

MIGRASI ZURB FOUNDATION KE BOOTSTRAP 4

HAPSARI LAKSMI W-2015730037

1 Data Skripsi

Pembimbing utama/tunggal: **Pascal Alfadian .N**

Pembimbing pendamping: -

Kode Topik : **PAN4791**

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

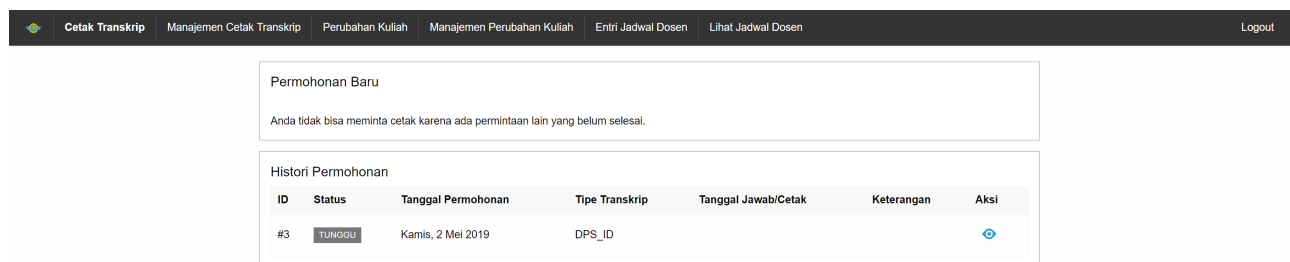
Pengambilan pertama kali topik ini pada : Semester **47 - Ganjil 19/20**

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

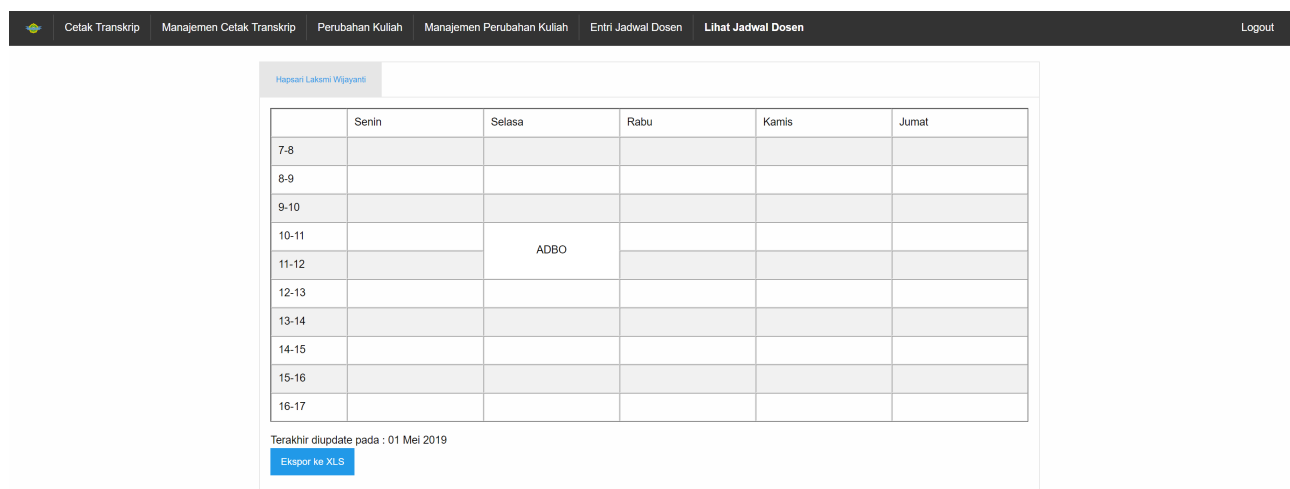
Tipe Laporan : **B -** Dokumen untuk reviewer pada presentasi dan **review Skripsi 1**

2 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 *Zurb Foundation*.



Gambar 1: Tampilan Cetak Transkrip



Gambar 2: Tampilan Lihat Jadwal Dosen

The screenshot shows the 'Manajemen Cetak Transkrip' (Transcript Management) page. At the top, there is a search bar for 'Cari NPM:' with the value '2013730013' and a 'Cari' button. Below this is a table with the following data:

ID	Status	Tanggal Permohonan	Tipe Transkrip	NPM	Aksi
#3	MENUNGGU	Kamis, 2 Mei 2019	DPS_ID	-	View Edit Print Delete
#2	DITOLAK	Rabu, 1 Mei 2019	DPS_ID	2015730037	View Edit Print Delete

Gambar 3: Tampilan-Manajemen-Cetak-Transkrip

The screenshot shows the 'Tambah Jadwal' (Add Schedule) page. It includes a form with the following fields:

- Hari:
- Durasi:
- Label:
- Jam Mulai:
- Jenis:
- Tambah button

Below the form is a 'Daftar Jadwal' (Schedule List) table with columns for days of the week (Senin, Selasa, Rabu, Kamis, Jumat) and time slots (7-8, 8-9, 9-10, 10-11, 11-12, 12-13, 13-14, 14-15, 15-16, 16-17). The table shows a schedule entry for 'ADBO' on Tuesday (Selasa) from 10-11 to 11-12.

At the bottom, there is a 'Terakhir diupdate pada : 01 Mei 2019' message and buttons for 'Delete All' and 'Ekspor ke XLS'.

Gambar 4: Tampilan Entri Jadwal Dosen

The screenshot shows the 'Permohonan Baru' (New Request) page. It includes a form with the following fields:

- Pemohon:
- Nama:
- Kode MK:
- Nama Mata Kuliah:
- Kelas:
- Jenis Perubahan:
- Dari Hari & Jam:
- Dari Ruang:
- Keterangan Tambahan:
- Menjadi Hari & Jam:
- Menjadi Ruang:
- Menjadi Hari & Jam:
- Menjadi Ruang:
- Hapus button
- Kirim Permohonan button
- Tambah Pertemuan Ekstra button

Below the form is a 'Histori Permohonan' (Request History) table with the following data:

ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Tanggal Jawab	Keterangan	Aksi
#1	DITOLAK	Rabu, 1 Mei 2019	AIF123	Ditiadakan	Rabu, 1 Mei 2019	tidak bisa	View

Gambar 5: Tampilan Perubahan Kuliah

Meskipun *open-source*, saat ini *Zurb Foundation* tidak sepopuler *framework Bootstrap*. *Bootstrap* adalah *Javascript framework* yang didesain untuk membantu membangun komponen *user interface* yang terdiri dari *CSS*, *JavaScript/jQuery*, dan *glyphicons*. Pembangunan *website* yang lebih cepat dan besarnya komunitas yang ada berdampak pada banyaknya pengembang *web* yang memanfaatkan *framework Bootstrap*. Sehingga jumlah proyek yang dihasilkan oleh *framework Bootstrap* lebih banyak dibanding *Zurb Foundation*.

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

Permohonan Perubahan Kuliah					
ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Aksi
#1	DITOLAK	Rabu, 1 Mei 2019	AIF123	Ditiadakan	Detail Print Like Comment Delete

Gambar 6: Tampilan Manajemen Perubahan Kuliah

3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini :

1. Membuat template cetak transkrip nilai, template manajemen cetak transkrip, template perubahan kuliah, module manajemen perubahan kuliah, modul entri jadwal dosen dan module lihat jadwal dosen dengan framework Bootstrap 4 yang responsive untuk berbagai platform.
2. Mengimplentasikan plugin yang tersedia dalam library Bootstrap 4.

4 Rumusan Masalah

1. Bagaimana membuat template manajemen cetak transkrip, manajemen perubahan kuliah dan manajemen jadwal dosen.
2. Bagaimana mengimplentasikan plugin yang tersedia di dalam Bootstrap 4.

5 Detail Perkembangan Pengerjaan Skripsi

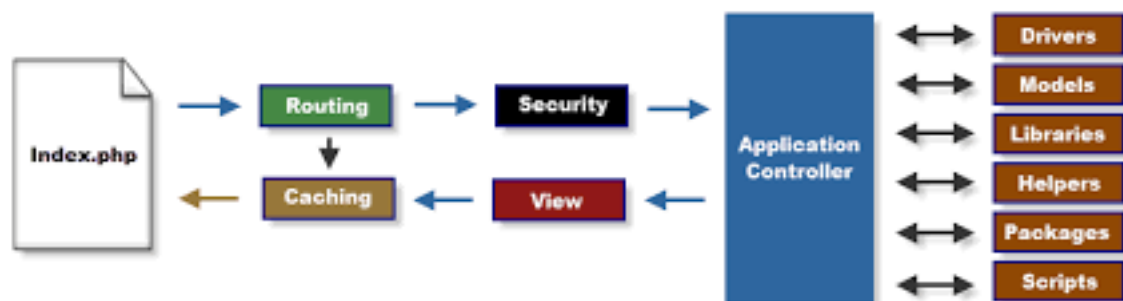
Detail bagian pekerjaan skripsi sesuai dengan rencan kerja/laporan perkembangan terakhir :

1. **Mempelajari framework PHP Codeigniter.**

Status : Ada sejak rencana kerja skripsi.

Hasil : Mempelajari konsep codeigniter dan mencoba langsung didalam proyek BlueTape

Application Flow Chart Gambar berikut mengilustrasikan bagaimana alur data pada sistem :



Gambar 7: Flow Chart Aplikasi CodeIgniter

- (a) **Index.php** : bertindak sebagai *front controller*, menginisiasi *base resources* yang dibutuhkan untuk menjalankan CodeIgniter.
- (b) **Router** : akan memeriksa permintaan HTTP untuk menetapkan hal apa yang harus dilakukan dengan permintaan tersebut.

- (c) **Cache** : Apabila terdapat *cache*, maka *cache* tersebut akan dikirimkan langsung ke browser, dengan melewati sistem eksekusi normal.
- (d) **Security** : Sebelum *controller* dimuat, *HTTP request* dan *user* mana pun yang mengirimkan data diseleksi dahulu untuk keamanan.
- (e) **Controller** : Terdiri dari *model*, *core libraries*, *helpers*, dan *resources* yang dibutuhkan untuk proses *request* tertentu.
- (f) **View** : Tampilan yang telah selesai dirender kemudian dikirim ke *web browser* untuk dilihat. Jika *caching* diaktifkan, tampilan di *cache* terlebih dahulu sehingga pada permintaan selanjutnya dapat dilayani.[?]

CodeIgniter URLs Codeigniter menggunakan pendekatan berbasis segment :

```
1 blackexample.blackcom/blackclass/blackfunction/blackID
```

- (a) Segmen pertama menyatakan kelas *controller* yang harus dipanggil.
- (b) Segmen kedua menyatakan fungsi kelas, atau metode, yang harus dipanggil.
- (c) Segmen ketiga dan setiap segmen setelahnya menyatakan ID dan variabel apa pun yang akan diteruskan ke controller.

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 :

```
1 <?blackphp
2 blackclass blackModel_name blackextends blackCI_Model {
3
4 }
```

```
1 blackapplication/blackmodels/blackUser_model.blackphp
```

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

```
1 black$this->blackload->blackmodel('blackmodel_name');
```

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

```
1 black$config['blackhostname'] = 'blacklocalhost';
2 black$config['blackusername'] = 'blackmyusername';
3 black$config['blackpassword'] = 'blackmypassword';
4 black$config['blackdatabase'] = 'blackmydatabase';
5 black$config['blackdbdriver'] = 'blackmysql';
6 black$config['blackdbprefix'] = '';
7 black$config['blackpconnect'] = blackFALSE;
8 black$config['blackdb_debug'] = blackTRUE;
9
10 black$this->blackload->blackmodel('blackmodel_name', '', black$config);
```

listing only

View *View* adalah informasi yang sedang dilihat oleh *user*. Sebuah *View* normalnya menjadi sebuah halaman web, namun dalam CodeIgniter, sebuah *view* dapat menjadi sebuah *page fragment* seperti *header* atau *footer*. Dapat juga menjadi halaman RSS, atau tipe apapun dari "page".

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:

```

1      <blackhtml>
2      <blackhead>
3      <blacktitle>blackMy blackBlog</blacktitle>
4      </blackhead>
5      <blackbody>
6      <blackh1>blackWelcome blackto blackmy blackBlog!</blackh1>
7      </blackbody>
8      </blackhtml>

```

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

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

```

1      black$this->blackload->blackview('blackname');

```

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

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

```

1      black$this->blackload->blackview()

```

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

```

1      <?blackphp
2
3      blackclass blackPage blackextends blackCI_Controller {
4
5      blackpublic blackfunction blackindex()
6      {
7      black$data['blackpage_title'] = 'blackYour blacktitle';
8      black$this->blackload->blackview('blackheader');
9      black$this->blackload->blackview('blackmenu');
10     black$this->blackload->blackview('blackcontent', black$data);
11     black$this->blackload->blackview('blackfooter');
12     }
13
14     }

```

Menyimpan Views didalam Sub Direktori Untuk menyimpan didalam sub direktori maka dapat menyertakan nama direktori yang memuat *view*.

```

1      black$this->blackload->blackview('blackdirectory_name/blackfile_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:

```

1      black$data = blackarray(
2      'blacktitle' => 'blackMy blackTitle',
3      'blackheading' => 'blackMy blackHeading',
4      'blackmessage' => 'blackMy blackMessage'
5      );
6
7      black$this->blackload->blackview('blackblogview', black$data);

```

Kemudian, penggunaan dengan objek:

```
1 black$data = blacknew blackSomeclass();
2 black$this->blackload->blackview('blackblogview', black$data);
```

Sehingga apabila dimasukan ke controller, kode yang ditambahkan adalah:

```
1 <?blackphp
2 blackclass blackBlog blackextends blackCI.Controller {
3
4     blackpublic blackfunction blackindex()
5     {
6         black$data['blacktitle'] = "blackMy blackReal blackTitle";
7         black$data['blackheading'] = "blackMy blackReal blackHeading";
8
9         black$this->blackload->blackview('blackblogview', black$data);
10    }
11 }
```

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

```
1 <blackhtml>
2 <blackhead>
3 <blacktitle><?blackphp blackecho black$title;?></blacktitle>
4 </blackhead>
5 <blackbody>
6 <blackh1><?blackphp blackecho black$heading;?></blackh1>
7 </blackbody>
8 </blackhtml>
```

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:

```
1 <?blackphp
2 blackexample.blackcom/blackindex.blackphp/blackblog/
```

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.

```
1 <?blackphp
2 blackclass blackBlog blackextends blackCI.Controller {
3
4     blackpublic blackfunction blackindex()
5     {
6         blackecho 'blackHello blackWorld'
7     }
8 }
```

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()*.

```
1 <?blackphp
2 blackclass blackBlog blackextends blackCI.Controller {
3
4     blackpublic blackfunction blackindex()
5     {
6         blackecho 'blackHello blackWorld!';
7     }
8
9     blackpublic blackfunction blackcomments()
10    {
11        blackecho 'blackLook blackat blackthis!';
12    }
13 }
```

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

```
1 blackexample.blackcom/blackindex.blackphp/blackblog/blackindex/
```

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

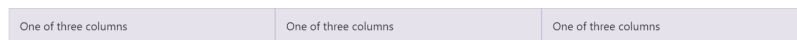
```
1 blackexample.blackcom/blackindex.blackphp/blackblog/blackcomments/
```

2. Mempelajari framework front-end Bootstrap 4 beserta plugin-plugin yang tersedia.

Status : Ada sejak rencana kerja skripsi.

Hasil : Sudah dicoba pada sebagian besar proyek BlurTape.

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*. [?]



Gambar 8: Grid pada Bootstrap

```
1 <blackdiv blackclass="blackcontainer">
2 <blackdiv blackclass="blackrow">
3 <blackdiv blackclass="blackcol-blacksm">
4 blackOne blackof blackthree blackcolumns
5 </blackdiv>
6 <blackdiv blackclass="blackcol-blacksm">
7 blackOne blackof blackthree blackcolumns
8 </blackdiv>
9 <blackdiv blackclass="blackcol-blacksm">
10 blackOne blackof blackthree blackcolumns
11 </blackdiv>
12 </blackdiv>
13 </blackdiv>
```

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 `.col-sm` akan secara otomatis membuat lebar kolom sebesar 25%.
- Kelas kolom menunjukkan jumlah kolom yang ingin digunakan, dengan maksimal 12 kolom per baris. Apabila *developer* menginginkan tiga kolom yang memiliki lebar yang sama maka dapat menggunakan `.col-4`.
- Lebar kolom diatur dalam persentase, sehingga kolom akan memiliki lebar yang berubah-ubah dan ukuran bergantung dengan elemen *parent* nya.

Pilihan Grid Bootstrap menggunakan px untuk grid breakpoint dan lebar container. Ini dikarenakan lebar *viewport* ditentukan dengan satuan pixels. Berikut ini tabel 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	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
# of columns	12				
Gutter width	30px (15px on each side of a column)				
Nestable	Yes				
Column ordering	Yes				

Gambar 9: Pilihan kelas grid pada Bootstrap

5.1 Konten

5.1.1 Tabel

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 10: Tabel default pada Bootstrap

```

1  <blacktable blackclass="blacktable">
2  <blackthead>
3  <blacktr>
4  <blackth blackscope="blackcol">#</blackth>
5  <blackth blackscope="blackcol">blackFirst</blackth>
6  <blackth blackscope="blackcol">blackLast</blackth>
7  <blackth blackscope="blackcol">blackHandle</blackth>
8  </blacktr>
9  </blackthead>
10 <blacktbody>
11 <blacktr>
12 <blackth blackscope="blackrow">1</blackth>
13 <blacktd>blackMark</blacktd>
14 <blacktd>blackOtto</blacktd>
15 <blacktd>black@mdo</blacktd>
16 </blacktr>
17 <blacktr>
18 <blackth blackscope="blackrow">2</blackth>
19 <blacktd>blackJacob</blacktd>
20 <blacktd>blackThornton</blacktd>
21 <blacktd>black@fat</blacktd>
22 </blacktr>
23 <blacktr>
24 <blackth blackscope="blackrow">3</blackth>
25 <blacktd>blackLarry</blacktd>
26 <blacktd>blackthe blackBird</blacktd>
27 <blacktd>black@twitter</blacktd>
28 </blacktr>
29 </blacktbody>
30 </blacktable>

```

Opsi Tabel dengan *head* yang dimodifikasi Penggunaan kelas `.thead-light` atau `.thead-dark` dapat digunakan untuk membuat `<thead>` menjadi abu muda atau abu tua.

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

Gambar 11: Tabel dengan thead yang dimodifikasi pada Bootstrap

```

1  <blacktable blackclass="blacktable">
2  <blackthead blackclass="blackthead-blackdark">
3  <blacktr>
4  <blackth blackscope="blackcol">#</blackth>
5  <blackth blackscope="blackcol">blackFirst</blackth>
6  <blackth blackscope="blackcol">blackLast</blackth>
7  <blackth blackscope="blackcol">blackHandle</blackth>
8  </blacktr>
9  </blackthead>
10 <blacktbody>
11 <blacktr>
12 <blackth blackscope="blackrow">1</blackth>
13 <blacktd>blackMark</blacktd>
14 <blacktd>blackOtto</blacktd>
15 <blacktd>black@mdo</blacktd>
16 </blacktr>
17 <blacktr>
18 <blackth blackscope="blackrow">2</blackth>
19 <blacktd>blackJacob</blacktd>
20 <blacktd>blackThornton</blacktd>
21 <blacktd>black@fat</blacktd>
22 </blacktr>
23 <blacktr>
24 <blackth blackscope="blackrow">3</blackth>
25 <blacktd>blackLarry</blacktd>
26 <blacktd>blackthe blackBird</blacktd>
27 <blacktd>black@twitter</blacktd>
28 </blacktr>
29 </blacktbody>
30 </blacktable>
31
32 <blacktable blackclass="blacktable">
33 <blackthead blackclass="blackthead-blacklight">
34 <blacktr>
35 <blackth blackscope="blackcol">#</blackth>
36 <blackth blackscope="blackcol">blackFirst</blackth>
37 <blackth blackscope="blackcol">blackLast</blackth>
38 <blackth blackscope="blackcol">blackHandle</blackth>
39 </blacktr>
40 </blackthead>
41 <blacktbody>
42 <blacktr>
43 <blackth blackscope="blackrow">1</blackth>
44 <blacktd>blackMark</blacktd>
45 <blacktd>blackOtto</blacktd>
46 <blacktd>black@mdo</blacktd>
47 </blacktr>
48 <blacktr>
49 <blackth blackscope="blackrow">2</blackth>
50 <blacktd>blackJacob</blacktd>
51 <blacktd>blackThornton</blacktd>
52 <blacktd>black@fat</blacktd>
53 </blacktr>
54 <blacktr>
55 <blackth blackscope="blackrow">3</blackth>
56 <blacktd>blackLarry</blacktd>
57 <blacktd>blackthe blackBird</blacktd>
58 <blacktd>black@twitter</blacktd>
59 </blacktr>
60 </blacktbody>
61 </blacktable>

```

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 12: Menyelaraskan gambar ke kanan dan kiri pada bootstrap

```
1 <blackimg blacksrc="..." blackclass="blackrounded blackfloat-blackleft" blackalt="...">
2 <blackimg blacksrc="..." blackclass="blackrounded blackfloat-blackright" blackalt="...">
```

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.

5.1.2 Form Controls

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

```
1 <blackform>
2 <blackdiv blackclass="blackform-blackgroup">
3 <blacklabel blackfor="blackexampleFormControlInput1">blackEmail blackaddress </blacklabel>
4 <blackinput blacktype="blackemail" blackclass="blackform-blackcontrol" blackid="blackexampleFormControlInput1"
   blackplaceholder="blackname@example.blackcom">
5 </blackdiv>
6 <blackdiv blackclass="blackform-blackgroup">
7 <blacklabel blackfor="blackexampleFormControlSelect1">blackExample blackselect </blacklabel>
8 <blackselect blackclass="blackform-blackcontrol" blackid="blackexampleFormControlSelect1">
9 <blackoption>1</blackoption>
10 <blackoption>2</blackoption>
11 <blackoption>3</blackoption>
12 <blackoption>4</blackoption>
13 <blackoption>5</blackoption>
14 </blackselect>
15 </blackdiv>
16 <blackdiv blackclass="blackform-blackgroup">
17 <blacklabel blackfor="blackexampleFormControlSelect2">blackExample blackmultiple blackselect </blacklabel>
18 <blackselect blackmultiple blackclass="blackform-blackcontrol" blackid="blackexampleFormControlSelect2">
19 <blackoption>1</blackoption>
20 <blackoption>2</blackoption>
21 <blackoption>3</blackoption>
22 <blackoption>4</blackoption>
23 <blackoption>5</blackoption>
24 </blackselect>
25 </blackdiv>
26 <blackdiv blackclass="blackform-blackgroup">
27 <blacklabel blackfor="blackexampleFormControlTextarea1">blackExample blacktextarea </blacklabel>
28 <blacktextarea blackclass="blackform-blackcontrol" blackid="blackexampleFormControlTextarea1" blackrows
   = "3"></blacktextarea>
29 </blackdiv>
30 </blackform>
```

Gambar 13: Forms Basic pada Bootstrap

5.1.3 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 14: Forms Basic pada Bootstrap

```

1  <blackform>
2  <blackdiv blackclass="blackform-blackrow">
3  <blackdiv blackclass="blackcol-7">
4  <blackinput blacktype="blacktext" blackclass="blackform-blackcontrol" blackplaceholder="blackCity">
5  </blackdiv>
6  <blackdiv blackclass="blackcol">
7  <blackinput blacktype="blacktext" blackclass="blackform-blackcontrol" blackplaceholder="blackState">
8  </blackdiv>
9  <blackdiv blackclass="blackcol">
10 <blackinput blacktype="blacktext" blackclass="blackform-blackcontrol" blackplaceholder="blackZip">
11 </blackdiv>
12 </blackdiv>
13 </blackform>

```

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 15: Disabled Basic pada Bootstrap

```

1  <blackform>
2  <blackfieldset blackdisabled>
3  <blackdiv blackclass="blackform-blackgroup">
4  <blacklabel blackfor="blackdisabledTextInput">blackDisabled blackinput</blacklabel>
5  <blackinput blacktype="blacktext" blackid="blackdisabledTextInput" blackclass="blackform-blackcontrol"
6  blackplaceholder="blackDisabled blackinput">
7  </blackdiv>
8  <blackdiv blackclass="blackform-blackgroup">
9  <blacklabel blackfor="blackdisabledSelect">blackDisabled blackselect blackmenu</blacklabel>
10 <blackselect blackid="blackdisabledSelect" blackclass="blackform-blackcontrol">
11 <blackoption>blackDisabled blackselect</blackoption>
12 </blackselect>
13 </blackdiv>
14 <blackdiv blackclass="blackform-blackcheck">
15 <blackinput blackclass="blackform-blackcheck-blackinput" blacktype="blackcheckbox" blackid="
16 blackdisabledFieldsetCheck" blackdisabled>
17 <blacklabel blackclass="blackform-blackcheck-blacklabel" blackfor="blackdisabledFieldsetCheck">
18 blackCan't blackcheck blackthis
19 </blacklabel>
20 </blackdiv>
21 <blackbutton blacktype="blacksubmit" blackclass="blackbtn blackbtn-blackprimary">blackSubmit</blackbutton>
22 </blackfieldset>
23 </blackform>

```

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



Gambar 16: Button pada Bootstrap

```

1 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackprimary">blackPrimary</blackbutton>
2 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blacksecondary">blackSecondary</blackbutton>
3 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blacksuccess">blackSuccess</blackbutton>
4 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackdanger">blackDanger</blackbutton>
5 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackwarning">blackWarning</blackbutton>
6 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackinfo">blackInfo</blackbutton>
7 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blacklight">blackLight</blackbutton>
8 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackdark">blackDark</blackbutton>
9
10 <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blacklink">blackLink</blackbutton>

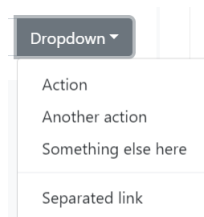
```

5.1.4 Button with Dropdowns

```

1 <blackdiv blackclass="blackinput-blackgroup blackmb-3">
2 <blackdiv blackclass="blackinput-blackgroup-blackprepend">
3 <blackbutton blackclass="blackbtn blackbtn-blackoutline-blacksecondary blackdropdown-blacktoggle" blacktype="
  blackbutton" blackdata-blacktoggle="blackdropdown"
4 blackaria-blackhaspopup="blacktrue" blackaria-blackexpanded="blackfalse">blackDropdown</blackbutton>
5 <blackdiv blackclass="blackdropdown-blackmenu">
6 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackAction</blacka>
7 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackAnother blackaction</blacka>
8 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackSomething blackelse blackhere</blacka>
9 <blackdiv blackrole="blackseparator" blackclass="blackdropdown-blackdivider"></blackdiv>
10 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackSeparated blacklink</blacka>
11 </blackdiv>
12 </blackdiv>
13 <blackinput blacktype="blacktext" blackclass="blackform-blackcontrol" blackaria-blacklabel="blackText blackinput
  blackwith blackdropdown blackbutton">
14 </blackdiv>
15
16 <blackdiv blackclass="blackinput-blackgroup">
17 <blackinput blacktype="blacktext" blackclass="blackform-blackcontrol" blackaria-blacklabel="blackText blackinput
  blackwith blackdropdown blackbutton">
18 <blackdiv blackclass="blackinput-blackgroup-blackappend">
19 <blackbutton blackclass="blackbtn blackbtn-blackoutline-blacksecondary blackdropdown-blacktoggle" blacktype="
  blackbutton" blackdata-blacktoggle="blackdropdown"
20 blackaria-blackhaspopup="blacktrue" blackaria-blackexpanded="blackfalse">blackDropdown</blackbutton>
21 <blackdiv blackclass="blackdropdown-blackmenu">
22 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackAction</blacka>
23 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackAnother blackaction</blacka>
24 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackSomething blackelse blackhere</blacka>
25 <blackdiv blackrole="blackseparator" blackclass="blackdropdown-blackdivider"></blackdiv>
26 <blacka blackclass="blackdropdown-blackitem" blackhref="#">blackSeparated blacklink</blacka>
27 </blackdiv>
28 </blackdiv>
29 </blackdiv>

```

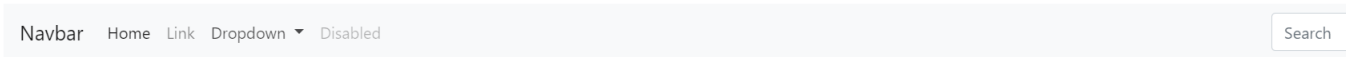


Gambar 17: Tombol *dropdown* pada Bootstrap

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



Gambar 18: Navigation Bar pada Bootstrap

```

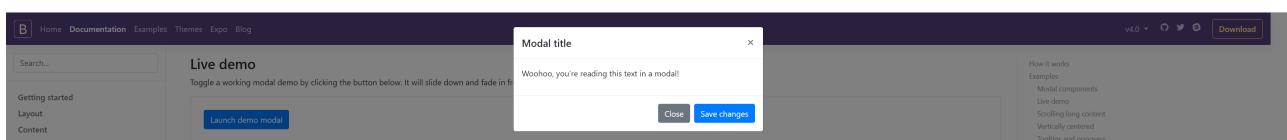
1 <blacknav black class="blacknavbar blacknavbar-blackexpand-blacklg blacknavbar-blacklight blackbg-blacklight">
2 <blacka black class="blacknavbar-blackbrand" black href="#">blackNavbar</blacka>
3 <blackbutton black class="blacknavbar-blacktoggler" black type="blackbutton" black data-blacktoggle="blackcollapse"
  " black data-blacktarget="#blacknavbarsupportedcontent"
4 black aria-blackcontrols="blacknavbarsupportedcontent" black aria-blackexpanded="blackfalse" black aria-
  blacklabel="blackToggle blacknavigation">
5 <blackspan black class="blacknavbar-blacktoggler-blackicon"></blackspan>
6 </blackbutton>
7
8 <blackdiv black class="blackcollapse blacknavbar-blackcollapse" black id="blacknavbarsupportedcontent">
9 <blackul black class="blacknavbar-blacknav blackmr-blackauto">
10 <blackli black class="blacknav-blackitem blackactive">
11 <blacka black class="blacknav-blacklink" black href="#">blackHome <blackspan black class="blacksr-blackonly">{
  blackcurrent}</blackspan></blacka>
12 </blackli>
13 <blackli black class="blacknav-blackitem">
14 <blacka black class="blacknav-blacklink" black href="#">blackLink</blacka>
15 </blackli>
16 <blackli black class="blacknav-blackitem blackdropdown">
17 <blacka black class="blacknav-blacklink blackdropdown-blacktoggle" black href="#" black id="blacknavbardropdown"
  black role="blackbutton"
18 black data-blacktoggle="blackdropdown" black aria-blackhaspopup="blacktrue" black aria-blackexpanded="blackfalse">
19 blackDropdown
20 </blacka>
21 <blackdiv black class="blackdropdown-blackmenu" black aria-blacklabelledby="blacknavbardropdown">
22 <blacka black class="blackdropdown-blackitem" black href="#">blackAction</blacka>
23 <blacka black class="blackdropdown-blackitem" black href="#">blackAnother blackaction</blacka>
24 <blackdiv black class="blackdropdown-blackdivider"></blackdiv>
25 <blacka black class="blackdropdown-blackitem" black href="#">blackSomething blackelse blackhere</blacka>
26 </blackdiv>
27 </blackli>
28 <blackli black class="blacknav-blackitem">
29 <blacka black class="blacknav-blacklink blackdisabled" black href="#">blackDisabled</blacka>
30 </blackli>
31 </blackul>
32 <blackform black class="blackform-blackinline blackmy-2 blackmy-blacklg-0">
33 <blackinput black class="blackform-blackcontrol blackmr-blacksm-2" black type="blacksearch" black placeholder="
  blackSearch" blackaria-blacklabel="blackSearch">
34 <blackbutton black class="blackbtn blackbtn-blackoutline-blacksuccess blackmy-2 blackmy-blacksm-0" black type="
  blacksubmit">blackSearch</blackbutton>
35 </blackform>
36 </blackdiv>
37 </blacknav>

```

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 19: Modal pada Bootstrap

```

1      <!-- blackButton blacktrigger blackmodal -->
2      <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackprimary" blackdata-blacktoggle="
3      blackmodal" blackdata-blacktarget="#blackmyModal">
4      blackLaunch blackdemo blackmodal
5      </blackbutton>
6
7      <!-- blackModal -->
8      <blackdiv blackclass="blackmodal blackfade" blackid="blackexampleModal" blacktabindex="-1" blackrole="
9      blackdialog"
10     blackaria-blacklabelledby="blackexampleModalLabel" blackaria-blackhidden="blacktrue">
11     <blackdiv blackclass="blackmodal-blackdialog" blackrole="blackdocument">
12     <blackdiv blackclass="blackmodal-blackcontent">
13     <blackdiv blackclass="blackmodal-blackheader">
14     <blackh5 blackclass="blackmodal-blacktitle" blackid="blackexampleModalLabel">blackModal blacktitle </blackh5>
15     <blackbutton blacktype="blackbutton" blackclass="blackclose" blackdata-blackdismiss="blackmodal" blackaria-
16     blacklabel="blackClose">
17     <blackspan blackaria-blackhidden="blacktrue">&blacktimes;</blackspan>
18     </blackbutton>
19     </blackdiv>
20     <blackdiv blackclass="blackmodal-blackbody">
21     ...
22     </blackdiv>
23     <blackdiv blackclass="blackmodal-blackfooter">
24     <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blacksecondary" blackdata-blackdismiss="
25     blackmodal">blackClose </blackbutton>
26     <blackbutton blacktype="blackbutton" blackclass="blackbtn blackbtn-blackprimary">blackSave blackchanges </
     blackbutton>
     </blackdiv>
     </blackdiv>
     </blackdiv>
     </blackdiv>

```

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).

```

1      <blacki blackclass="blackfa blackfa-blackcoffee"></blacki>

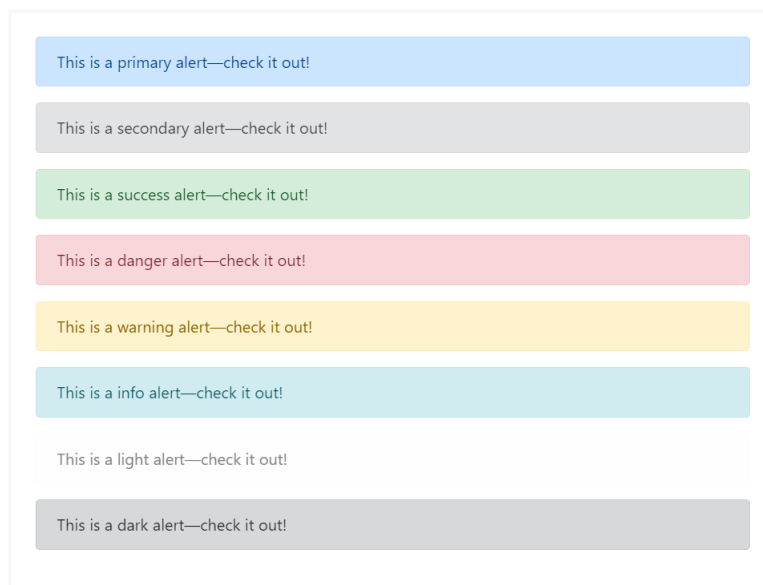
```



Gambar 20: Ikon *Coffee* pada Font Awesome

5.1.5 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 21: Alert pada Bootstrap

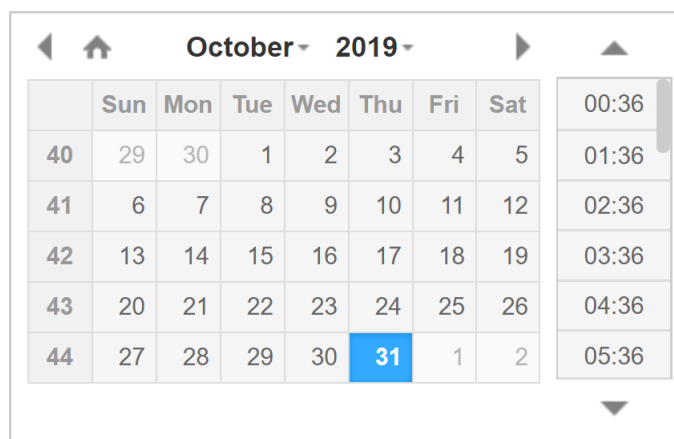
```

1      <blackdiv blackclass="blackalert blackalert-blackprimary" blackrole="blackalert">
2      blackThis blackis blacka blackprimary black a l e r t check blackit blackout!
3      </blackdiv>
4      <blackdiv blackclass="blackalert blackalert-blacksecondary" blackrole="blackalert">
5      blackThis blackis blacka blacksecondary black a l e r t check blackit blackout!
6      </blackdiv>
7      <blackdiv blackclass="blackalert blackalert-blacksuccess" blackrole="blackalert">
8      blackThis blackis blacka blacksuccess black a l e r t check blackit blackout!
9      </blackdiv>
10     <blackdiv blackclass="blackalert blackalert-blackdanger" blackrole="blackalert">
11     blackThis blackis blacka blackdanger black a l e r t check blackit blackout!
12     </blackdiv>
13     <blackdiv blackclass="blackalert blackalert-blackwarning" blackrole="blackalert">
14     blackThis blackis blacka blackwarning black a l e r t check blackit blackout!
15     </blackdiv>
16     <blackdiv blackclass="blackalert blackalert-blackinfo" blackrole="blackalert">
17     blackThis blackis blacka blackinfo black a l e r t check blackit blackout!
18     </blackdiv>
19     <blackdiv blackclass="blackalert blackalert-blacklight" blackrole="blackalert">
20     blackThis blackis blacka blacklight black a l e r t check blackit blackout!
21     </blackdiv>
22     <blackdiv blackclass="blackalert blackalert-blackdark" blackrole="blackalert">
23     blackThis blackis blacka blackdark black a l e r t check blackit blackout!
24     </blackdiv>

```

Plugin :

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



Gambar 22: Datetimepicker pada Bootstrap

Penggunaan nya dalam kode HTML sebagai berikut :

```

1      <blackinput blackid="blackdatetimepicker" blacktype="blacktext" >

```

Penggunaan dalam kode Javascript sebagai berikut :

```

1      blackjQuery('#blackdatetimepicker').blackdatetimepicker();

```

3. Membuat rancangan tampilan website dan template.

Status : Ada sejak rencana kerja skripsi.

Hasil : Sebagian besar sudah diimplementasikan.

4. Mengimplementasikan keseluruhan rancangan tampilan dan template.

Status : Ada sejak rencana kerja skripsi.

Hasil : Sebagian besar tampilan sudah diimplementasi

5. Menulis dokumen skripsi.

Status : Ada sejak rencana kerja skripsi.

Hasil : Mengerjakan bab 1, bab 2 dan bab 3 Proyek BlueTape dijalankan menggunakan *framework*

Anda berhasil logout



Silahkan login di aplikasi BlueTape, dengan menekan tombol di bawah ini.
Gunakan akun **@unpar.ac.id** atau **@student.unpar.ac.id**.

[Login dengan Google](#)
[Petunjuk Penggunaan](#)

Gambar 23: Tampilan Cetak Transkrip dengan Bootstrap

[Cetak Transkrip](#)
[Manajemen Cetak Transkrip](#)
[Perubahan Kuliah](#)
[Manajemen Perubahan Kuliah](#)
[Entri Jadwal Dosen](#)
[Lihat Jadwal Dosen](#)
[Logout](#)

Permohonan Baru

Yang memohon:

amihapsahapsa@gmail.com

NPM:

-

Nama:

Hapsari Laksmi Wijayanti

Tipe Transkrip:

DPS (Seluruh Semester, Bilingual)

Keperluan:

Kirim Permohonan

Histori Permohonan

ID	Status	Tanggal Permohonan	Tipe Transkrip	Tanggal Jawab/Cetak	Keterangan	Aksi
#4	DITOLAK	Minggu, 27 Oktober 2019	DPS_EN	Selasa, 12 November 2019	-	👁
#3	TERCETAK	Kamis, 2 Mei 2019	DPS_ID	Minggu, 27 Oktober 2019	tercetak	👁

Gambar 24: Tampilan Cetak Transkrip dengan Bootstrap

front-end Foundation. Secara garis besar, file-file yang berkaitan dengan Foundation seperti file javascript dan css akan dipanggil di file BlueTape/www/application/views/templates/script_foundation.php dan BlueTape/www/application/views/templates/head_loggedin.php. Kemudian penggunaan komponen Foundation terdapat pada file-file main.php yang terletak di setiap modul pada folder BlueTape/www/application/views/ beserta kegunaannya pada website.

Analisis Frontend Library

Foundation Dalam proyek BlueTape, file Foundation tersimpan di folder js dan css. Foundation yang digunakan pada proyek ini adalah versi 6.1.2. Detail komponen yang digunakan adalah sebagai berikut :

(a) `mygray.row`

- Untuk membuat konten yang terletak di dalam satu baris untuk setiap halaman website.
- Untuk memisahkan baris dari sekumpulan field dalam sebuah form.

(b) `mygray.column`

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

Hapsari Laksmi Wijayanti

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

Terakhir diupdate pada : 29 Oktober 2019

[Ekspor ke XLS](#)

Gambar 25: Tampilan Lihat Jadwal Dosen dengan Bootstrap

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

Permintaan Transkrip

Cari NPM: 2013730013 [Cari](#)

ID	Status	Tanggal Permohonan	Tipe Transkrip	NPM	Aksi
#4	DITOLAK	Minggu, 27 Oktober 2019	DPS_EN	-	View Edit Delete
#3	TERCETAK	Kamis, 2 Mei 2019	DPS_ID	-	View Edit Delete
#2	DITOLAK	Rabu, 1 Mei 2019	DPS_ID	2015730037	View Edit Delete

Gambar 26: Tampilan-Manajemen-Cetak-Transkrip dengan Bootstrap

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

Tambah Jadwal

Hari: Durasi: Label:

Jam Mulai: Jenis: [Tambah](#)

Daftar Jadwal

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

Terakhir diupdate pada : 29 Oktober 2019

[Delete All](#) [Ekspor ke XLS](#)

Gambar 27: Tampilan Entri Jadwal Dosen dengan Bootstrap

- Membuat kolom untuk menampung konten
- Membuat kolom pada field dalam form

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

Permohonan Baru

Pemohon: amihapsahapsa@gmail.com Nama: Hapsari Laksmi Wijayanti

Kode MK: Nama Mata Kuliah: Kelas: Jenis Perubahan: Diganti

Dari Hari & Jam: Dari Ruang: Keterangan Tambahan:

Menjadi Hari & Jam: Menjadi Ruang:

[Kirim Permohonan](#) [Tambah Pertemuan Ekstra](#)

Histori Permohonan

ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Tanggal Jawab	Keterangan	Aksi
#3	DITOLAK	Rabu, 30 Oktober 2019	AIF123	Diganti			Detail
#2	TERKONFIRMASI	Selasa, 29 Oktober 2019	AIF001	Diganti	Selasa, 29 Oktober 2019	pengumuman sudah ditempel	Detail
#1	DITOLAK	Rabu, 1 Mei 2019	AIF123	Ditadakan	Rabu, 1 Mei 2019	tidak bisa	Detail

Gambar 28: Tampilan Perubahan Kuliah dengan Bootstrap

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

Permohonan Perubahan Kuliah

ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Aksi
#3	MENUNGGU	Rabu, 30 Oktober 2019	AIF123	Diganti	Detail Print Like Share Delete
#2	TERKONFIRMASI	Selasa, 29 Oktober 2019	AIF001	Diganti	Detail Print Like Share Delete
#1	DITOLAK	Rabu, 1 Mei 2019	AIF123	Ditadakan	Detail Print Like Share Delete

Gambar 29: Tampilan Manajemen Perubahan Kuliah dengan Bootstrap

(c) `mygray<table>`

- Seluruh tabel dalam proyek BlueTape memiliki format yang akan menyesuaikan posisinya dengan menampilkan data nya secara bertumpuk, sehingga dibutuhkan tag `mygray<table>` dan kelas `mygray.stack`. Tabel akan terdiri dari satu tag `mygray<thead>` dan `mygray<tbody>`.
- Bagian thead terdiri dari satu tag `mygray<tr>` dan beberapa tag `mygray<th>` yang membuat tulisan di dalam sel bersifat **bold**. Bagian tbody terdiri dari satu tag `mygray<tr>` dan beberapa tag `mygray<td>`.

Tabel yang menggunakan kelas ini sebagai berikut :

Tabel	Modul	Keterangan
Daftar Jadwal	Entri Jadwal Dosen	
Permohonan Perubahan Kuliah	Perubahan Kuliah Manage	
Detail Permohonan	Perubahan Kuliah Manage	Modal dari Aksi Lihat
Histori Permohonan	Perubahan Kuliah Request	
Permintaan Transkrip	Transkrip Manage	
Detail Permohonan	Transkrip Manage	Modal dari Aksi Lihat
Histori Permohonan	Transkrip Request	Modal dari Aksi Lihat

(d) Kelas `mygray.callout` : Border untuk menampung setiap konten di website BlueTape seperti form dan tabel.

(e) Kelas `mygray.large-*`, `mygray.medium-*` :

- Komponen *medium-12* digunakan dalam callout, sehingga apabila layar berukuran sedang, callout akan memiliki lebar 12 grid.
- Sedangkan komponen *large-** digunakan untuk mengatur lebar suatu input field dalam satu form. Lebar field berbeda - beda tergantung presentase suatu input field dalam satu baris.

(f) Tag `mygray<form>` :

Ada dua jenis metode yang digunakan dalam `form` :

- **POST** : Digunakan untuk memasukan input user yang disertai oleh aksi yang memanggil suatu method tertentu dari controller. Dalam BlueTape form post digunakan untuk membuat request transkrip, setiap aksi untuk konfirmasi permohonan transkrip, permohonan baru perubahan kuliah.
- **GET** : Digunakan untuk mencari permintaan transkrip berdasarkan NPM.

Atribut yang digunakan dalam website BlueTape :

- `mygrayaria-label` : digunakan untuk memberi keterangan pada input field dalam form.
- `mygrayaria-hidden` : digunakan ketika ingin mengambil data dari database atau data dari akun yang teregistrasi.
- `mygrayaria-selected` : digunakan apabila input terdiri dari beberapa pilihan dan menggunakan empat

(g) **Reveal Modal** digunakan untuk menampilkan data tertentu berdasarkan aksi yang dipilih. Terdapat empat aksi yang menggunakan modal yaitu modal lihat, setuju, tolak dan hapus.

- Modal memiliki kelas `mygray.reveal` dan atribut `mygraydata-reveal`.
- Setiap modal akan memiliki tombol hapus. Penggunaan kelas `.close-button` dan atribut `mygraydata-close` akan diaplikasikan.

5.2 Antarmuka Cetak Transkrip

Antarmuka diaplikasikan pada file `BlueTape/www/application/views/TranskripRequest/main.php`. Isi dari halaman antarmuka cetak transkrip terdiri dari dua bagian yaitu :

(a) **Permohonan Baru** : Sistem akan memberikan dua tampilan untuk bagian ini, dengan kondisi sebagai berikut:

- Sistem akan menampilkan form pengajuan transkrip, apabila mahasiswa belum pernah mengajukan permohonan atau pengajuan sebelumnya dikonfirmasi staf TU, maka mahasiswa dapat mengajukan permohonan baru.
- Sistem akan menampilkan informasi "Anda tidak bisa meminta cetak karena ada permintaan lain yang belum selesai", apabila mahasiswa memiliki pengajuan permohonan transkrip yang belum dikonfirmasi staf TU.

(b) **Histori Permohonan** : Tabel untuk menampilkan informasi permohonan transkrip seorang mahasiswa. Status, tanggal pembuatan, tipe transkrip, tanggal cetak keterangan dan aksi.

Desain antarmuka sebagai berikut :

Konten **Permohonan Baru** dan **Histori Permohonan** akan diletakkan pada satu *row* yang memiliki kolom sebesar 12 grid pada layar medium dengan menggunakan komponen *medium-12 column*, untuk setiap kontennya akan dipisahkan oleh panel yang disebut dengan *callout*.

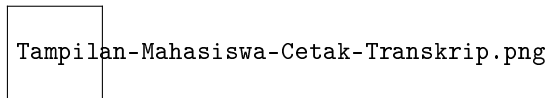
Untuk konten **Permohonan Baru** :

- Bagian isi akan memiliki dua tampilan yaitu berbentuk form atau notifikasi yang berbentuk paragraf. Sehingga dibutuhkan komponen `<form>` dan `<p>`.

- Form terdiri dari dua baris yang dipisahkan oleh komponen `row`.
- Untuk field pada baris pertama akan memiliki kolom dengan panjang 4 grid. Sehingga dibutuhkan kelas `large-4 column`. Jenis input yang digunakan bertipe email dan text. Selain itu untuk setiap input akan disertakan satu label sehingga menggunakan komponen `<label>`.
- Untuk field pada baris kedua memiliki dua jenis lebar grid yaitu 4 grid dan 8 grid. Sehingga membutuhkan kelas `large-4 column`, `large-8 column`. Untuk field pada baris ketiga, tombol "Kirim Permohonan" memiliki *background color* berwarna biru sehingga akan digunakan kelas `button`.

Untuk konten "Histori Permohonan" :

- Pada kolom "Status" akan memiliki tiga jenis bentuk alert dengan *background* warna hijau, abu-abu dan merah. Sehingga dibutuhkan kelas `warning`, `success`, `alert`.
- Aksi memiliki satu ikon "lihat" berwarna biru yang akan menampilkan sebuah modal. Sehingga dibutuhkan kelas ikon `fi-eye`.



Gambar 30: Antarmuka Cetak Transkrip bagian 1

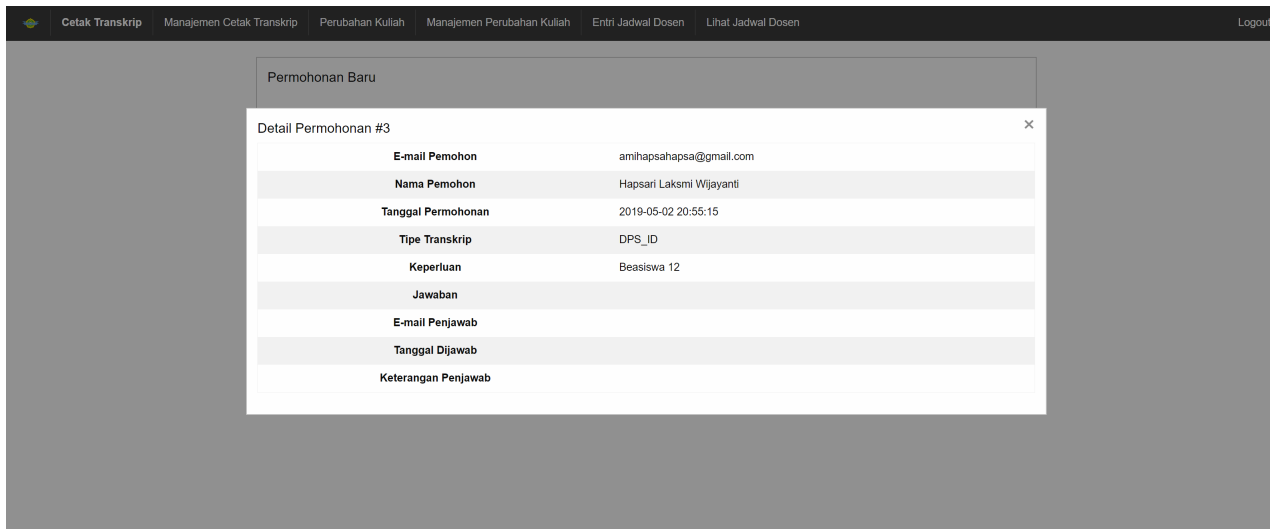
ID	Status	Tanggal Permohonan	Tipe Transkrip	Tanggal Jawab/Cetak	Keterangan	Aksi
#3	TUNGGU	Kamis, 2 Mei 2019	DPS_ID			

Gambar 31: Antarmuka Cetak Transkrip bagian 2

Detail dari form permohonan baru, semua field akan disertai sebuah label untuk menjelaskan fungsi setiap field sehingga membutuhkan tag `<label>` :

- **Yang memohon** : Berisi email UNPAR mahasiswa, otomatis terisi saat login melalui gmail. Sehingga field tidak bisa diisi atau *disabled* dibutuhkan atribut boolean `readonly`
- **NPM** : Berisi NPM mahasiswa yang ter-*generate* secara otomatis. Sehingga field tidak bisa diisi atau *disabled* dibutuhkan atribut boolean `readonly` dan `<input>` dengan tipe text.
- **Nama** : Nama mahasiswa yang tergenerate secara otomatis. Sehingga field tidak bisa diisi atau *disabled* dibutuhkan atribut boolean `readonly` dan `<input>` dengan tipe text.
- **Tipe Transkrip** : Terdiri dari tiga pilihan yaitu DPS Bahasa Indonesia(Seluruh Semester), DPS Bahasa Inggris(Seluruh Semester), LHS (Semester Terakhir). Wajib diisi. Tag `<select>` dan `<option>` yang *value* nya diambil dari variabel `$type`.
- **Keperluan** : Keterangan keperluan dibuat nya transkrip, wajib diisi mahasiswa. Menggunakan tag `<input>` dengan tipe text.

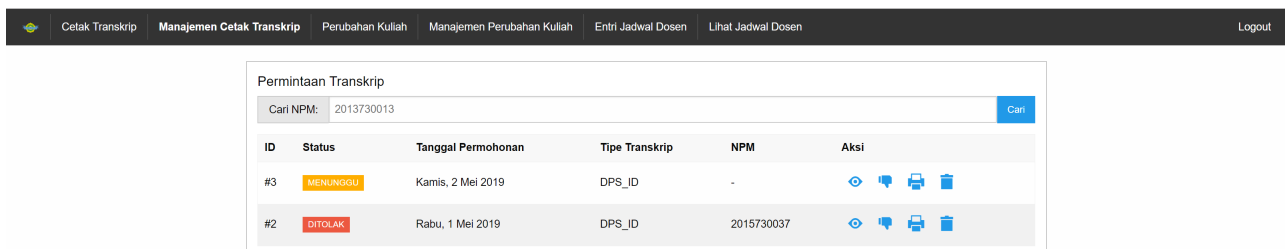
Apabila ada form yang belum diisi maka akan terdapat *warning* untuk *field* yang kosong. Berikut ini apabila mahasiswa menekan tombol aksi lihat 👁 :



Gambar 32: Modal Lihat Cetak Transkrip

Disini aksi 'lihat' akan menampilkan sebuah modal yang berisi sebuah tabel bergaris yang menyimpan informasi detail permohonan, baik detail informasi dari mahasiswa maupun konfirmasi dari staf Tata Usaha. Sehingga dibutuhkan kelas `.reveal` dan atribut `data-reveal`.

5.3 Antarmuka Manajemen Cetak Transkrip



Gambar 33: Tampilan Manajemen Cetak Transkrip

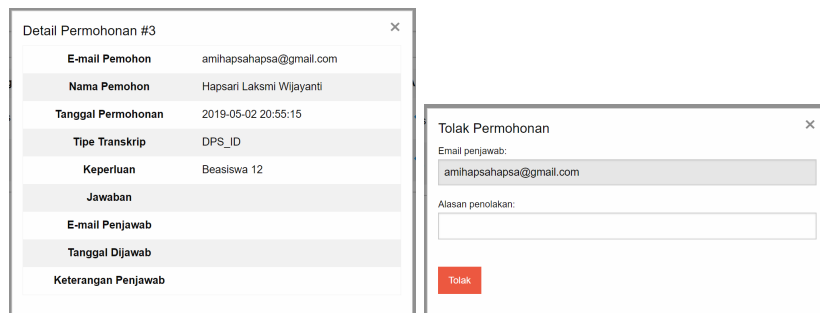
Tampilan Manajemen Cetak Transkrip berisi sebuah tabel permintaan transkrip yang terdiri dari daftar permintaan transkrip dan form pencarian transkrip berdasarkan NPM.

Detail penjelasan untuk field 'Status' dan 'Aksi' :

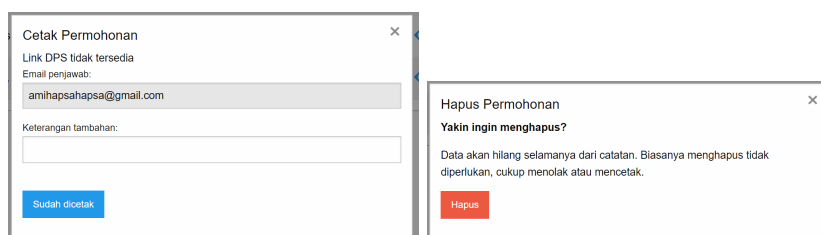
- **Status** : Output terdiri dari tiga jenis label yaitu 'MENUNGGU' (berwarna kuning), 'DITOLAK' (berwarna merah) dan 'TERCETAK' (berwarna hijau). Sehingga dibutuhkan kelas `secondary`, `success`, `alert`.
- **Aksi** : Terdiri dari empat ikon font-awesome yaitu `fi-eye`, `fi-dislike`, `fi-print`, `fi-trash`, yang akan menampilkan modal berisi informasi yang sesuai dengan perintah.

Modal akan menggunakan kelas `reveal` dan atribut `data-reveal` Detail penjelasan untuk modal :

- (a) Modal Lihat : Terdiri dari sebuah **table** yang menampilkan data permintaan transkrip. Ikon menggunakan kelas **fi-eye** dan menerapkan atribut **data-open** yang berisi method hapus menuju ID tertentu.
- (b) Modal Tolak : Terdiri dari sebuah **form** yang memiliki method **POST** yang memanggil sebuah method **"/TranskripManage/answer"**. Terdapat tiga tipe input yang digunakan yaitu **hidden**, **text**, **submit**. Pada input text untuk label Alasan Penolakan, menggunakan kelas **"input-group-field"**. Lalu untuk input bertipe **submit** menggunakan kelas **alert-button** untuk membuat button berwarna merah. Ikon menggunakan kelas **fi-dislike** dan menerapkan atribut **data-open** yang berisi method tolak menuju ID tertentu.
- (c) Modal Print : Terdiri dari sebuah form yang terdiri dari label, field dan tombol berwarna biru sehingga membutuhkan kelas **input-group-field**, **button**. Ikon menggunakan kelas **fi-print** dan menerapkan atribut **data-open** yang berisi method cetak menuju ID tertentu.
- (d) Modal Hapus : Terdiri dari sebuah form yang terdiri dari paragraf yang bersifat **bold**, beberapa input dan tombol berwarna merah sehingga membutuhkan kelas **<p>**, ****, **<input>**. Ikon menggunakan kelas **fi-trash** dan menerapkan atribut **data-open** yang berisi method hapus menuju ID tertentu.




Gambar 34: Tampilan Modal untuk aksi 'Lihat' dan 'Tolak'



Gambar 35: Tampilan Modal untuk aksi 'Print' dan 'Hapus'

5.4 Antarmuka Perubahan Kuliah


[Cetak Transkrip](#)
[Manajemen Cetak Transkrip](#)
[Perubahan Kuliah](#)
[Manajemen Perubahan Kuliah](#)
[Entri Jadwal Dosen](#)
[Lihat Jadwal Dosen](#)
[Logout](#)

Permohonan Baru

Pemohon: Nama:

Kode MK: Nama Mata Kuliah: Kelas: Jenis Perubahan:

Dari Hari & Jam: Dari Ruang: Keterangan Tambahan:

Menjadi Hari & Jam: Menjadi Ruang:

Menjadi Hari & Jam: Menjadi Ruang:

Histori Permohonan

ID	Status	Tanggal Permohonan	Kode MK	Perubahan	Tanggal Jawab	Keterangan	Aksi
#1	DITOLAK	Rabu, 1 Mei 2019	AIF123	Ditiadakan	Rabu, 1 Mei 2019	tidak bisa	

Gambar 36: Tampilan Perubahan Kuliah

Modul Perubahan Kuliah terdiri dari dua tabel yaitu :

5.5 Permohonan Baru

Permohonan baru diletakkan dalam sebuah row selebar 12 grid dan dikelilingi oleh sebuah border, membutuhkan kelas `row`, `large-12 column` dan `callout`. Selain itu sebuah form dengan tipe `POST` akan ditampilkan untuk menyimpan masukan pemohon. Format dari form sebagai berikut :

Nama Kolom	Pilihan	Keterangan
Pemohon		Field terisi secara otomatis, user tidak bisa mengisi field atau <i>disabled</i> dan memiliki lebar 4 grid pada layar besar, sehingga membutuhkan kelas <code>readonly, large-4 column</code> .
Nama		Field terisi secara otomatis, user tidak bisa mengisi field atau <i>disabled</i> dan memiliki lebar 4 grid pada layar besar, sehingga membutuhkan kelas <code>readonly, large-8 column</code> .
Kode MK		Field bertipe text dan memiliki lebar 2 grid pada layar besar, sehingga membutuhkan kelas <code>large-2 column</code> .
Nama Mata Kuliah		Field bertipe text dan memiliki lebar 5 grid pada layar besar, sehingga membutuhkan kelas <code>large-5 column</code>
Kelas		Field bertipe text dan memiliki lebar 1 grid pada layar besar, sehingga membutuhkan kelas <code>large-1 column</code>
Jenis Perubahan	Diganti	Field 'Dari Hari dan Jam' dan 'Dari Ruang' disabled dan memiliki lebar 4 grid pada layar besar, sehingga membutuhkan kelas <code>large-4 column</code>
	Tambahkan	Field 'Dari Hari dan Jam' dan 'Dari Ruang' dapat ditambahkan lebih dari satu kolom dan memiliki lebar 4 grid pada layar besar, sehingga membutuhkan kelas <code>large-4 column</code>
	Ditiadakan	Field 'Menjadi Hari dan Jam' dan 'Menjadi Ruangan' disabled dan memiliki lebar 4 grid pada layar besar, sehingga membutuhkan kelas <code>large-4 column</code> dan <code>readonly</code>
Dari Hari dan Jam		Menggunakan plugin <code>xdatetimepicker</code> yang bisa menampilkan pilihan tanggal dan waktu. Memiliki lebar 3 grid pada layar besar, sehingga membutuhkan kelas <code>large-3 column</code>
Dari Ruang		Field bertipe text dan memiliki lebar 3 grid pada layar besar, sehingga membutuhkan kelas <code>large-3 column</code>
Keterangan Tambahan		Field bertipe text, sehingga membutuhkan kelas <code>form-control</code>
Menjadi Hari dan Jam		Menggunakan plugin <code>xdatetimepicker</code> yang bisa menampilkan pilihan tanggal dan waktu. Memiliki lebar 3 grid pada layar besar, sehingga membutuhkan kelas <code>large-3 column</code>
Menjadi Ruang		Field bertipe text dan memiliki lebar 3 grid pada layar besar, sehingga membutuhkan kelas <code>large-3 column</code>
Tombol Aksi Tambah Pertemuan Ekstra	Kirim Permohonan Berbentuk button dengan menggunakan kelas <code>button secondary</code>	Berbentuk button dengan menggunakan kelas <code>button</code>

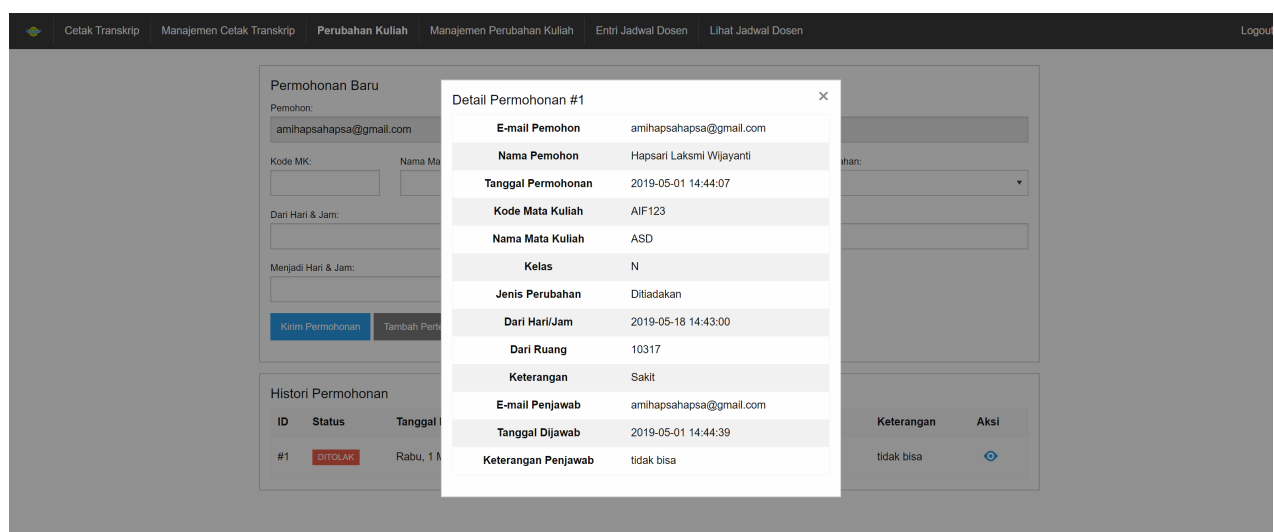
5.6 Histori Pemohonan

Histori Permohonan menggunakan tabel yang bisa menyesuaikan lebarnya dengan menampilkan data secara bertumpuk sehingga menggunakan kelas `stack`. Jenis data akan diletakkan dalam tag `<thead>` dalam satu baris sehingga menggunakan satu tag `<tr>` dan delapan tag `<th>`. Untuk bagian isi data

menggunakan tag `<tbody>`.

Nama Kolom	Pilihan	Keterangan
Status	dikonfirmasi, menggunakan kelas <code>success</code>	Apabila staf TU menyetujui permohonan
	ditolak, menggunakan kelas <code>alert</code>	Apabila staf TU menolak permohonan
	ditunggu, menggunakan kelas <code>secondary</code>	Apabila staf TU belum konfirmasi permohonan
Tanggal Permohonan		Data bertipe tanggal dengan format yang sesuai
Kode MK		Data berbentuk text
Perubahan		Data berbentuk text
Tanggal Jawab		Data bertipe tanggal
Keterangan		Data berbentuk text
Aksi		Terdapat tombol aksi 'Lihat', menggunakan kelas <code>fas fa-eyes</code> yang akan memanggil sel yang diinginkan user. Untuk modal, menggunakan atribut <code>data-reveal</code>

Ketika user menekan tombol aksi 'lihat', maka modal berisi sebuah tabel informasi data permohonan akan ditampilkan sesuai dengan ID.



Gambar 37: Modal Lihat Perubahan Kuliah

5.7 Antarmuka Manajemen Perubahan Kuliah



Gambar 38: Tampilan Manajemen Perubahan Kuliah

Tabel Pemohonan Kuliah memiliki detail yang sama dengan tabel histori permohonan, namun aksi yang dilakukan terdiri dari lima perintah:

Nama Kolom	Pilihan	Keterangan
Status	dikonfirmasi, menggunakan kelas success	Apabila staf TU menyetujui permohonan
	ditolak, menggunakan kelas alert	Apabila staf TU menolak permohonan
	ditunggu, menggunakan kelas secondary	Apabila staf TU belum konfirmasi permohonan
Tanggal Permohonan		Data bertipe tanggal dengan format yang sudah ditentukan
Kode MK		Data berbentuk text
Perubahan		Data berbentuk text
Tanggal Jawab		Data bertipe tanggal
Keterangan		Data berbentuk text
Aksi	ikon Lihat	Ikon menggunakan font awesome dan kelas fas fa-eyes yang akan memanggil sebuah modal sesuai id yang diinginkan user. Untuk modal, menggunakan kelas reveal dan atribut data-reveal
	ikon Print	Ikon menggunakan font awesome dan kelas fas fa-eyes yang akan memanggil sebuah modal sesuai id yang diinginkan user. Untuk modal, menggunakan kelas reveal dan atribut data-reveal
	ikon Setuju	Ikon menggunakan font awesome dan kelas fas fa-eyes yang akan memanggil sebuah modal sesuai id yang diinginkan user. Untuk modal, menggunakan kelas reveal dan atribut data-reveal
	ikon Tolak	Ikon menggunakan font awesome dan kelas fas fa-eyes yang akan memanggil sebuah modal sesuai id yang diinginkan user. Untuk modal, menggunakan kelas reveal dan atribut data-reveal
	ikon Hapus	Ikon menggunakan font awesome dan kelas fas fa-eyes yang akan memanggil sebuah modal sesuai id yang diinginkan user. Untuk modal, menggunakan kelas reveal dan atribut data-reveal

PENGUMUMAN
Jadwal Kuliah

Detail Permohonan #1

E-mail Pemohon	amhaphapusa@gmail.com
Nama Pemohon	Hapsari Laksmi Wijayanti
Tanggal Permohonan	2019-05-01 14:44:07
Kode Mata Kuliah	AIF123
Nama Mata Kuliah	ASD
Kelas	N
Jenis Perubahan	Ditambahkan
Dari Hari/Jam	2019-05-15 14:43:00
Dari Ruang	10217
Keterangan	Sakit
E-mail Penjawab	amhaphapusa@gmail.com
Tanggal Dijawab	2019-05-01 14:44:39
Keterangan Penjawab	tidak bisa

Mata Kuliah	: ASD
Kode MK	: AIF123
Kelas	: N
sementara	
Hari	: Sabtu, 18 Mei 2019
Jam	: 14:43
Ruang	: 10217
ditambahkan	
Keterangan	: Sakit
Buatang, 27-05-2019	
Adminitrasi FTIS	

Gambar 39: Modal aksi Lihat dan Print Manajemen Perubahan Kuliah

Konfirmasi Permohonan

Diajukan pemohon: amhaphapusa@gmail.com

Keterangan:

Tolak Permohonan

Diajukan pemohon: amhaphapusa@gmail.com

Alasan penolakan:

Gambar 40: Modal aksi Setuju dan Tolak Manajemen Perubahan Kuliah

Hapus Permohonan

Yakin ingin menghapus?

Saya akan tetap menghapus dari database. Biasanya menghapus tidak dibatalkan, cukup membatalkan atau membatalkan.

Gambar 41: Modal Hapus Manajemen Perubahan Kuliah

5.8 Antarmuka Entri Jadwal Dosen

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

Tambah Jadwal

Hari
Senin

Durasi
1 jam

Label

Jam Mulai
7:00

Jenis
Konsultasi

Daftar Jadwal

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

Terakhir diupdate pada : 01 Mei 2019

Gambar 42: Modal Print Manajemen Perubahan Kuliah

Detail mengenai tabel Tambah Jadwal :

- **Hari** : Terdiri dari nama hari dari senin sampai jumat.
- **Durasi** : Terdiri dari rentang jam kelas berlangsung dari 1 jam hingga 9 jam.
- **Label** : Field bertipe text.
- **Jam Mulai** : Terdiri dari jam dari rentang 07:00 sampai 16:00.
- **Jenis** : Terdiri dari tiga macam pilihan
 - (a) **Konsultasi** : Memiliki background berwarna hijau.

- (b) Terjadwal : Memiliki background berwarna biru.
- (c) Kelas : Memiliki background putih.

Desain antarmuka sebagai berikut :

Konten "Tambah Jadwal" dan "Daftar Jadwal" akan diletakkan pada satu *row* yang memiliki kolom sebesar 12 grid pada layar medium dengan menggunakan komponen *large-12 column*, untuk setiap konten nya akan dipisahkan oleh panel yang disebut dengan *callout*.

Untuk konten **Tambah Jadwal** :

- Hari : Penggunaan kelas `<select>` dan `<option>`.
- Jam Mulai : Penggunaan kelas `<select>` dan `<option>`.
- Durasi : Penggunaan kelas `large-4 columns`

Tabel Daftar jadwal akan *retrieve* data jadwal dari dosen yang dibuat. Terdiri dari rentang waktu dan hari. Jadwal yang terlihat pada tabel ini bisa diedit dan dihapus. Menggunakan kelas `large-12 column` *callout* dan `table-scroll`. Dibagian bawah tabel akan terlihat tanggal jadwal tersebut di update dan memiliki dua tombol :

- **Delete All** : Menghapus semua jadwal yang sudah dibuat, tombol berwarna merah dengan menerapkan kelas `alert button`.
- **Export ke XLS** : Secara otomatis akan membuat file excel dan mendownload di device secara lokal. Tombol berwarna biru dengan menerapkan kelas `button`.

Setiap data yang ditampilkan bisa diedit atau hapus, apabila ditekan maka sebuah modal akan muncul.



Gambar 43:

Modal merupakan sebuah form dengan metode "POST". Untuk field **Hari**, **Jam Mulai**, **Durasi**, **Jenis** dan **label** memiliki input bertipe `select`. Lalu terdapat dua button **Save** dan **Submit** yang masing - masing menggunakan kelas `button` dan `alert button`.

5.9 Antarmuka Lihat Jadwal Dosen

Cetak Transkrip

Manajemen Cetak Transkrip

Perubahan Kuliah

Manajemen Perubahan Kuliah

Entri Jadwal Dosen

Lihat Jadwal Dosen

Logout

Hapsari Laksmi Wijayanti

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

Terakhir diupdate pada : 01 Mei 2019

Ekspor ke XLS

Gambar 44: Struktur File Zurb Foundation

Tabel Jadwal Dosen terdiri dari list nama dosen yang dibuat dalam bentuk tabs sehingga menggunakan kelas `tabs-panel`. Apabila dipilih sebuah tabs maka tabel akan menampilkan data jadwal dosen. Terdiri dari rentang waktu dan hari. Dibagian bawah tabel terdapat tanggal kapan terakhir jadwal dibuat dan tombol "Ekspor ke XLS".

6 Pencapaian Rencana Kerja

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

1. Mempelajari framework PHP Codeigniter.
2. Mempelajari framework front-end Bootstrap 4 beserta plugin-plugin yang tersedia.
3. Mengimplementasikan sebagian besar framework front-end Bootstrap 4 beserta plugin-plugin yang tersedia.
4. Menulis dokumen skripsi.

7 Kendala yang Dihadapi

Kendala - kendala yang dihadapi selama mengerjakan skripsi :

- Sulit mengubah lebar cell dalam tabel.

Bandung, 11/14/2019

Hapsari Laksmi W

Menyetujui,

Nama: Pascal Alfadian .N
Pembimbing Tunggal