

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

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
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
2 LANDASAN TEORI	3
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 Foundation 6	10
2.3.1 Struktur File	10
2.3.2 Sistem Grid pada Foundation	11
3 ANALISIS TAMPILAN WEBSITE BLUETAPE	13
3.1 Menjalankan Framework Foundation	13
3.1.1 Folder untuk Menyimpan File Foundation 6 dan Plugin	13
3.1.2 Folder untuk Implementasi Framework Foundation 6 dan Plugin	13
3.2 Import File Foundation	14
3.3 Grid System Foundation 6 dalam Website BlueTape	14
3.3.1 Navigation Bar	15
3.3.2 Halaman Login	16
3.3.3 Halaman Cetak Transkrip	17
3.3.4 Halaman Manajemen Cetak Transkrip	19

3.3.5	Halaman Perubahan Kuliah	22
3.3.6	Halaman Manajemen Perubahan Kuliah	24
3.3.7	Halaman Entri Jadwal Dosen	28
3.3.8	Halaman Lihat Jadwal Dosen	30
DAFTAR REFERENSI		31
A KODE PROGRAM		33
B HASIL EKSPERIMEN		35

DAFTAR GAMBAR

2.1	Flow Chart Aplikasi CodeIgniter	6
2.2	Struktur File Zurb Foundation	10
2.3	Grid pada Zurb Foundation	12
3.1	Analisis Tampilan Login	15
3.2	Analisis Tampilan Login pada layar Medium dan Small	15
3.3	Analisis Tampilan Login	16
3.4	Analisis Halaman Cetak Transkrip	17
3.5	Analisis Modal Lihat	18
3.6	Analisis Manajemen Cetak Transkrip	19
3.7	Analisis Modal Lihat	20
3.8	Analisis Modal Print	20
3.9	Analisis Modal Tolak	21
3.10	Analisis Modal Hapus	21
3.11	Analisis Tampilan Perubahan Kuliah	22
3.12	Analisis Modal Lihat	23
3.13	Analisis Halaman Manajemen Perubahan Kuliah	24
3.14	Analisis Modal Lihat	25
3.15	Analisis Modal Setuju	26
3.16	Analisis Modal Tolak	26
3.17	Analisis Modal Hapus	27
3.18	Analisis Halaman Entri Jadwal Dosen	28
3.19	Analisis Modal Edit Jadwal	29
3.20	Analisis Modal Edit Jadwal	30
B.1	Hasil 1	35
B.2	Hasil 2	35
B.3	Hasil 3	35
B.4	Hasil 4	35

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.

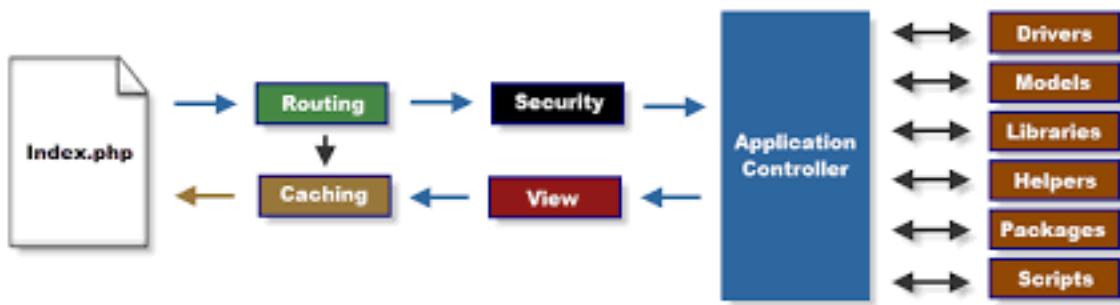
Beberapa tombol yang tersedia untuk setiap permohonan :

- 🔍 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 sebagaimana 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 'Hello World!';
    }

    public function comments()
    {
        echo 'Look at this!';
    }
}
```

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

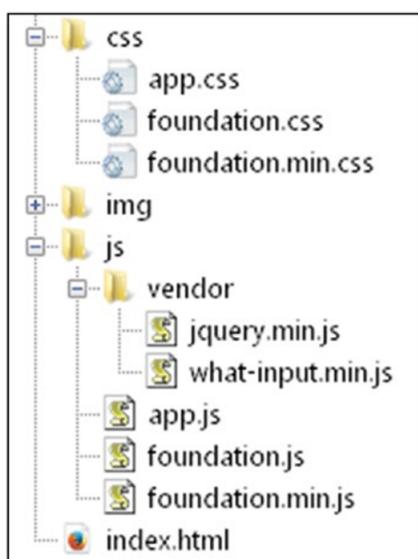
example . com / index . php / blog / index /

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

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

2.3 Foundation 6

2.3.1 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.

[1]

2.3.2 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 ditentukan kelasnya dengan tiga pilihan yaitu **.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

BAB 3

ANALISIS TAMPILAN WEBSITE BLUETAPE

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 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.

3.1.1 Folder untuk Menyimpan File Foundation 6 dan Plugin

Seluruh file terkait dengan Framework Foundation 6 diletakkan dalam folder **BlueTape2_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 UNPAR.
3. js : Menyimpan file javascript bawaan dari Foundation 6 dan file jquery yang tersimpan dalam folder vendor.
4. lib : Menyimpan plugin xdan-datetimepicker.

3.1.2 Folder untuk Implementasi Framework Foundation 6 dan Plugin

Untuk melihat code tampilan website, maka *user* akan mengakses file - file dalam folder **BlueTape/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

3.2 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>
```

3.3 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

3.3.1 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.

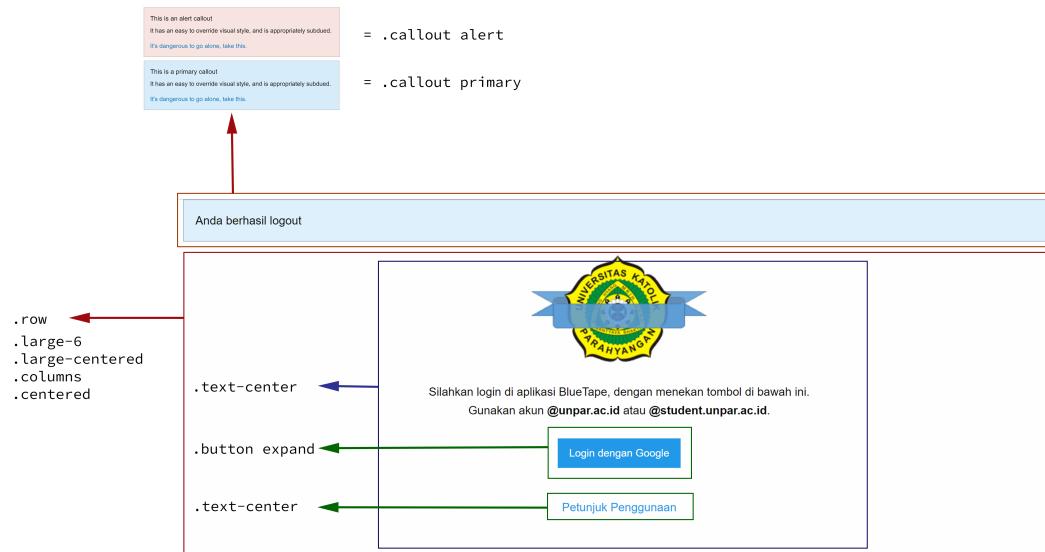
.menu-icon : Kelas untuk membuat icon menu.

.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.

3.3.2 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.

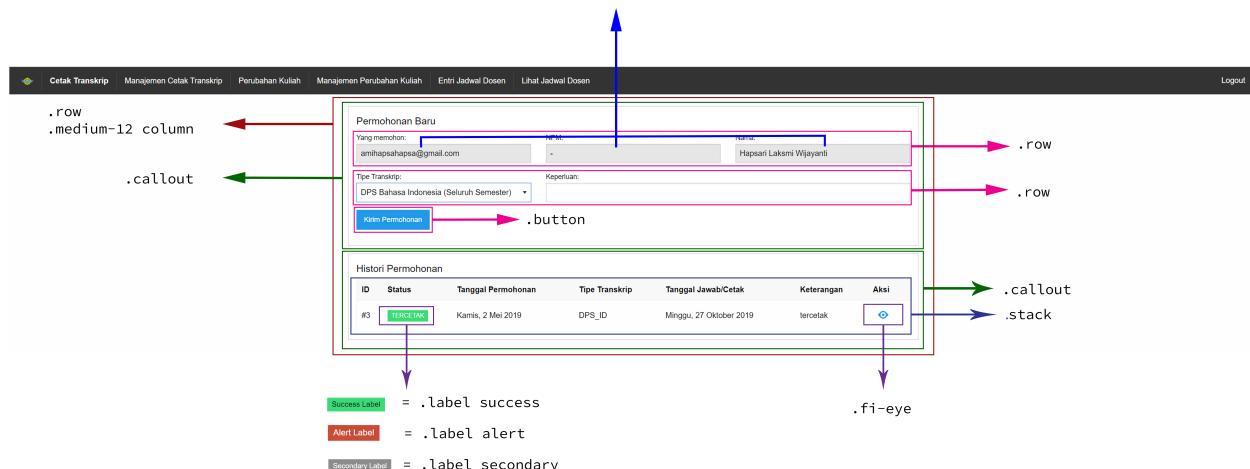
Callout : Terdapat dua jenis kelas yang dipakai

.callout alert : Notifikasi bahwa user harus login untuk mengakses website.

.callout primary : Notifikasi bahwa user berhasil logout.

3.3.3 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.
- **.fi-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 :

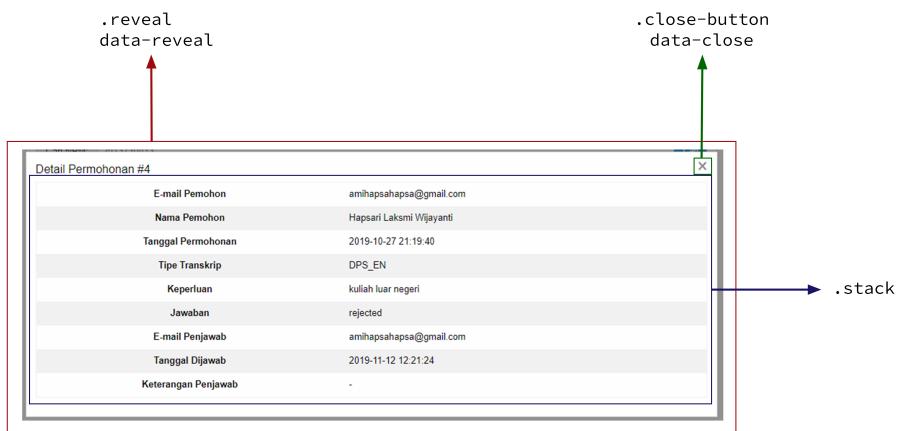
```
.label success : Label untuk transkrip yang telah tercetak  

.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 :

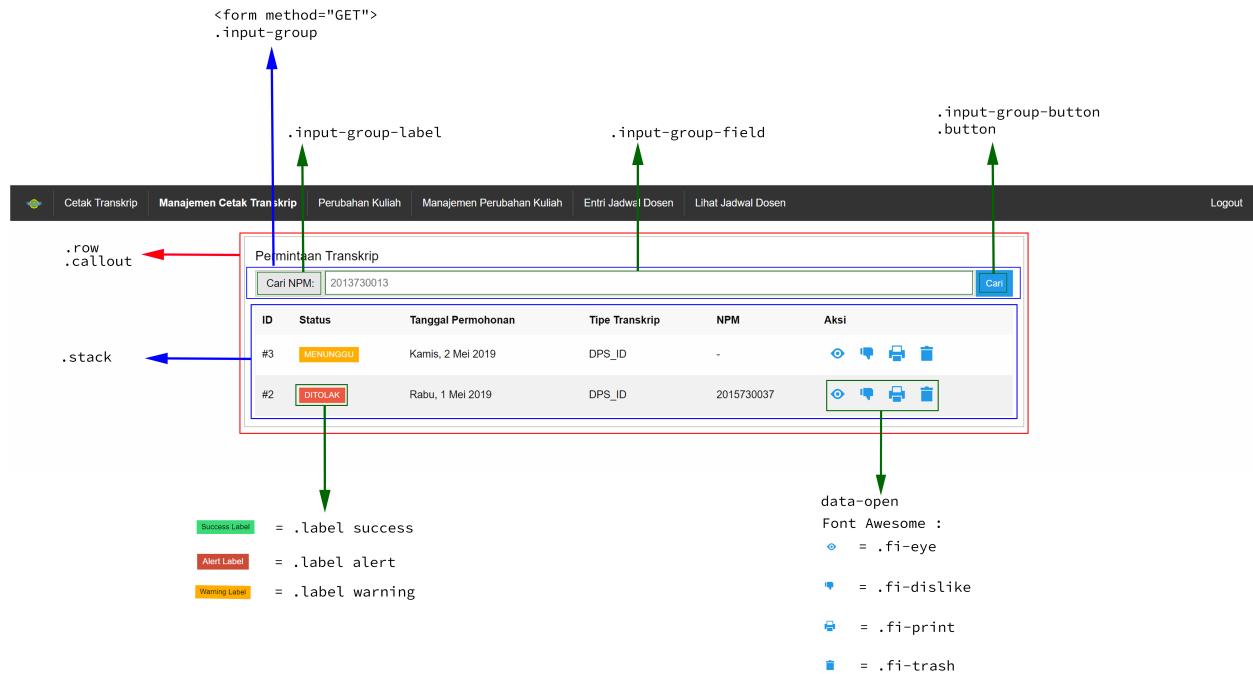
```
.reveal data-reveal : Membuat modal yang menampung tabel detail permohonan.  

.close-button data-close : Menutup modal yang telah terbuka.  

.stack : Membuat tabel detail permohonan.
```

3.3.4 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

.fi-eye : Ikon untuk menuju modal lihat transkrip.

.fi-dislike : Ikon untuk menuju modal tolak cetak transkrip.

.fi-print : Ikon untuk menuju modal cetak transkrip.

.fi-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

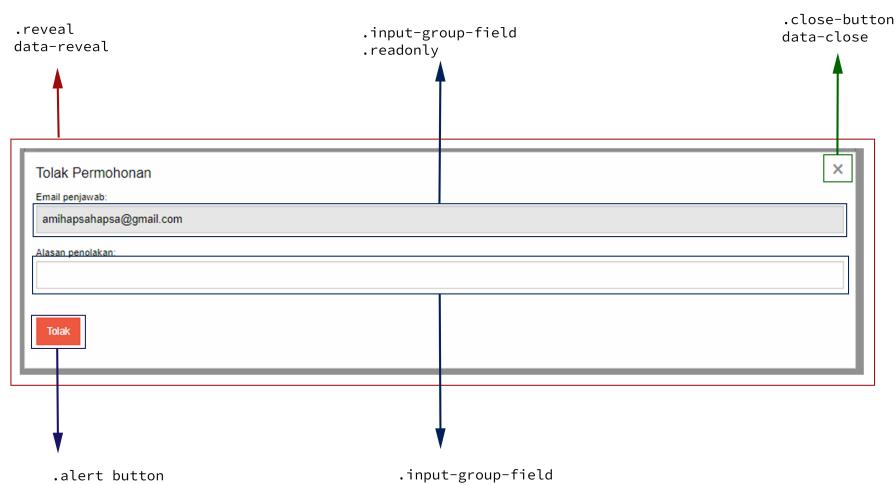
.label warning : Label untuk transkrip yang menunggu untuk tercetak



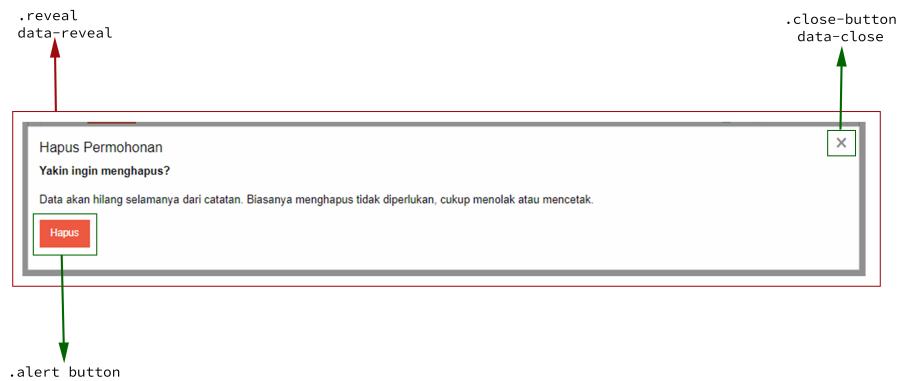
Gambar 3.7: Analisis Modal Lihat



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** : Menutup modal yang telah terbuka.

.**stack** : Membuat tabel detail permohonan.

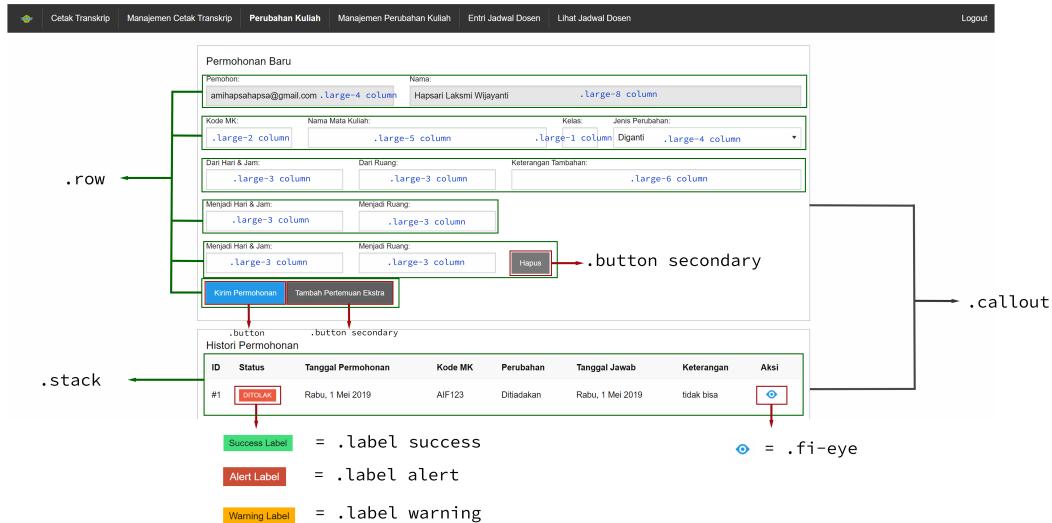
.**button** : Membuat button ‘Sudah dicetak’.

.**alert button** : Membuat button ‘Tolak’, ‘Hapus’.

.**alert button** : Membuat kolom input untuk keseluruhan form modal.

3.3.5 Halaman Perubahan Kuliah

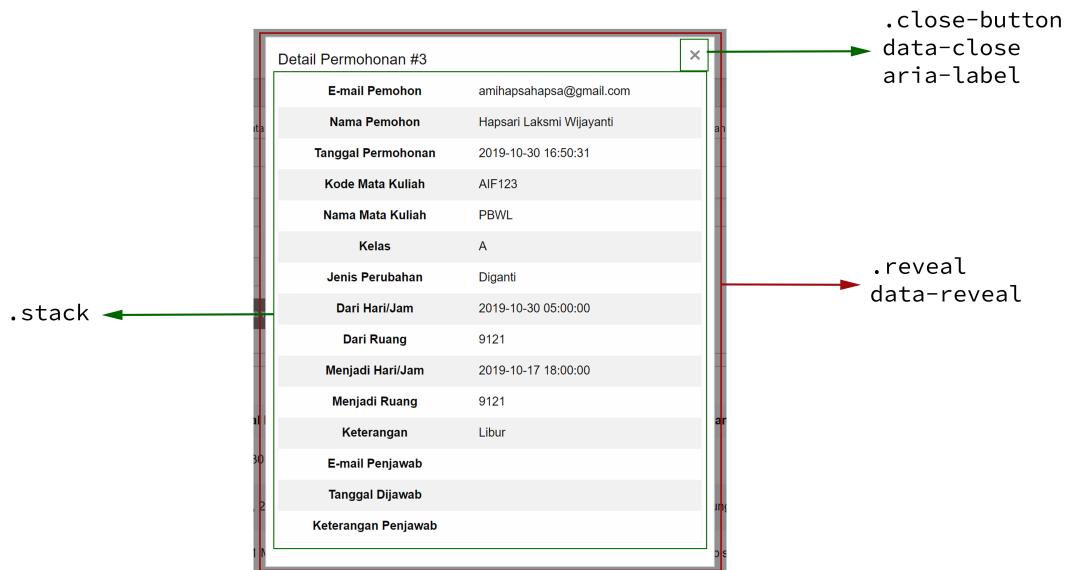
Halaman Utama



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
 - .fi-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.

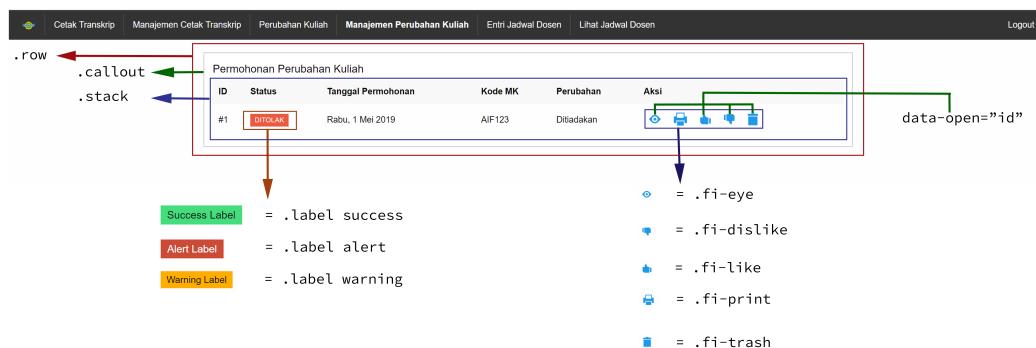
Modal



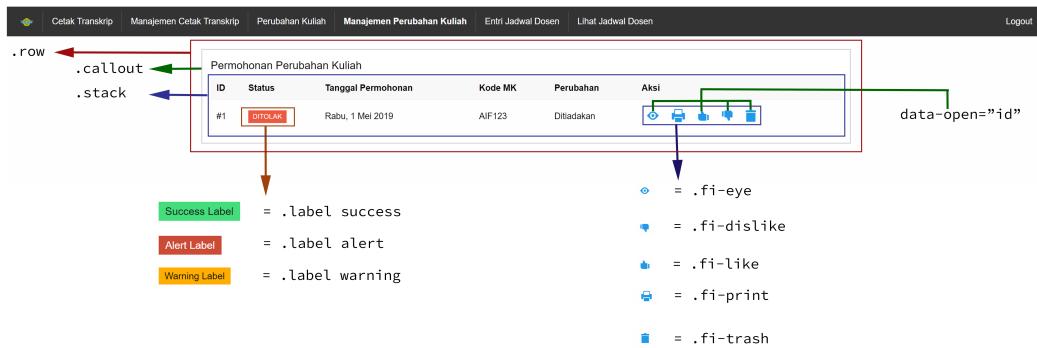
Gambar 3.12: Analisis Modal Lihat

3.3.6 Halaman Manajemen Perubahan Kuliah

3.3.7 Halaman Utama

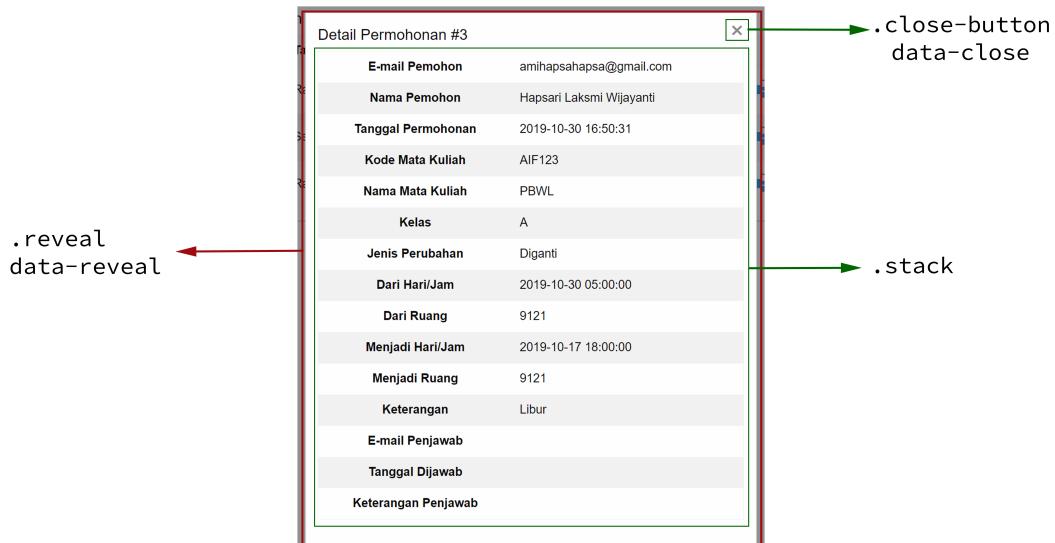


Gambar 3.13: Analisis Halaman Manajemen Perubahan Kuliah

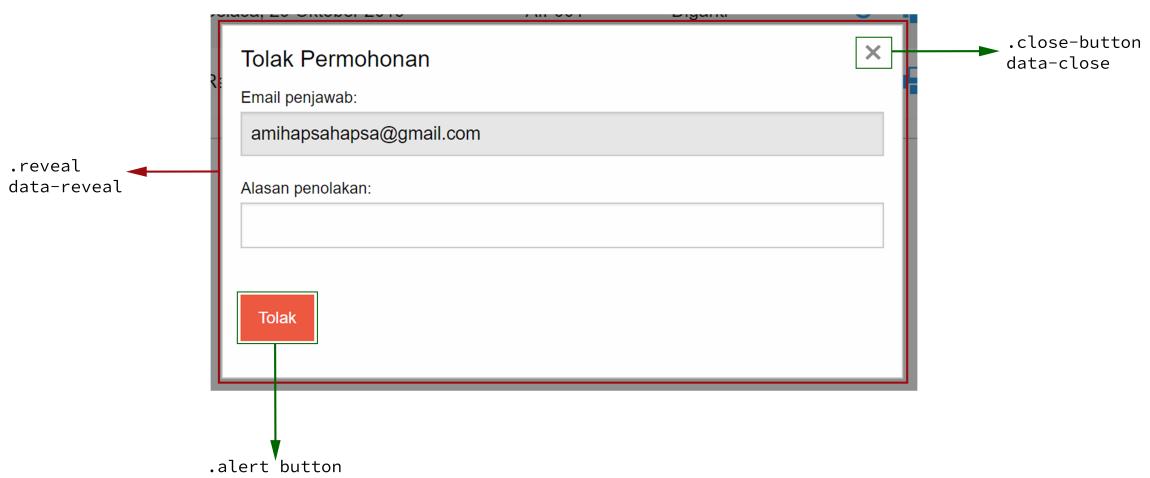


Gambar 3.14: Analisis Modal Lihat

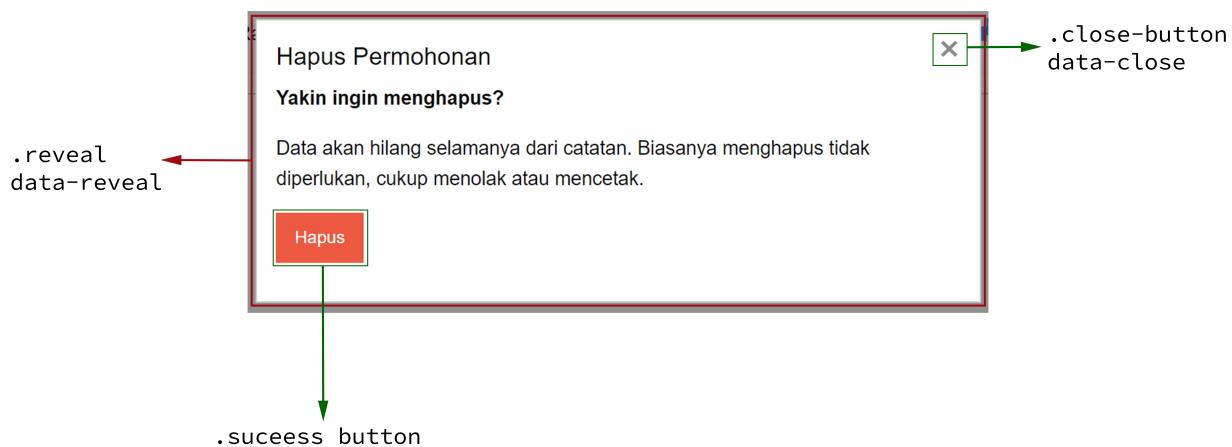
Tombol print akan menampilkan preview print jadwal pergantian mata kuliah.



Gambar 3.15: Analisis Modal Setuju

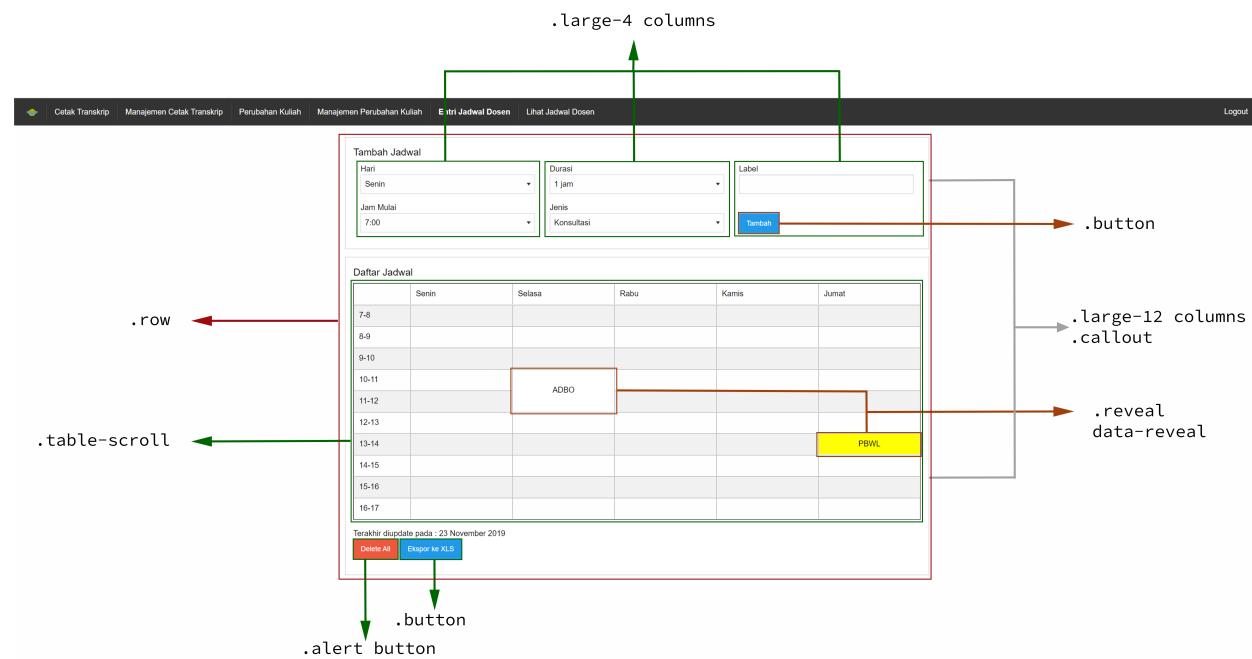


Gambar 3.16: Analisis Modal Tolak

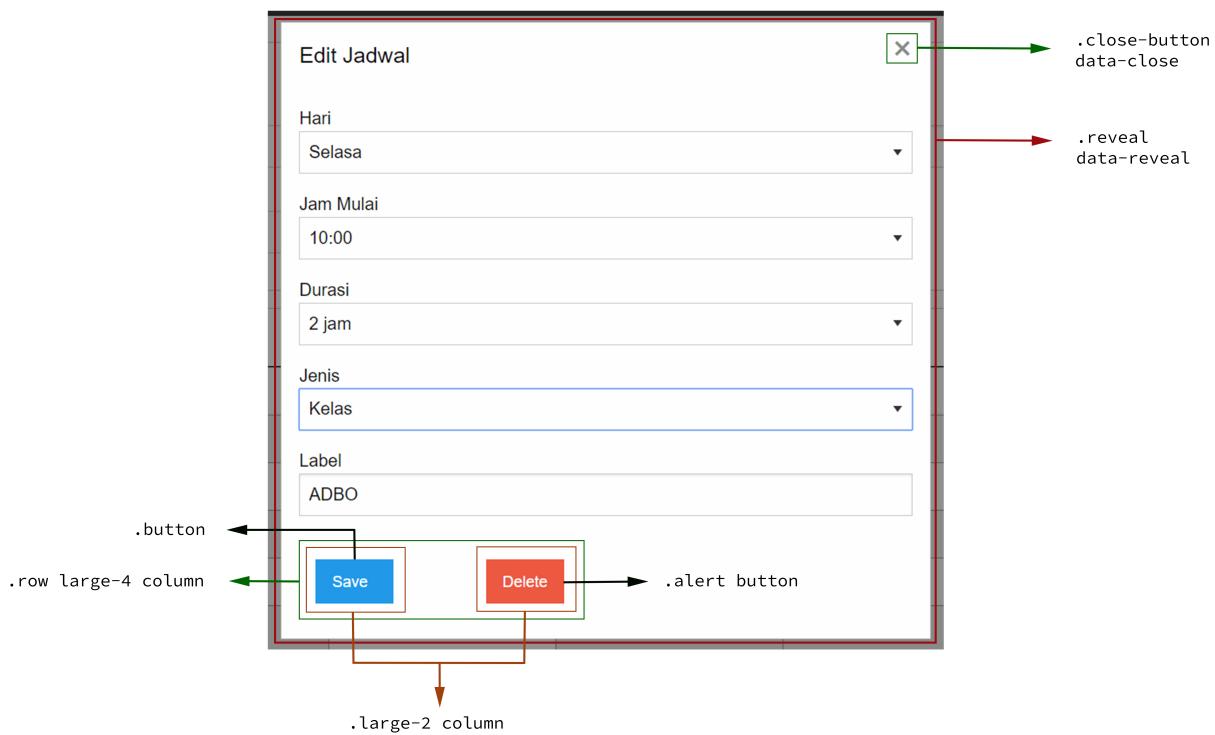


Gambar 3.17: Analisis Modal Hapus

3.3.8 Halaman Entri Jadwal Dosen

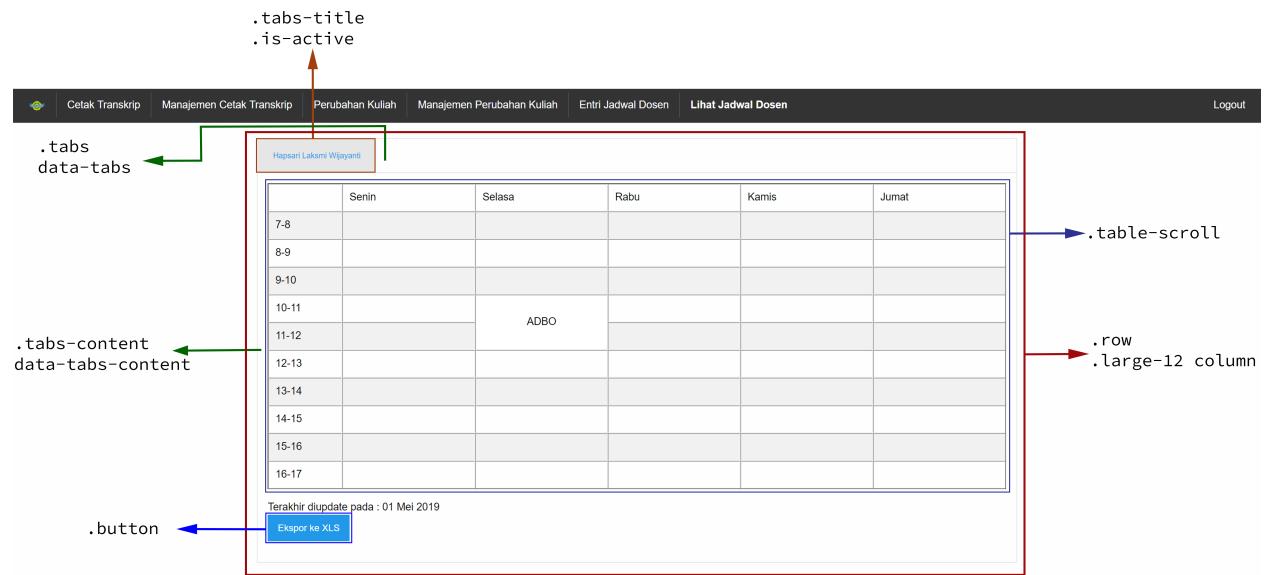


Gambar 3.18: Analisis Halaman Entri Jadwal Dosen



Gambar 3.19: Analisis Modal Edit Jadwal

3.3.9 Halaman Lihat Jadwal Dosen



Gambar 3.20: Analisis Modal Edit Jadwal

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[1] ) {
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 // Fonts for Displaying Program Code in LATEX
20 // Adrian P. Robson, nepswb.co.uk
21 // 8 October 2012
22 // http://nepswb.co.uk/docs/progfonts.pdf
23

```

Listing A.2: MyCode.java

```

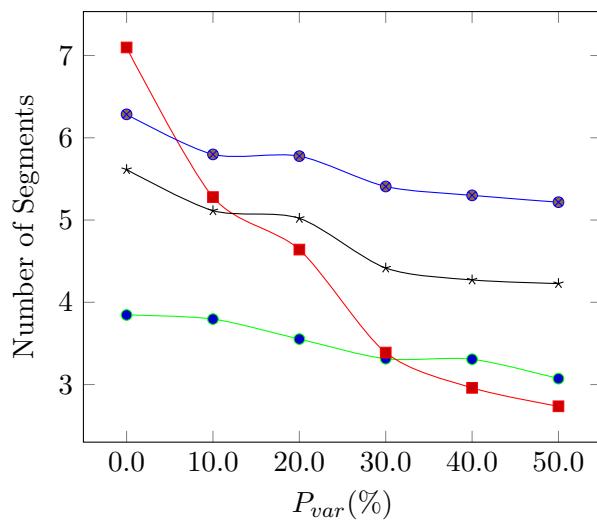
1 import java.util.ArrayList;
2 import java.util.Collections;
3 import java.util.HashSet;
4
5 //class for set of vertices close to furthest edge
6 public class MyFurSet {
7     protected int id;                                //id of the set
8     protected MyEdge FurthestEdge;                   //the 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

```

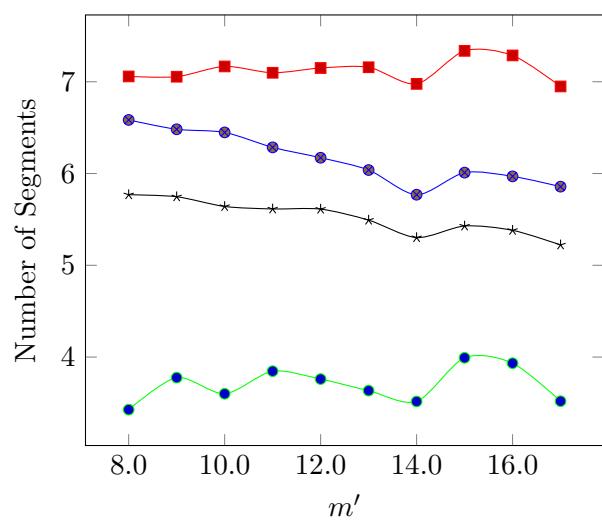

LAMPIRAN B

HASIL EKSPERIMENT

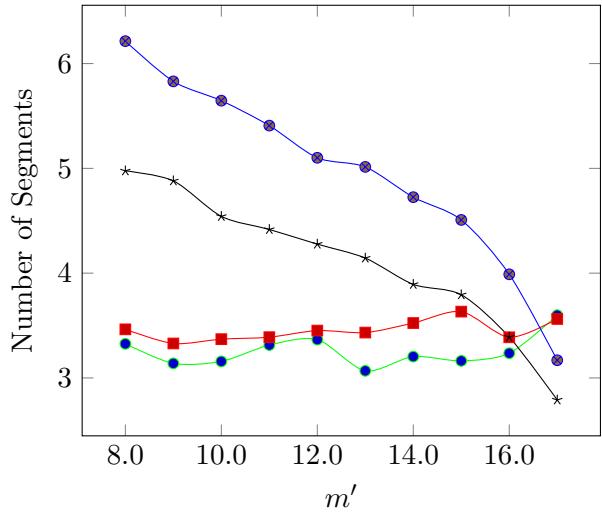
Hasil eksperimen berikut dibuat dengan menggunakan TIKZPICTURE (bukan hasil excel yg diubah ke file bitmap). Sangat berguna jika ingin menampilkan tabel (yang kuantitasnya sangat banyak) yang datanya dihasilkan dari program komputer.



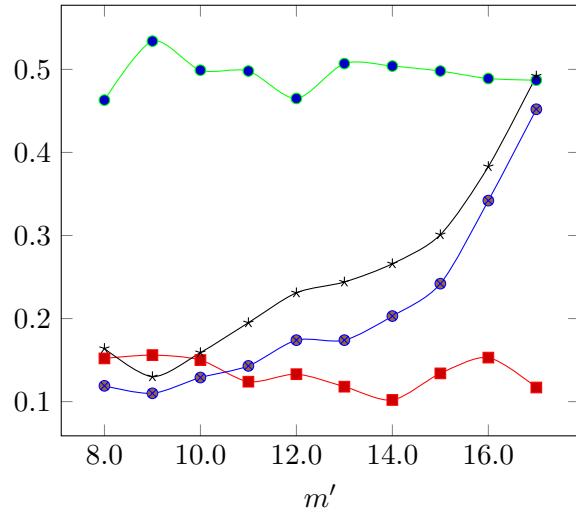
Gambar B.1: Hasil 1



Gambar B.2: Hasil 2



Gambar B.3: Hasil 3



Gambar B.4: Hasil 4