 4.1.10 Halaman Perubahan Kuliah

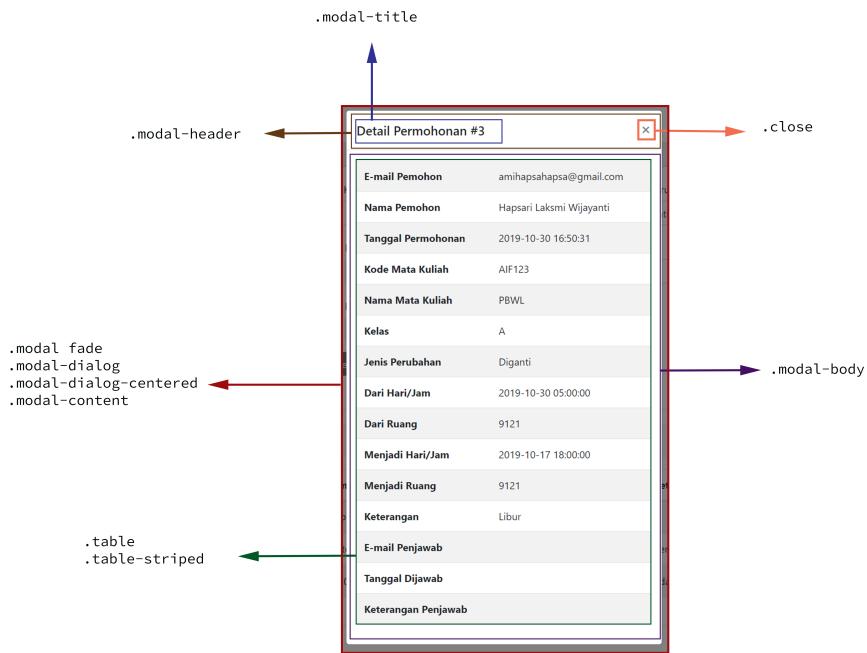
##### Halaman Utama



Gambar 4.11: Konversi Perubahan Kuliah

- **.row**: Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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.
- Ikon Font Awesome yang terdiri dari
  - `.fas fa-eye data-open`: Ikon untuk menuju modal lihat detail permohonan berdasarkan ID.
- Label: Terdiri dari tiga jenis kelas:
  - `.label success`: Label untuk perubahan kuliah berhasil.
  - `.label alert`: Label untuk perubahan kuliah gagal.
  - `.label warning`: Label untuk perubahan kuliah diproses.

## Modal



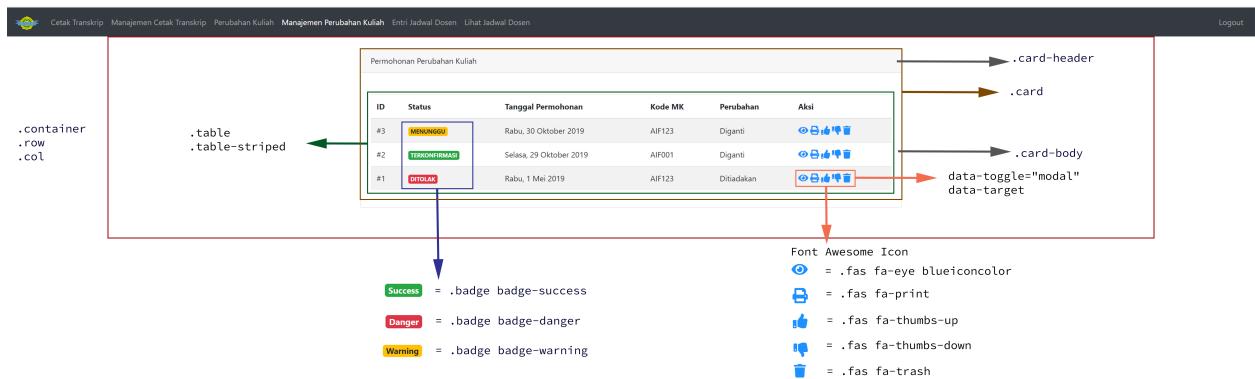
Gambar 4.12: Konversi Modal Lihat

Komponen Modal terdiri dari:

- `.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.

#### 4.1.11 Halaman Manajemen Perubahan Kuliah

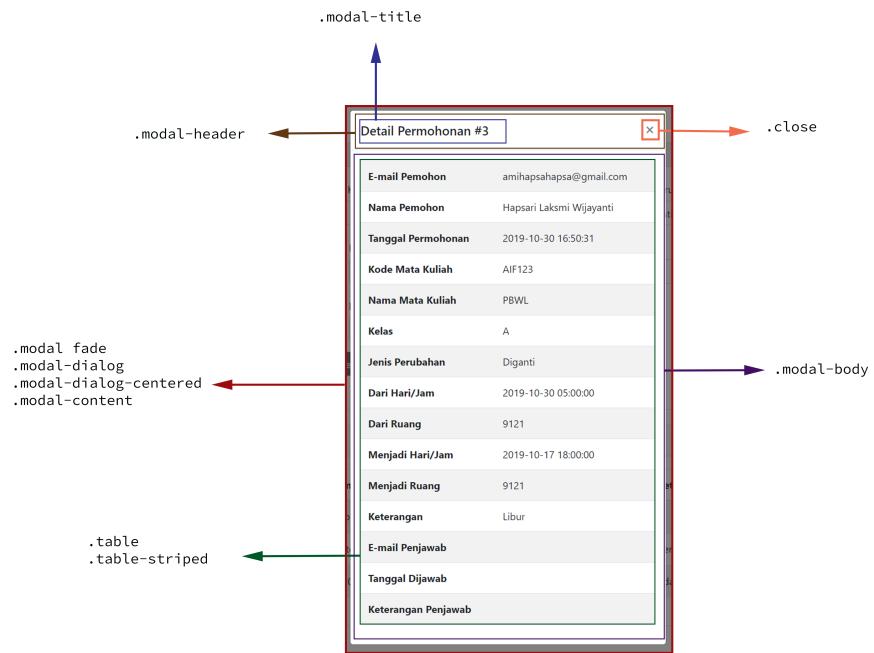
##### Halaman Utama



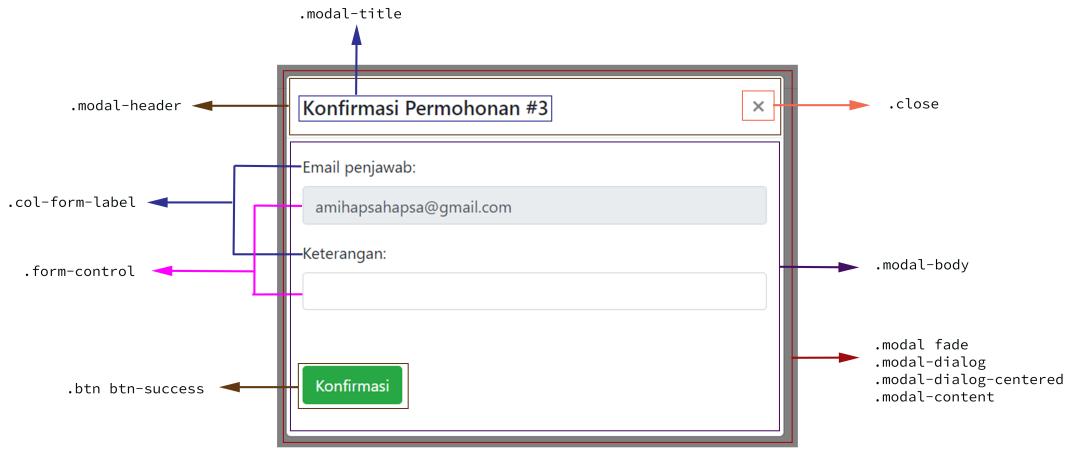
Gambar 4.13: Konversi Halaman Manajemen Perubahan Kuliah

- **.row**: Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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.
- Ikon Font Awesome yang terdiri dari
  - **.fas fa-eye**: Ikon menuju modal lihat permohonan perubahan kuliah.
  - **.fas fa-like**: Ikon menuju modal persetujuan permohonan perubahan kuliah.
  - **.fas fa-dislike**: Ikon menuju modal persetujuan penolakan perubahan kuliah.
  - **.fas fa-print**: Ikon untuk menuju halaman cetak jadwal perubahan kuliah.
  - **.fas fa-trash**: Ikon untuk menghapus permitaan perubahan kuliah.
- Label: Terdiri dari tiga jenis kelas:
  - **.label success**: Label untuk permitaan perubahan kuliah telah disetujui.
  - **.label alert**: Label untuk permitaan perubahan kuliah telah ditolak.
  - **.label warning**: Label untuk permintaan perubahan kuliah sedang diproses.

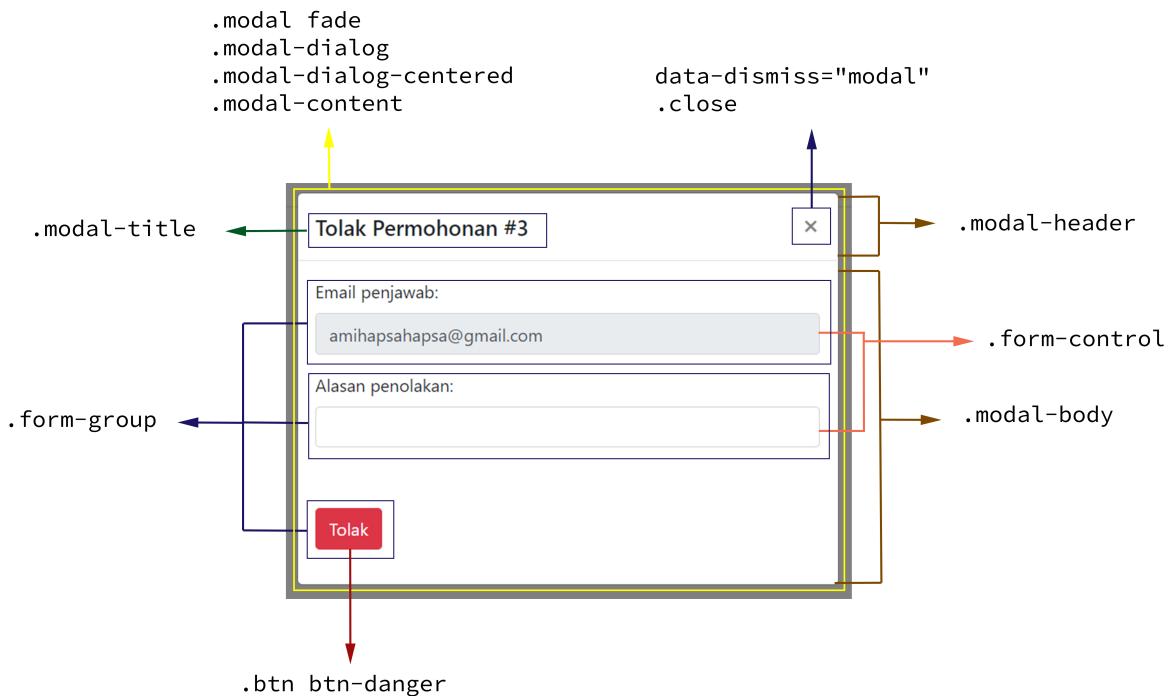
## Modal



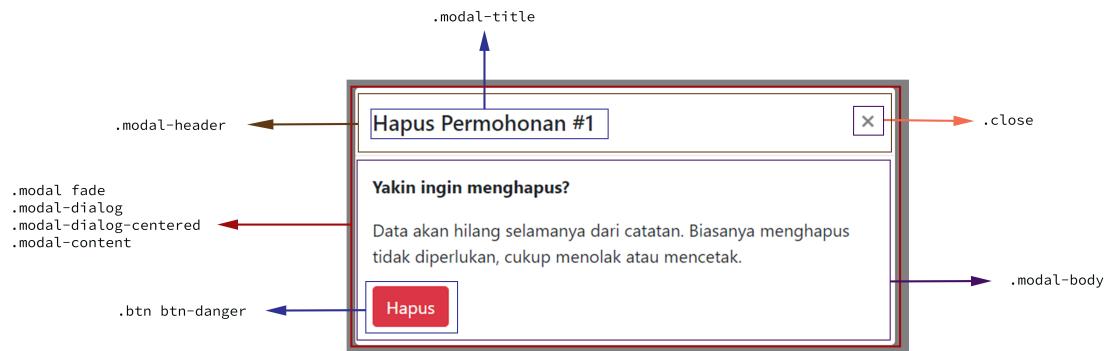
Gambar 4.14: Konversi Modal Lihat



Gambar 4.15: Analisis Modal Setuju



Gambar 4.16: Konversi Modal Tolak



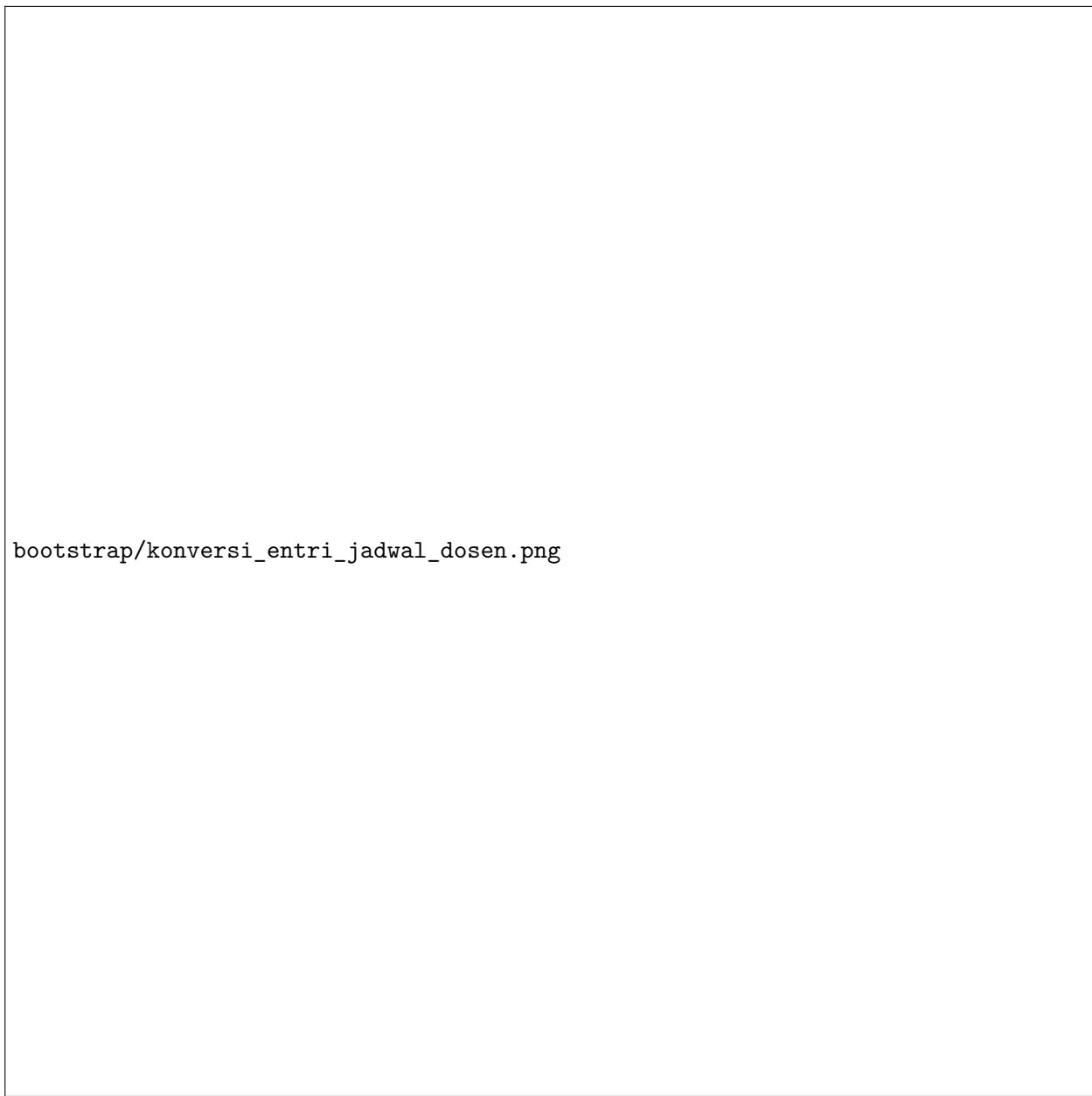
Gambar 4.17: Konversi Modal Hapus

Komponen Modal terdiri dari:

- `.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.
- `.alert button`: Membuat button pada tombol 'tolak' dan 'hapus'.

#### 4.1.12 Halaman Entri Jadwal Dosen

##### Halaman Utama



bootstrap/konversi\_entri\_jadwal\_dosen.png

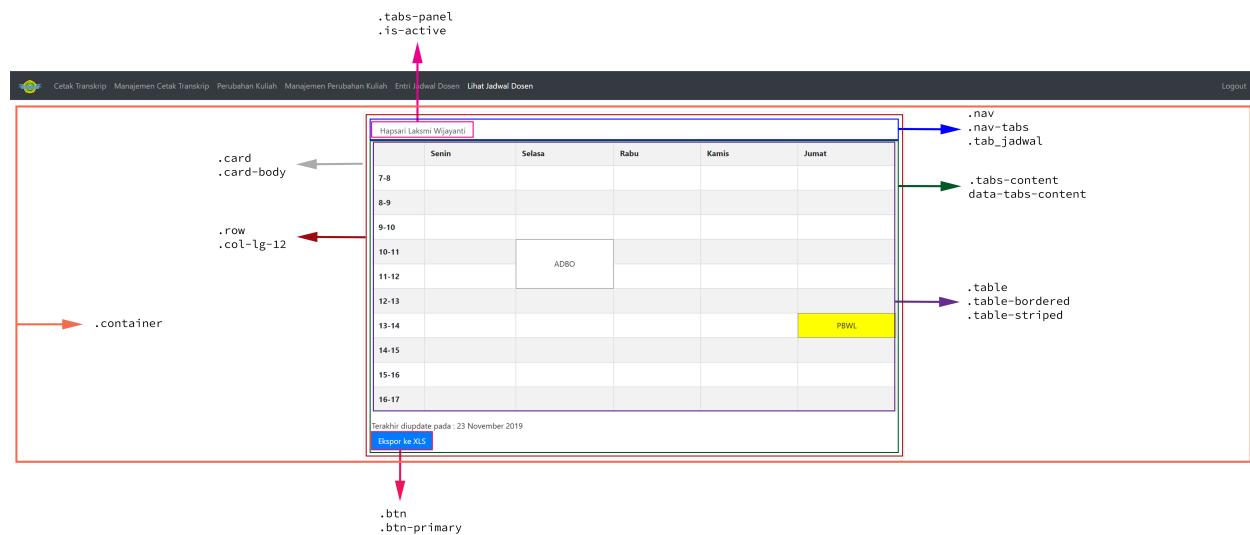
Gambar 4.18: Konversi Halaman Entri Jadwal Dosen

- **.row**: Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* menjadi satu baris.
- **.large-4 column**: Setiap field akan pada *form* Tambah Jadwal akan memiliki lebar masing-masing 4 grid pada layar *large*.
- **.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 ‘Ekspor ke XLS’ untuk konten Daftar Jadwal dan ‘Tambah’ pada konten Tambah Jadwal.
- **.alert button:** Membuat button pada tombol ‘Delete All’.

## Edit Jadwal

### 4.1.13 Halaman Lihat Jadwal Dosen



Gambar 4.19: Analisis Modal Edit Jadwal

- **.row:** Kelas ini memiliki dua fungsi sebagai container konten dan mengatur beberapa *field-form* 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 digerakan 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 judul tabs nama dosen.
- **.is-active:** Menunjukkan tabs nama dosen yang sedang dilihat





## BAB 5

### IMPLEMENTASI BOOTSTRAP 6 KE DALAM WEBSITE BLUETAPE

#### 5.1 Implementasi Tampilan Login

## **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.



# LAMPIRAN A

## KODE PROGRAM

Listing A.1: MyCode.c

```

1 // This does not make algorithmic sense,
2 // but it shows off significant programming characters.
3
4 #include<stdio.h>
5
6 void myFunction( int input, float* output ) {
7     switch ( array[i] ) {
8         case 1: // This is silly code
9             if ( a >= 0 || b <= 3 && c != x )
10                 *output += 0.005 + 20050;
11             *char = 'g';
12             b = 2^n + ~right_size - leftSize * MAX_SIZE;
13             c = (-aaa + &daa) / (bbb++ - ccc % 2 );
14             strcpy(a,"hello_@?");
15     }
16     count = ~mask | 0x00FF00AA;
17 }
18
19
20 // Fonts for Displaying Program Code in LATEX
21 // Adrian P. Robson, nepswb.co.uk
22 // 8 October 2012
23 // http://nepswb.co.uk/docs/progfonts.pdf

```

Listing A.2: MyCode.java

```

1 import java.util.ArrayList;
2 import java.util.Collections;
3 import java.util.HashSet;
4
5 //class for set of vertices close to furthest edge
6 public class MyFurSet {
7     protected int id;                                //id of the set
8     protected MyEdge FurthestEdge;                   //the furthest edge
9     protected HashSet<MyVertex> set;                //set of vertices close to furthest edge
10    protected ArrayList<ArrayList<Integer>> ordered; //list of all vertices in the set for each trajectory
11    protected ArrayList<Integer> closeID;           //store the ID of all vertices
12    protected ArrayList<Double> closeDist;          //store the distance of all vertices
13    protected int totaltrj;                          //total trajectories in the set
14
15    /*
16     * Constructor
17     * @param id : id of the set
18     * @param totaltrj : total number of trajectories in the set
19     * @param FurthestEdge : the furthest edge
20     */
21    public MyFurSet(int id,int totaltrj,MyEdge FurthestEdge) {
22        this.id = id;
23        this.totaltrj = totaltrj;
24        this.FurthestEdge = FurthestEdge;
25        set = new HashSet<MyVertex>();
26        ordered = new ArrayList<ArrayList<Integer>>();
27        for (int i=0;i<totaltrj;i++) ordered.add(new ArrayList<Integer>());
28        closeID = new ArrayList<Integer>(totaltrj);
29        closeDist = new ArrayList<Double>(totaltrj);
30        for (int i = 0;i <totaltrj;i++) {
31            closeID.add(-1);
32            closeDist.add(Double.MAX_VALUE);
33        }
34    }
35}
36

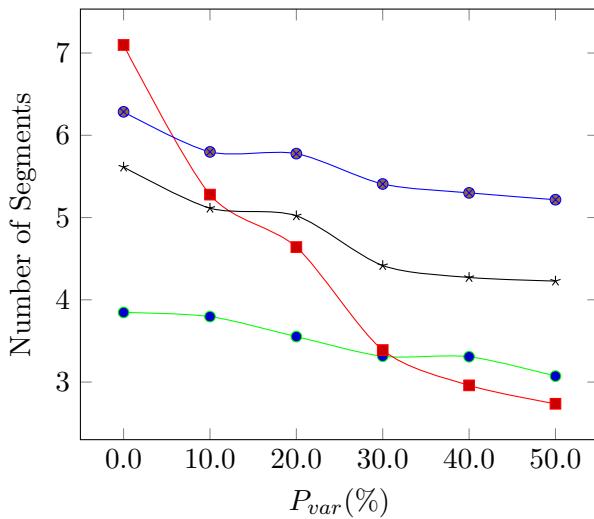
```



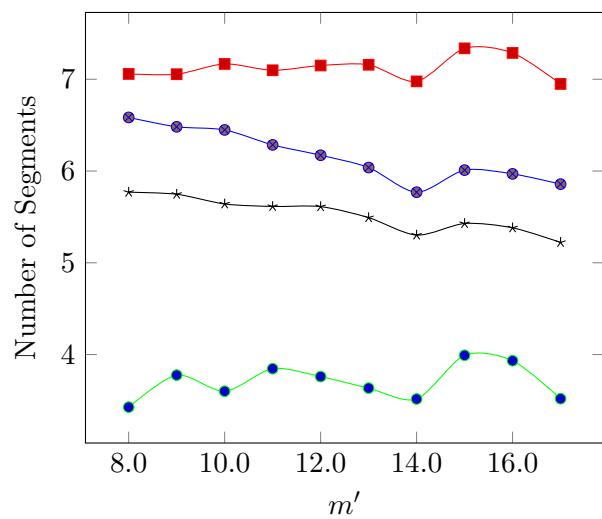
## LAMPIRAN B

### HASIL EKSPERIMENT

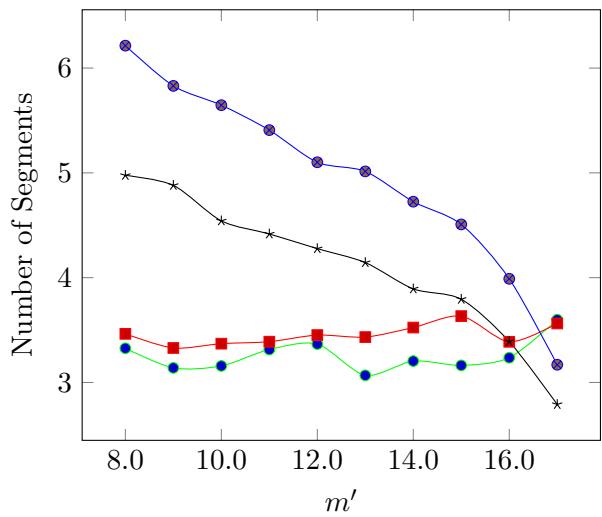
Hasil eksperimen berikut dibuat dengan menggunakan TIKZPICTURE (bukan hasil excel yg diubah ke file bitmap). Sangat berguna jika ingin menampilkan tabel (yang kuantitasnya sangat banyak) yang datanya dihasilkan dari program komputer.



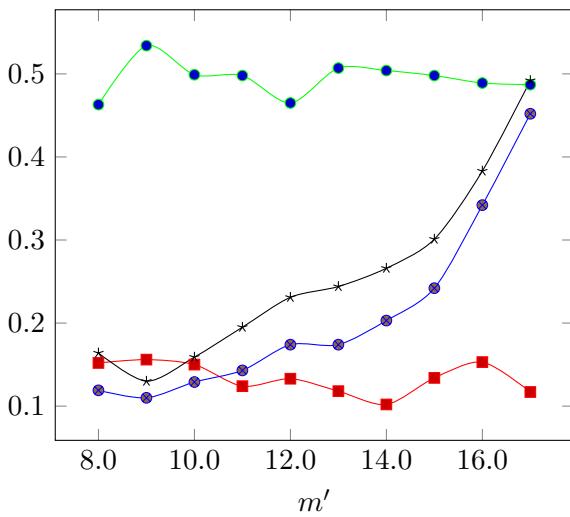
Gambar B.1: Hasil 1



Gambar B.2: Hasil 2



Gambar B.3: Hasil 3



Gambar B.4: Hasil 4