



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Posisi yang diberikan perusahaan dalam melakukan proses kerja magang di PT. Indosterling Technomedia adalah sebagai *Software Engineer*. Tugas yang diberikan adalah membuat sebuah aplikasi *form builder online* berbasis *website* bernama Modaftar. Bapak Yoas selaku *Chief Technology Officer* (CTO) PT. Indosterling Technomedia bertugas sebagai pembimbing dalam memberikan arahan serta sebagai *project leader* dalam pembuatan aplikasi Modaftar.

3.2 Tugas yang Dilakukan

Pekerjaan yang dilakukan selama proses magang di PT. Indosterling Technomedia adalah membangun aplikasi Modaftar. Modaftar adalah sebuah aplikasi *formbuilder online* berbasis *web*. Modaftar dibuat dengan tujuan untuk mengurangi penggunaan kertas yang biasa digunakan untuk pembuatan formulir umpan balik, formulir pendaftaran, dan lain-lain. Modaftar dibangun menggunakan bahasa pemrograman Python dengan *web framework* Django.

Aplikasi Modaftar yang dibangun masih bersifat *prototype*. Pada aplikasi Modaftar setiap *client* akan diberikan akun administrator. Akun administrator dapat digunakan untuk menambah, mengubah, dan menghapus *event* dan pada setiap *event* administrator dapat menambah, mengubah dan menghapus *questionnaire*. Administrator dapat membuat satu atau lebih *questionnaire* pada setiap *event* yang dibuatnya. Administrator dapat berpindah ke *user mode*, dimana di dalam mode ini

website akan melakukan penghapusan sesi dan menampilkan formulir yang telah dibuat sebelumnya.

3.3 Uraian Pelaksanaan Kerja Magang

Berikut adalah uraian pelaksanaan kerja magang yang akan dilakukan dalam membangun aplikasi Modafatar pada PT Indosterling Technomedia.

3.3.1 Proses Pelaksanaan Kerja Magang

Proses pembangunan aplikasi baik pada modul admin dan modul API dilakukan dengan menggunakan bahasa pemrograman Python dengan *framework* Django dan *Sublime Text* sebagai *Interactive Development Environment*. Proses pelaksanaan kerja magang dilakukan dalam kurun waktu sembilan minggu. Pada Tabel 3.1 menjelaskan rincian selama menjalani praktek kerja magang.

Tabel 3.1 Jadwal Proses Pelaksanaan Kerja Magang

No	Kegiatan	Minggu								
		1	2	3	4	5	6	7	8	9
1	Pengenalan Python, Flask, Django	■	■	■	■					
2	Perancangan Aplikasi		■	■	■					
3	Pembangunan Sistem									
4	Pengujian dan Evaluasi					■			■	■
5	Dokumentasi								■	■

Berikut adalah penjelasan dari masing-masing tahapan serta waktu pelaksanaan selama proses pembangunan aplikasi Modafatar.

1. Minggu Pertama

Pada minggu pertama, pembimbing memberikan pengetahuan seputar bahasa pemrograman dan *framework* yang digunakan untuk membangun

aplikasi yang akan diberikan. Pembimbing memberikan sebuah materi dan pembelajaran dilakukan secara individu, dimana pembimbing memberikan topik atau bahan yang harus dikuasai dan waktu untuk belajar sesuai dengan materi yang diberikan. Materi yang diberikan adalah bahasa pemrograman Python 3 dan *framework* Flask.

2. Minggu Kedua hingga Minggu Keempat

Pada minggu kedua, mulai dilakukannya perancangan dari struktur aplikasi Modafatar. Aplikasi Modafatar dirancang menggunakan *framework* Flask, sehingga pembimbing memberikan materi baru untuk dipelajari dan aplikasi mulai dibangun dengan *framework* Flask. Pada minggu ketiga dilakukan rapat dengan pembimbing dan *project leader*. Hasil dari rapat tersebut adalah aplikasi Modafatar akan dibangun menggunakan *web framework* Django dengan database Postgresql. Dengan pergantian *framework* ini, pembimbing memberikan beberapa materi dan sebuah buku yang harus dipelajari hingga minggu keempat.

3. Minggu Kelima

Pada minggu kelima, dilakukan persiapan pada bagian admin untuk melakukan *demo* dan evaluasi tahap satu ke pembimbing. Evaluasi dan *demo* dilakukan untuk pengecekan terhadap hasil yang telah dilakukan selama menjalani proses kerja magang.

4. Minggu Keenam dan Ketujuh

Pada minggu keenam, dilakukan pembuatan API pada sisi *user*. Pada minggu ketujuh terdapat pergantian pada sisi tampilan antar muka.

5. Minggu Kedelapan dan Kesembilan

Pada minggu kedelapan dan kesembilan dilakukan pengujian, *debugging* dan pencarian celah-celah kesalahan yang mungkin dapat mengakibatkan kerusakan terhadap aplikasi. Pengujian terhadap aplikasi dilakukan dengan menggunakan *testing tools* yang telah disediakan *web framework* Django. Setelah memastikan dan memperbaiki kesalahan-kesalahan sistem, dilakukan dokumentasi dan *refactoring* terhadap penulisan program dengan bahasa Python.

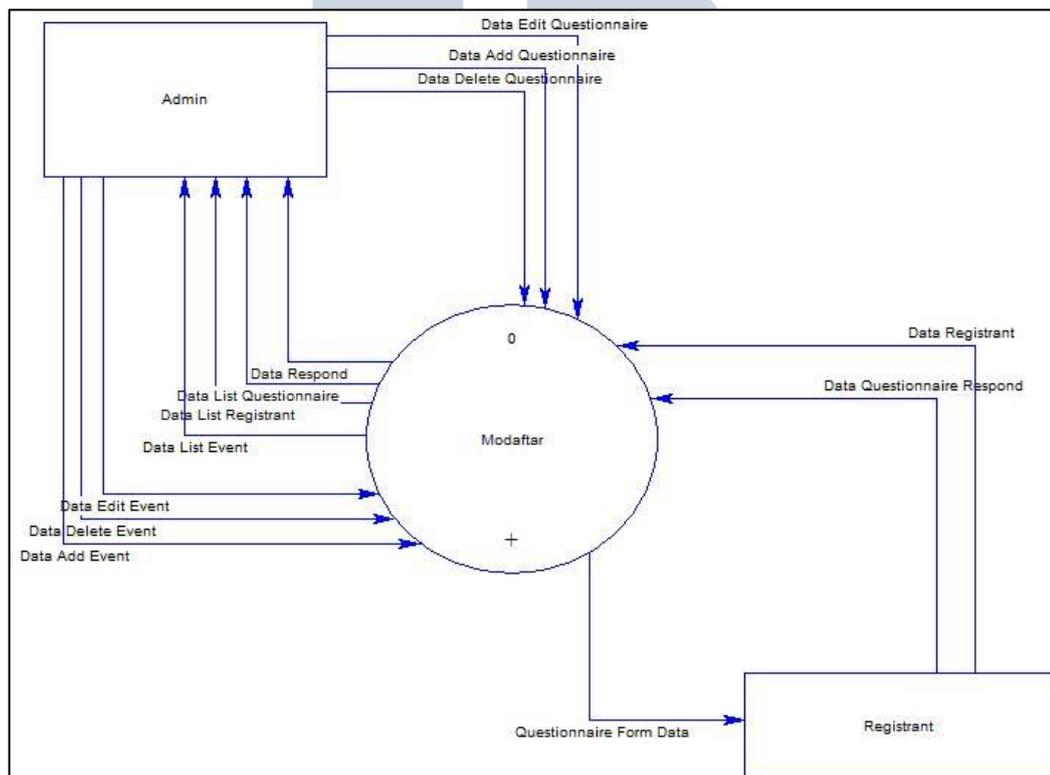


3.3.2 Perancangan

Berikut adalah uraian dari perancangan dalam membangun aplikasi Modaftar di PT Indosterling Technomedia.

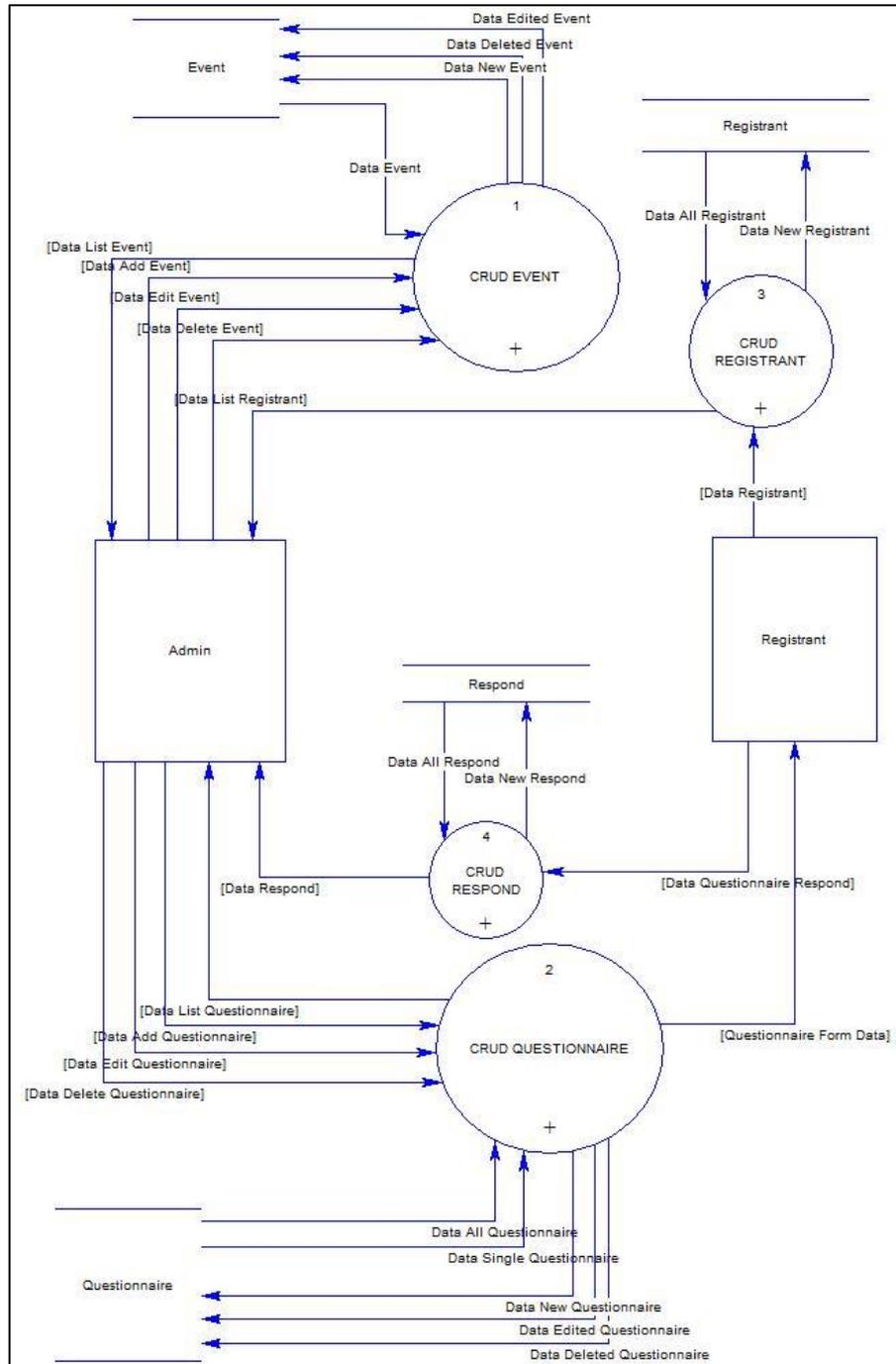
A. Data Flow Diagram (DFD)

Berikut adalah diagram aliran data (*Data Flow Diagram*) yang mendeskripsikan alur data pada saat sistem Modaftar dijalankan.



Gambar 3.1 DFD *Context Diagram* Modaftar

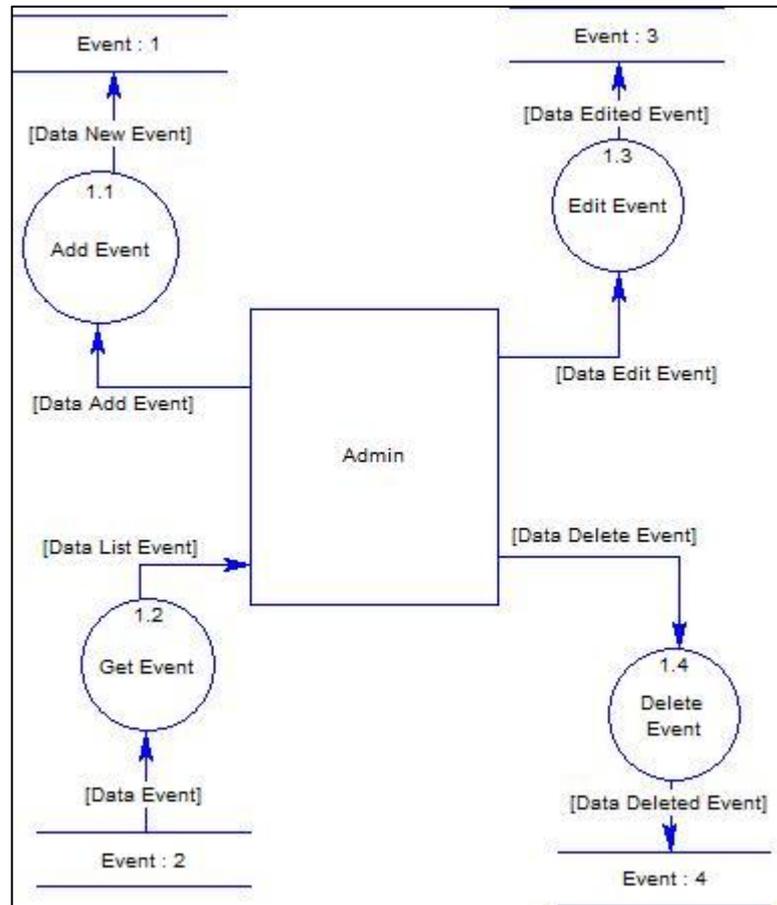
Pada Gambar 3.1 merupakan DFD *context diagram* dari modul admin Modaftar. Diagram ini menggambarkan alur-alur data pada modul admin secara keseluruhan. Pengguna pada modul admin adalah admin dan *registrant*. Admin dapat melakukan CRUD (*Create, Read, Update, and Delete*) pada *event* dan *questionnaire*. Data *questionnaire* yang dikirim admin akan diproses dan ditampilkan ke *registrant* dalam bentuk formulir yang dapat diisi.



Gambar 3.2 DFD Level 1 Modafatar

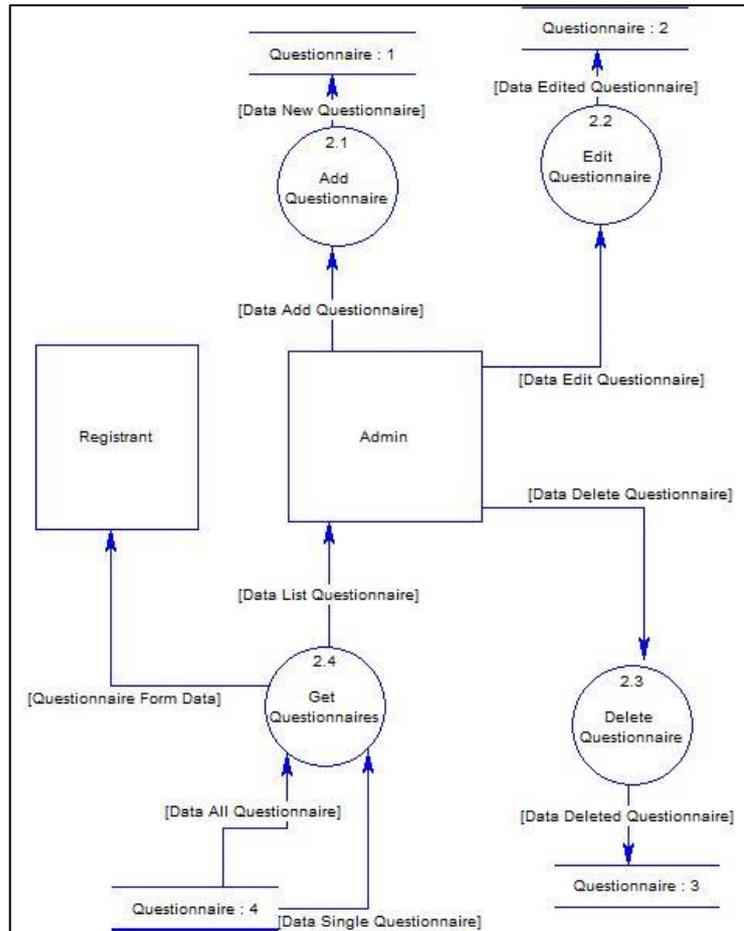
Pada Gambar 3.2 merupakan DFD level 1 modul admin Modafatar. DFD *level 1* menggambarkan 4 proses utama yang terdapat pada modul admin. Proses *CRUD event* berfungsi untuk membuat, melihat, mengubah, dan menghapus *event*. Proses *CRUD questionnaire* berfungsi untuk membuat, melihat, mengubah dan

menghapus *questionnaire*. Proses CRUD *registrant* berfungsi untuk membuat, melihat, dan menghapus *registrant*. Proses CRUD *respond* berfungsi untuk membuat, melihat, dan menghapus *respond*.



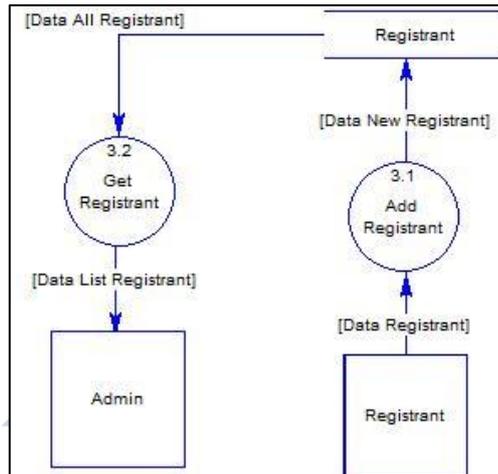
Gambar 3.3 DFD Level 2 Event

Pada Gambar 3.3 menggambarkan DFD level 2 dari proses *event*. Terdapat empat proses, yaitu: Add Event, Get Event, Edit Event, dan Delete Event. Pada proses Add Event dan Edit Event, data yang dimasukkan kedalam tabel Event adalah name, status, date_begin, dan date_end. Lalu pada proses Delete Event, admin memilih *event* yang akan dihapus dari *database*. Proses Get Event mengambil semua data *event* dari tabel Event.



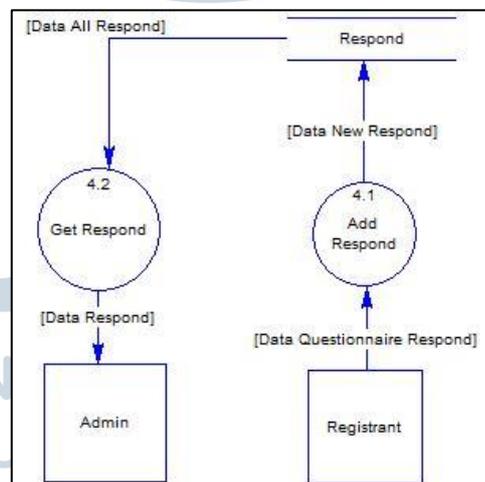
Gambar 3.4 DFD Level 2 Questionnaire

Pada Gambar 3.4 menggambarkan proses DFD level 2 dari proses *questionnaire* admin. Terdapat empat proses, yaitu: Add Questionnaire, Get Questionnaire, Edit Questionnaire, dan Delete Questionnaire. Pada proses Add Questionnaire dan Edit Questionnaire, data yang dimasukkan kedalam tabel Questionnaire adalah name, status, dan jsondata. Lalu pada proses Delete Questionnaire, admin memilih *questionnaire* yang akan dihapus dari *database*. Proses Get Questionnaire mengambil semua data *questionnaire* dari tabel Questionnaire.



Gambar 3.5 DFD Level 2 Registrant

Pada Gambar 3.5 menggambarkan DFD level 2 dari proses Registrant. Proses Get Registrant mengambil name, event_id, id_regist, birthdate, gender, email, phone, dan imgpath dari tabel Registrant. Pada proses Add Registrant, data yang dimasukkan ke tabel Registrant adalah name, event_id, id_regist, birthdate, gender, email, phone, dan imgpath.

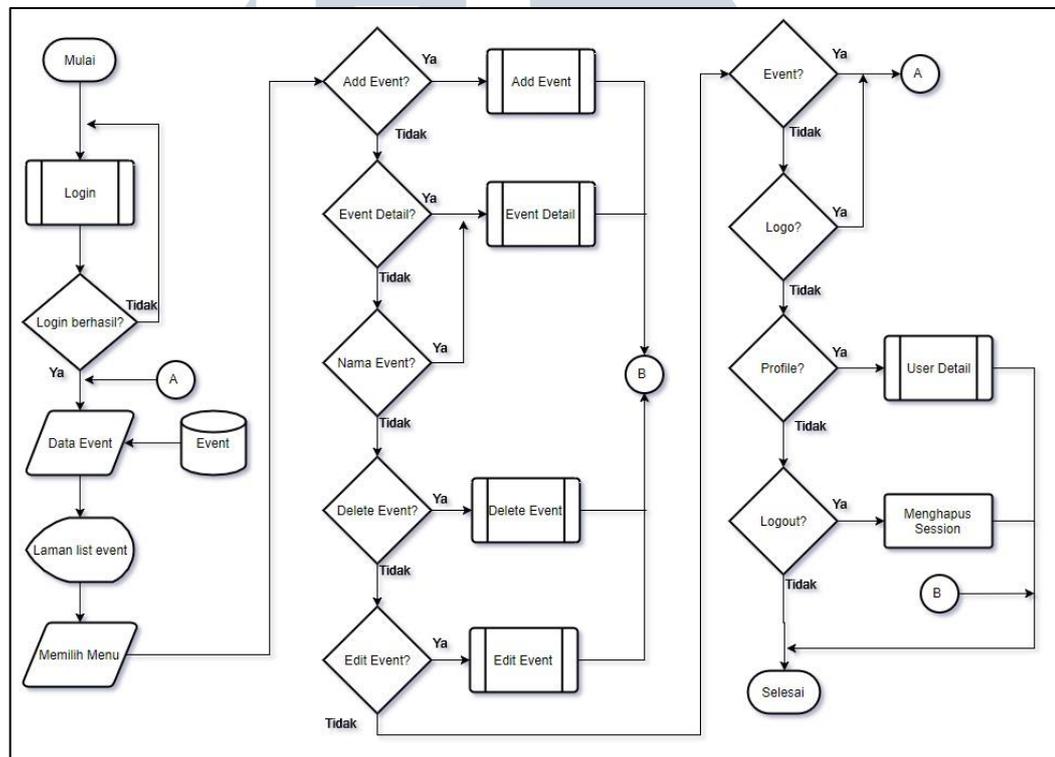


Gambar 3.6 DFD Level 2 Respond

Pada Gambar 3.6 menggambarkan DFD level 2 dari proses Respond. Proses Get Respond mengambil question_id, registrant_id, answer dan choice_id dari tabel Respond. Pada proses Add Respond, data yang dimasukkan ke tabel Respond adalah question_id, registrant_id, answer dan choice_id.

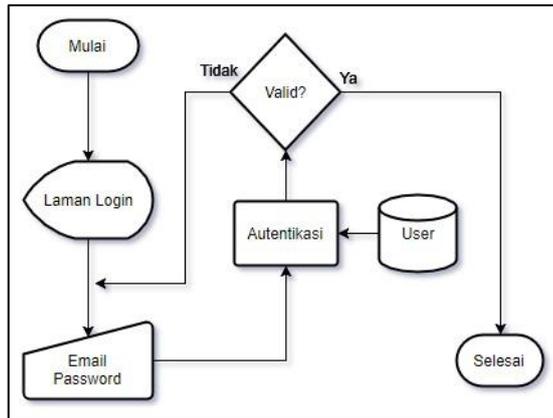
B. Flowchart

Berikut adalah *flowchart* modul admin yang dapat dipakai untuk menjabarkan proses-proses yang terdapat pada *website* Modaftar. Khusus untuk modul List Registrant, View Questionnaire, User Detail, dan Respond Questionnaire tidak akan dijabarkan dalam *flowchart* karena pengerjaan modul tersebut di luar dari tugas yang diberikan.



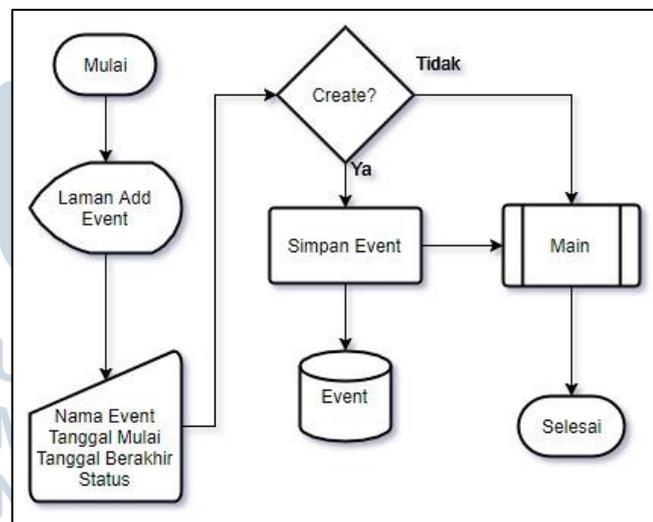
Gambar 3.7 Main Flowchart

Saat mengakses situs Modaftar, admin diharuskan untuk *login*. Setelah *login* berhasil admin bisa untuk menambahkan *event*, melihat Event Detail, menghapus *event*, membuka User Detail, dan *logout*. Modul Login yang dilakukan dijabarkan dalam *flowchart* berikut.



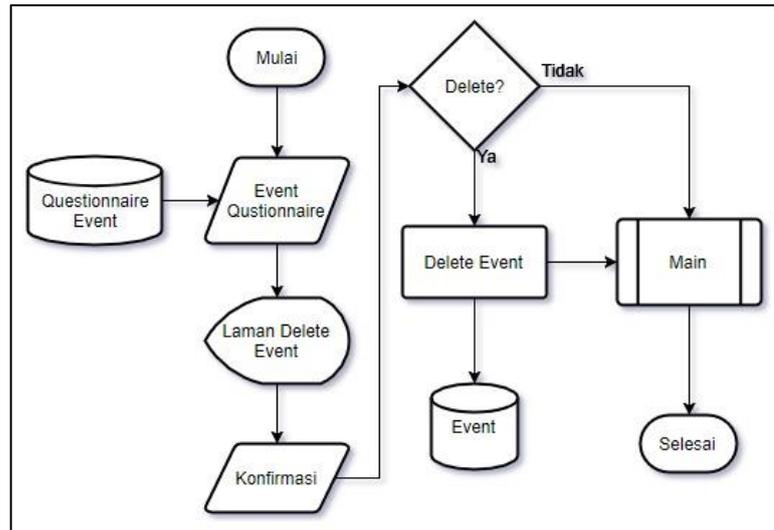
Gambar 3.8 Flowchart Login

Admin diminta untuk memasukkan *e-mail* dan *password* yang dimilikinya. Proses autentikasi menggunakan proses milik *framework* Django. Jika hasilnya valid maka selesai, jika hasil tidak valid maka pengguna diminta untuk memasukkan *e-mail* dan *password* kembali. Jika sudah masuk, admin dapat memilih beberapa menu yang berada di halaman utama. Proses Login digambarkan pada Gambar 3.8.



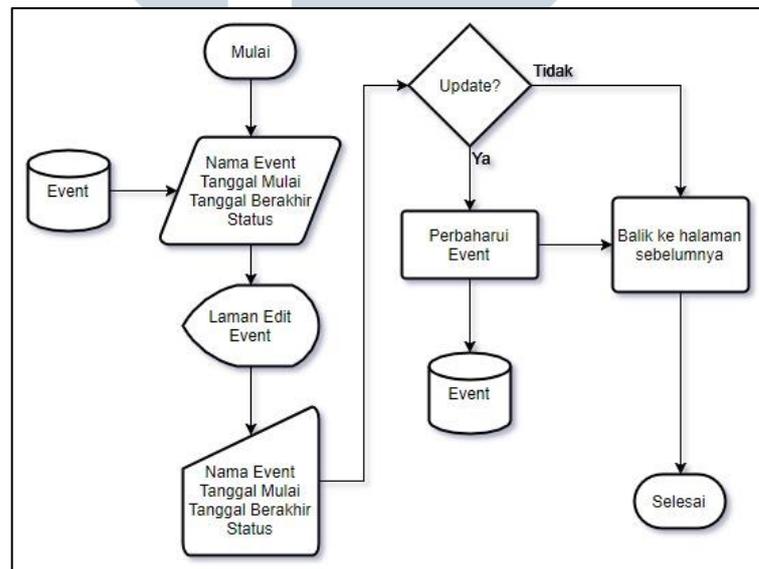
Gambar 3.9 Flowchart Add Event

Pada modul Add Event, admin diharuskan untuk mengisi formulir Add Event untuk memberikan informasi tentang *event* yang akan ditambahkan. Penyimpanan *event* dilakukan saat admin menekan tombol *create*.



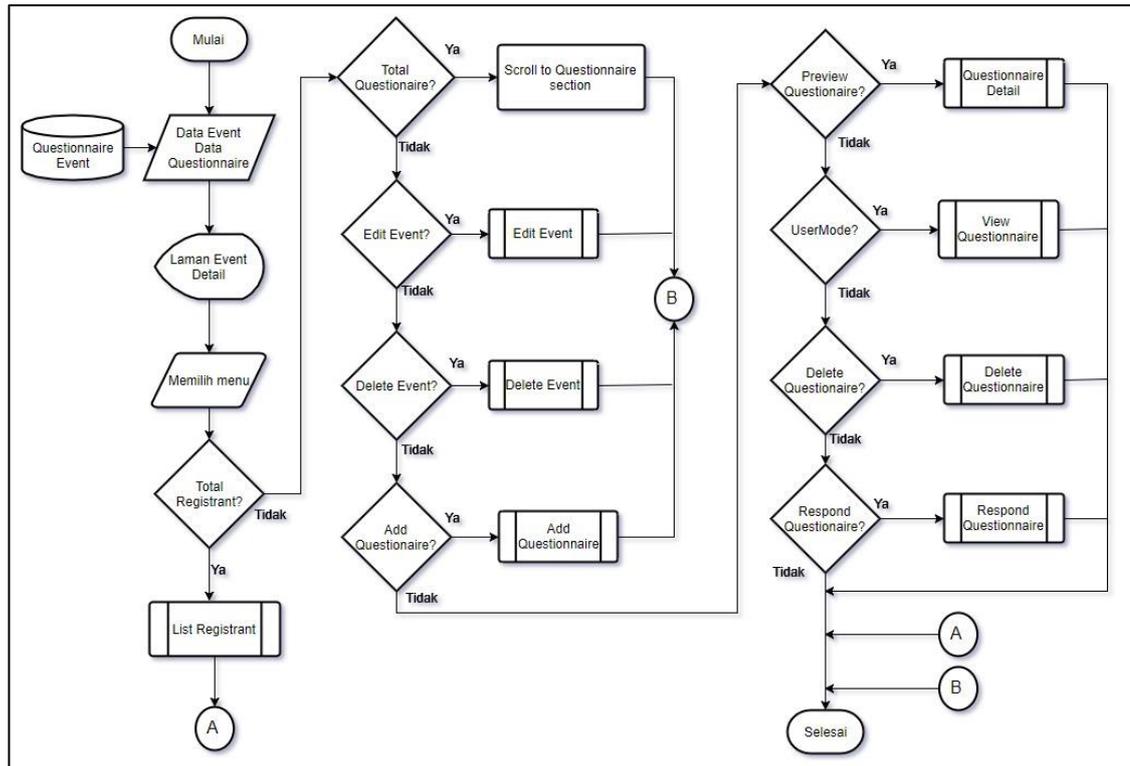
Gambar 3.10 Flowchart Delete Event

Pada modul Delete Event, admin dapat melihat informasi mengenai *event* yang akan dihapus dan admin diharuskan untuk mengonfirmasi dengan menekan tombol *delete* sebelum dapat menghapus *event* yang dipilih.



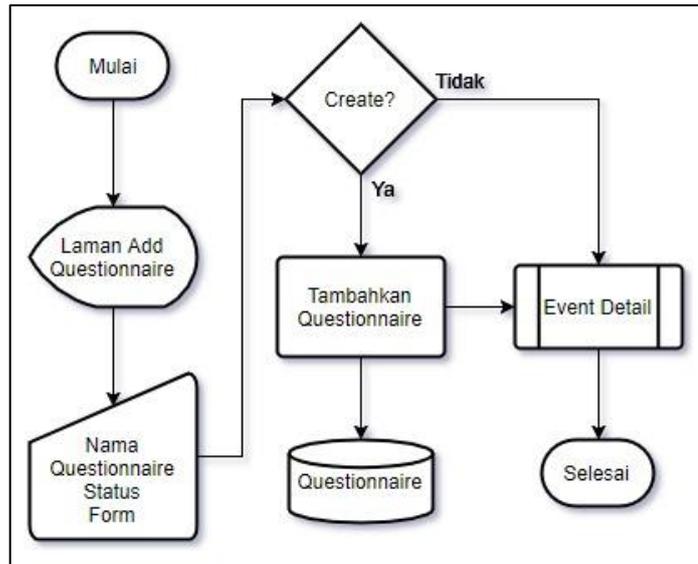
Gambar 3.11 Flowchart Edit Event

Pada modul Edit Event, admin dapat mengubah informasi *event* yang telah dibuat sebelumnya. Proses pengubahan data *event* akan dieksekusi setelah admin melakukan konfirmasi dengan menekan tombol *create*. Proses mengubah *event* dijabarkan pada Gambar 3.11.



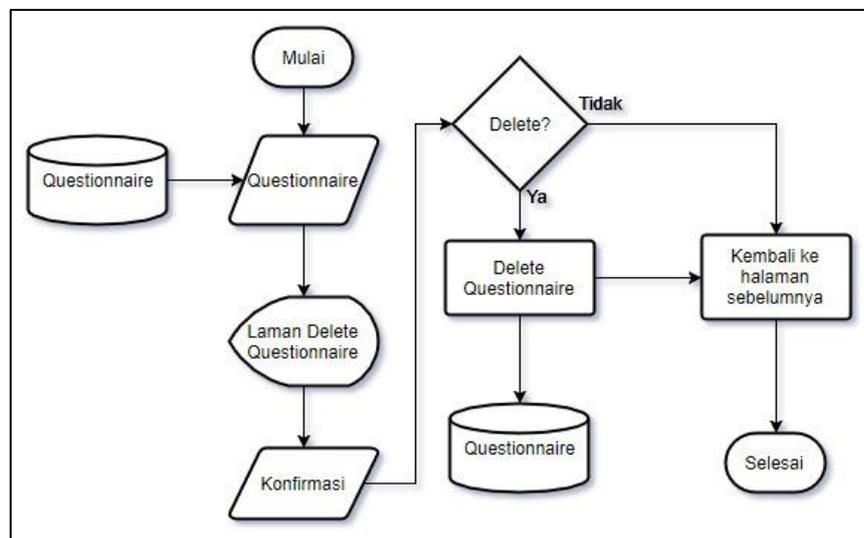
Gambar 3.12 Event Detail

Pada modul Event Detail, terdapat tiga *list* dibagian atas yang memberikan informasi jumlah *registrant* yang sudah mendaftar, jumlah *questionnaire* yang sudah dibuat di dalam *event* tersebut, dan jumlah *questionnaire* yang aktif. Di bagian tengah terdapat elemen jumbotron yang berisi informasi mengenai *event* yang sedang dibuka dan di bagian bawah jumbotron terdapat tombol untuk mengubah *event* dan tombol untuk menghapus *event*. Di bagian bawah halaman terdapat tabel *questionnaire*. Pada tabel *questionnaire* admin dapat menambahkan *questionnaire*, menghapus *questionnaire*, melihat *questionnaire*, melihat respon *questionnaire* milik *registrant*, dan berpindah ke *user mode*. Proses Event Detail dijabarkan pada Gambar 3.12



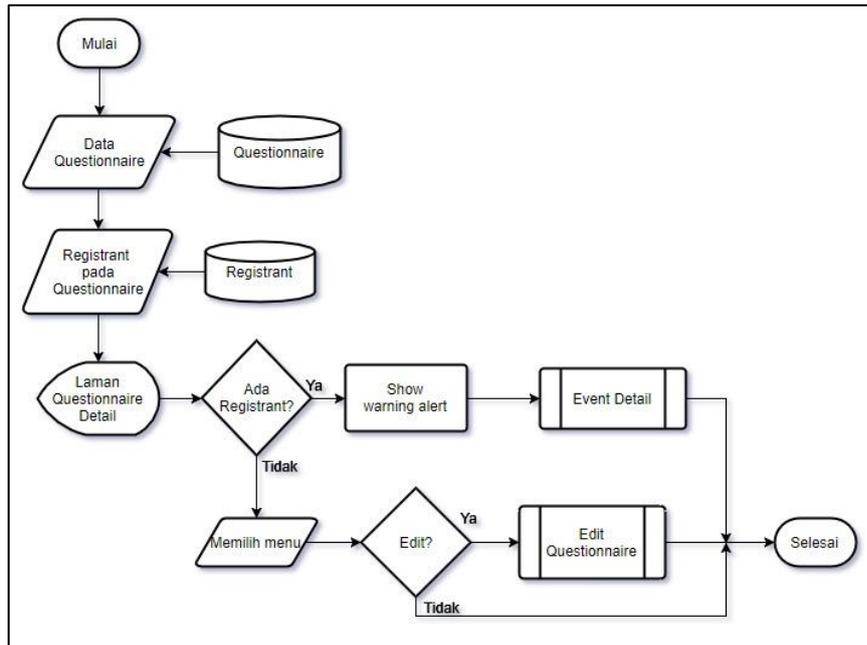
Gambar 3.13 Flowchart Add Questionnaire

Pada modul Add Questionnaire admin dapat membuat formulir dengan *library* yang telah disediakan. Pembuatan formulir menggunakan metode *drag-n-drop* dan admin diharuskan untuk memasukkan nama *questionnaire* dan *status questionnaire* untuk membuat *questionnaire*.



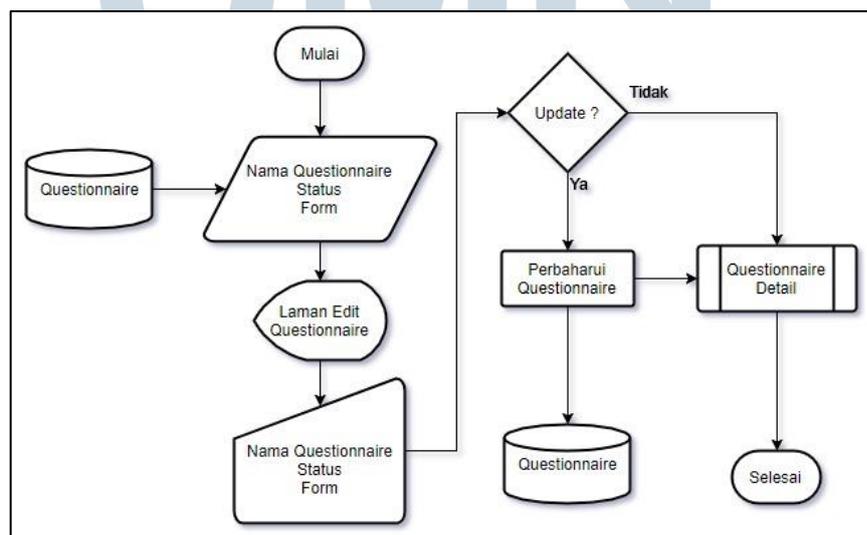
Gambar 3.14 Flowchart Delete Questionnaire

Pada modul Delete Questionnaire admin harus melakukan konfirmasi ulang sebelum dapat menghapus *questionnaire*. Proses Delete Questionnaire dijabarkan pada Gambar 3.14.



Gambar 3.15 Flowchart Questionnaire Detail

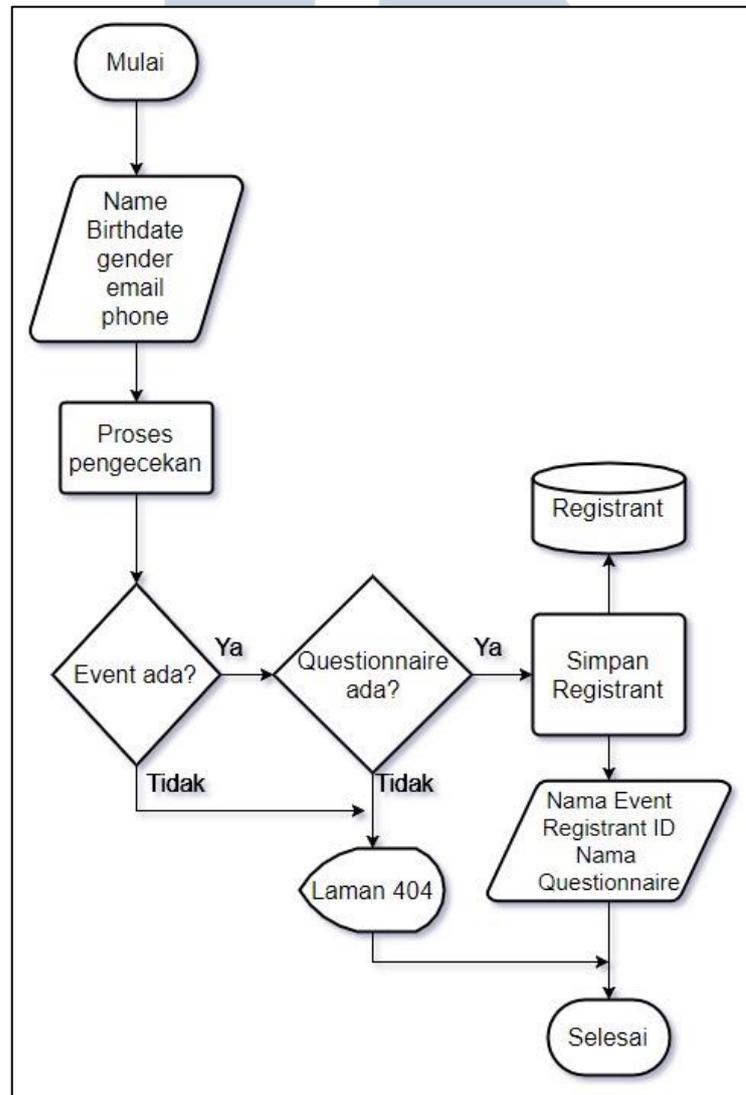
Pada modul Questionnaire Detail, admin dapat melihat *questionnaire* atau formulir yang telah dibuat sebelumnya. Jika pada *questionnaire* tersebut sudah terdapat *registrant* yang terdaftar, maka akan ditampilkan peringatan bahwa *questionnaire* tidak dapat diubah dan admin tidak dapat mengubah *questionnaire* atau formulir. Jika belum ada *registrant* yang terdaftar, maka admin dapat mengubah *questionnaire* atau formulir. Proses ini dijabarkan pada Gambar 3.15.



Gambar 3.16 Flowchart Edit Questionnaire

Pada modul Edit Questionnaire admin dapat mengubah kembali *questionnaire* yang sudah dibuat sebelumnya. Proses perubahan akan dieksekusi setelah admin melakukan konfirmasi dengan menekan tombol *update*. Proses Edit Questionnaire dijabarkan pada Gambar 3.16.

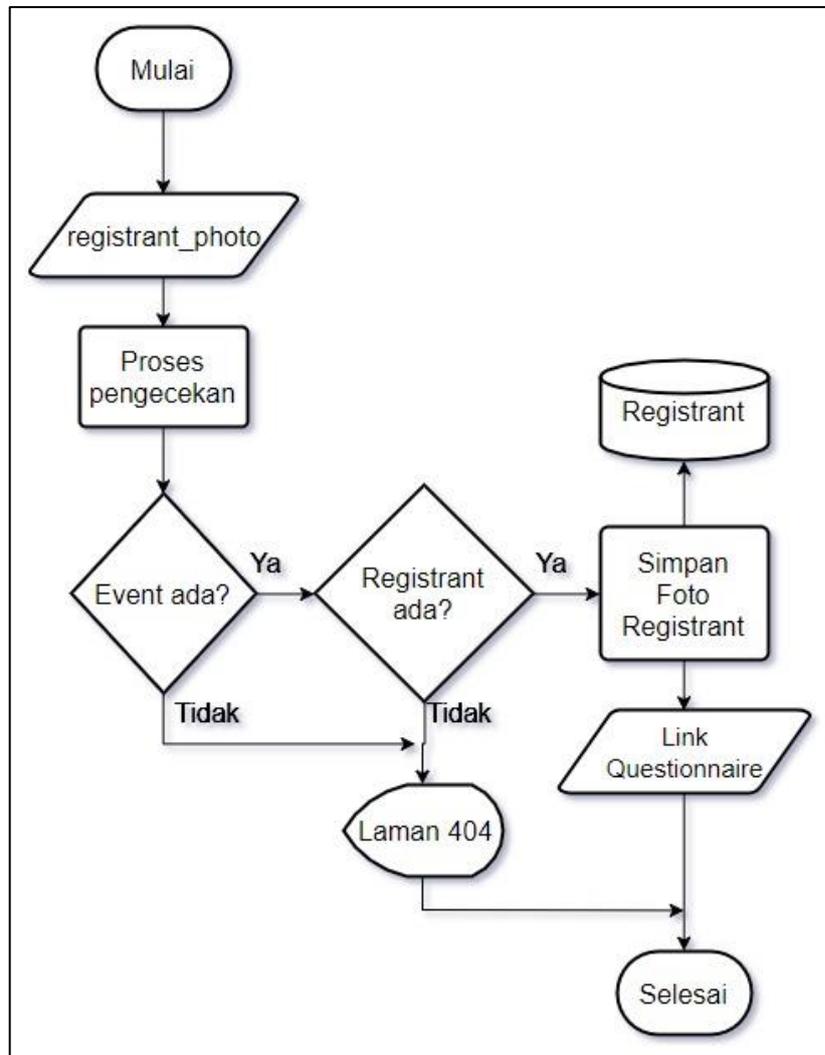
Berikut adalah *flowchart* modul API yang dipakai untuk menjabarkan proses-proses yang terdapat pada aplikasi Modafatar.



Gambar 3.17 Flowchart Registrant API

Pada Gambar 3.17 menggambarkan modul dari Registrant API. Modul Registrant API berfungsi untuk menerima data yang dikirim dalam bentuk POST.

Data yang diterima akan dicek, jika semua data valid, maka data akan disimpan dan API akan mengembalikan data berbentuk JSON.

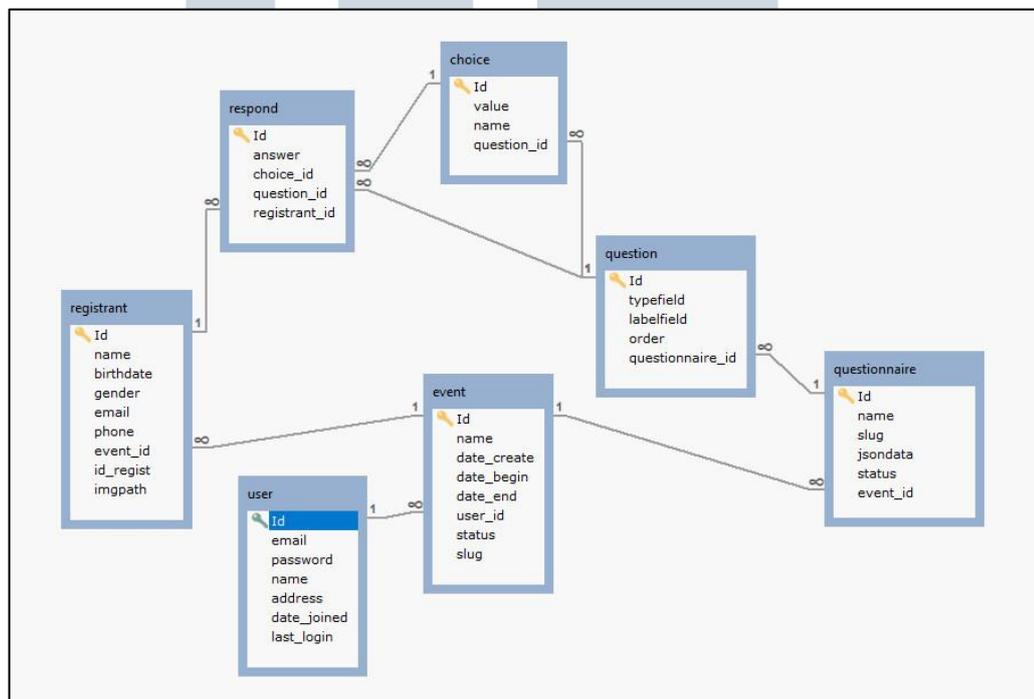


Gambar 3.18 Flowchart Registrant Image API

Pada Gambar 3.18 menggambarkan modul dari Registrant Image API. Modul Registrant Image API berfungsi untuk menerima file foto atau gambar yang dikirim. Data yang diterima akan dicek, jika semua data valid, maka data akan disimpan dan API akan mengembalikan data berbentuk JSON.

C. Database Schema

Pada Gambar 3.19 menggambarkan *database schema* dari aplikasi Modaftar. Aplikasi Modaftar memiliki 7 tabel utama yang saling berelasi satu sama lainnya. Tabel User berelasi dengan tabel Event, Tabel Event berelasi dengan tabel Questionnaire dan tabel Registrant, tabel Questionnaire berelasi dengan tabel Question, tabel Question berelasi dengan tabel Choice dan tabel Respond, tabel Choice berelasi dengan tabel Respond, dan tabel Registrant berelasi dengan tabel Respond.



Gambar 3.19 Database Schema Modaftar

D. Struktur Tabel

Tabel 3.2 menjabarkan isi dari tabel User, tabel User adalah tabel yang menyimpan data admin.

Tabel 3.2 Struktur Tabel User

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
email	varchar(254)	
password	varchar(128)	
name	varchar(255)	
address	varchar(255)	
date_joined	timestamp(6)	
last_login	timestamp(6)	

Tabel 3.3 menjabarkan isi dari tabel Event, tabel Event menyimpan data dari setiap *event* yang dibuat oleh admin.

Tabel 3.3 Struktur Tabel Event

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
name	varchar(100)	
date_create	date	
date_begin	date	
date_end	date	
user_id	integer	
status	verchar(100)	
slug	varchar(50)	

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Tabel 3.4 menjabarkan isi dari tabel Registrant, tabel Registrant menyimpan data dari setiap pendaftar pada setiap Event yang dibuat oleh admin.

Tabel 3.4 Struktur Tabel Registrant

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
name	varchar(100)	
birthdate	date	
gender	varchar(2)	
email	varchar(50)	
phone	varchar(15)	
event_id	integer	
id_regist	varchar(50)	
imgpath	varchar(100)	

Tabel 3.5 menjabarkan isi dari tabel Questionnaire, tabel Questionnaire menyimpan data formulir yang dibuat oleh admin.

Tabel 3.5 Struktur Tabel Questionnaire

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
name	varchar(100)	
slug	varchar(50)	
jsondata	text	
status	varchar(100)	
event_id	integer	

Tabel 3.6 menjabarkan isi dari tabel Question, tabel Question menyimpan setiap pertanyaan yang ada pada formulir.

Tabel 3.6 Struktur Tabel Question

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
typefield	varchar(100)	
labelfield	varchar(100)	
order	integer	
questionnaire_id	integer	

Tabel 3.7 menjabarkan isi dari tabel Choice, tabel Choice menyimpan setiap pilihan pada pertanyaan dengan tipe jawaban berganda seperti *checkbox* dan *radio button*.

Tabel 3.7 Struktur Tabel Choice

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
value	varchar(100)	
name	varchar(100)	
question_id	integer	

Tabel 3.8 menjabarkan isi dari tabel Respond, tabel Respond menyimpan setiap jawaban yang dijawab oleh *registrant* pada sebuah *questionnaire*.

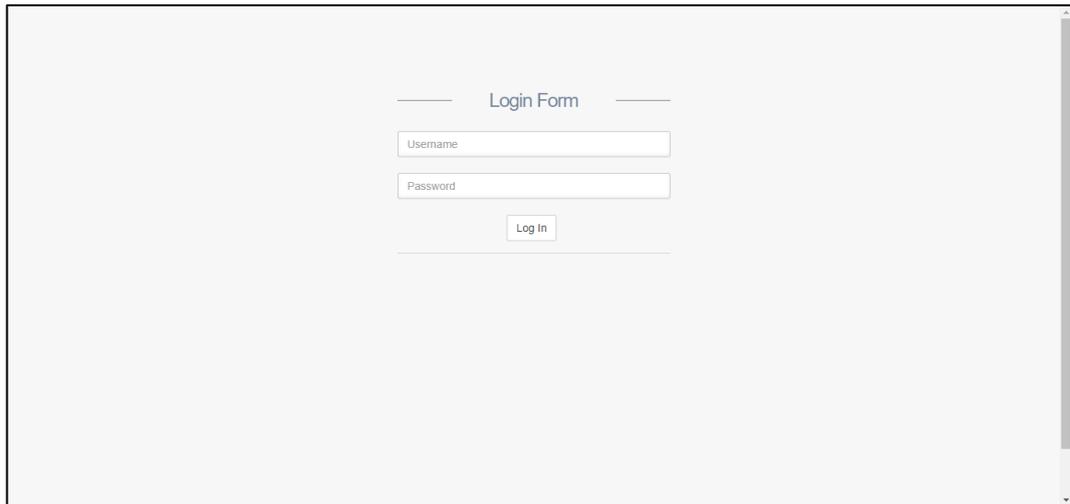
Tabel 3.8 Struktur Tabel Respond

Nama Kolom	Tipe Data	Keterangan
id	serial	Not Null, Auto Increment
answer	varchar(100)	
choice_id	integer	
question_id	integer	
registrant_id	integer	

E. Implementasi

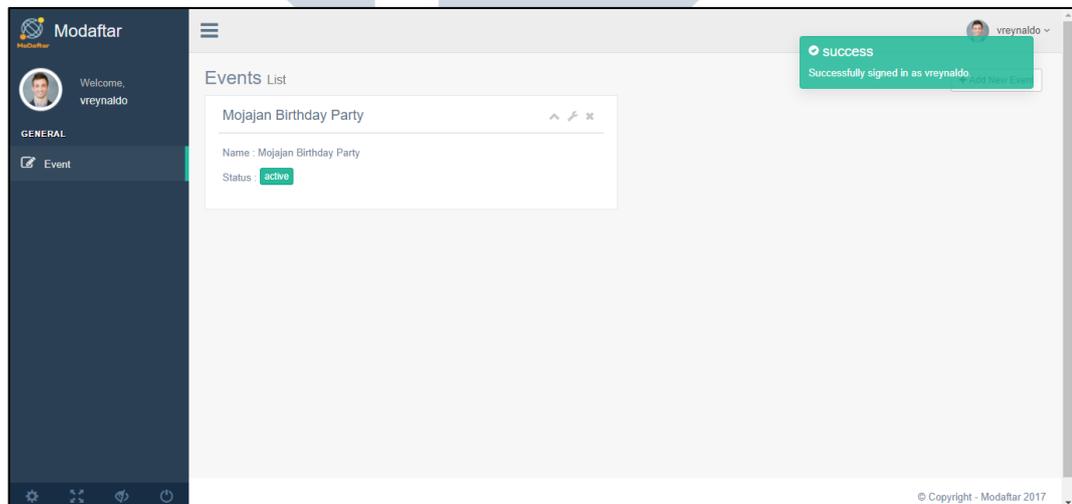
Berikut merupakan implementasi dari modul admin pada aplikasi Modaftar. Modul Login, Add Event, Event Detail, Delete Event, Edit Event, Add Questionnaire, Delete Questionnaire, Edit Questionnaire, dan Questionnaire Detail yang dibuat berdasarkan perancangan yang diberikan.

Pada Gambar 3.20 merupakan halaman Login dari aplikasi Modaftar. Setelah berhasil *login*, admin akan dibawa ke halaman utama seperti pada Gambar 3.21. Di halaman utama admin dapat melihat daftar *event* yang sudah pernah dibuat sebelumnya. Jika nama *event* diklik maka admin akan berpindah ke halaman Event Detail seperti pada Gambar 3.22.



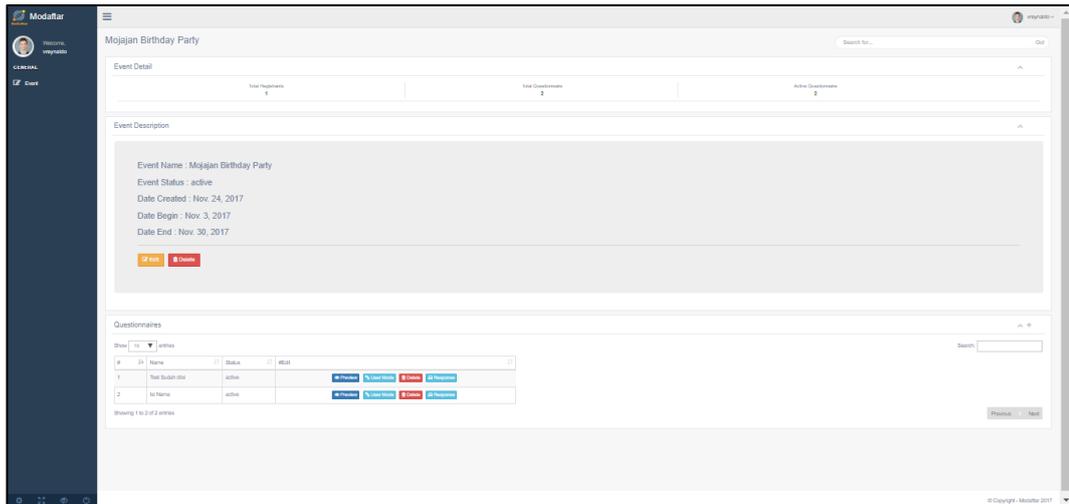
Gambar 3.20 Tampilan Login

Pada halaman Login, admin diminta untuk memasukkan *username* dan *password* yang dimiliki. Jika *login* berhasil, maka akan menampilkan halaman Main Menu seperti pada Gambar 3.21.



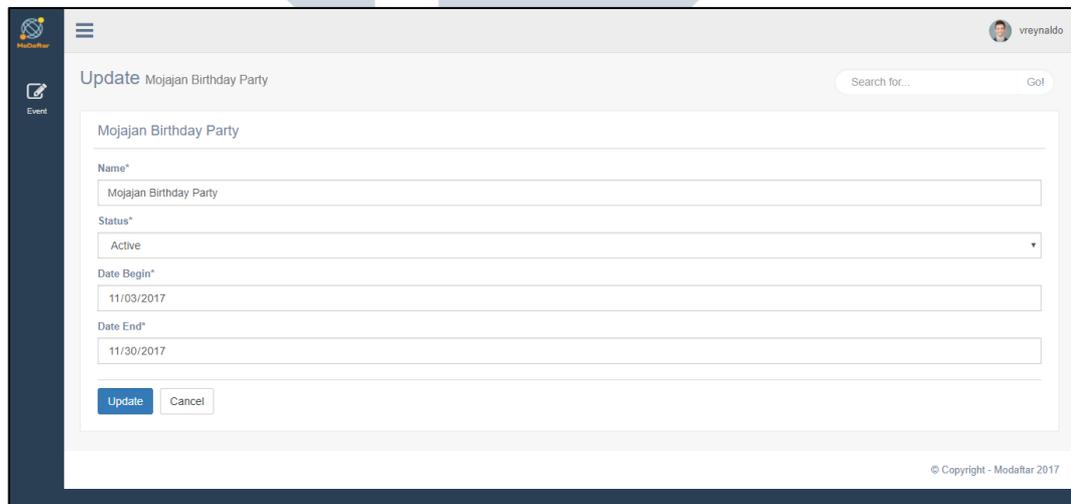
Gambar 3.21 Tampilan Main Menu

Pada halaman Main Menu, aplikasi akan menampilkan daftar *event* yang sudah dibuat sebelumnya dalam bentuk kartu. Aplikasi akan menampilkan *alert* berwarna hijau dipojok kanan atas saat admin selesai melakukan *login*.



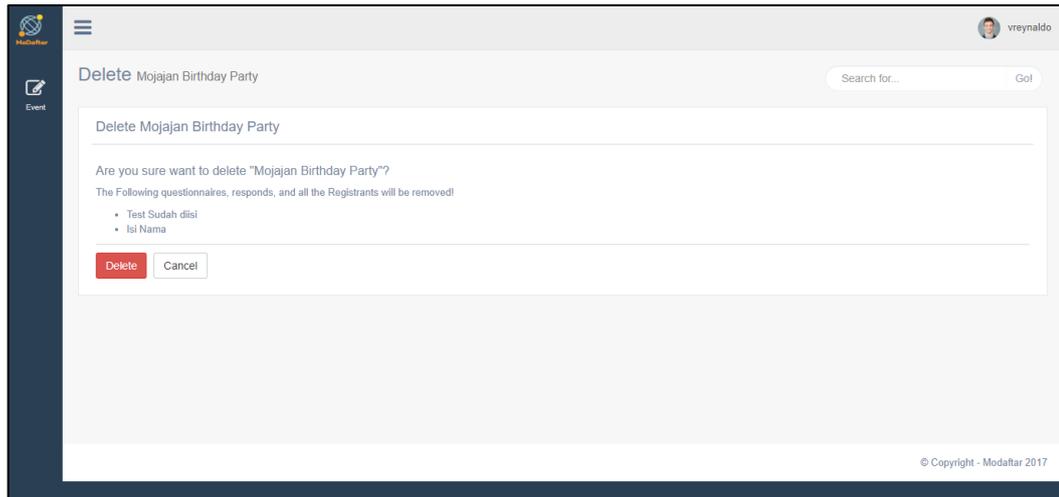
Gambar 3.22 Tampilan Event Detail

Pada halaman Event Detail, admin dapat melihat informasi seputar *event* yang sedang dibuka, jumlah *registrant* yang sudah mendaftar, jumlah *questionnaire* yang sudah dibuat, dan daftar dari *questionnaire*.



Gambar 3.23 Tampilan Edit Event

Pada Gambar 3.23 menggambarkan tampilan admin untuk mengubah informasi *event* yang telah dibuat sebelumnya. Pada halaman Edit Event, aplikasi menampilkan data *event* yang akan diubah. Setelah admin selesai mengubah data *event*, admin harus melakukan konfirmasi dengan menekan tombol *update* untuk menyimpan data *event* yang telah diubah ke dalam *database*.



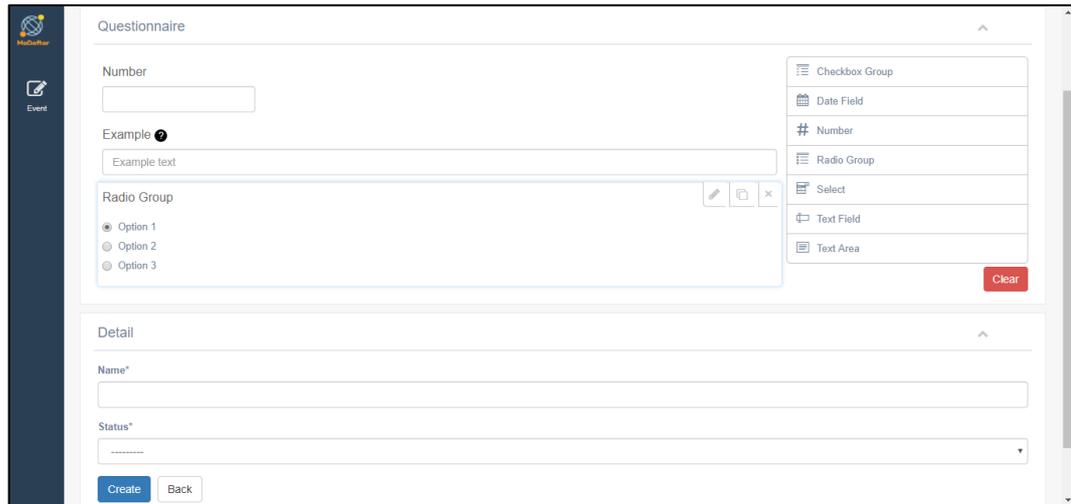
Gambar 3.24 Tampilan Delete Event

Gambar 3.24 merupakan halaman saat admin hendak menghapus sebuah *event*. Di halaman tersebut terdapat informasi seputar nama *event* dan nama *questionnaire* yang berada di dalam *event* yang ingin dihapus. Admin diharuskan melakukan konfirmasi dengan menekan tombol *delete* untuk menghapus *event* tersebut dari *database*.



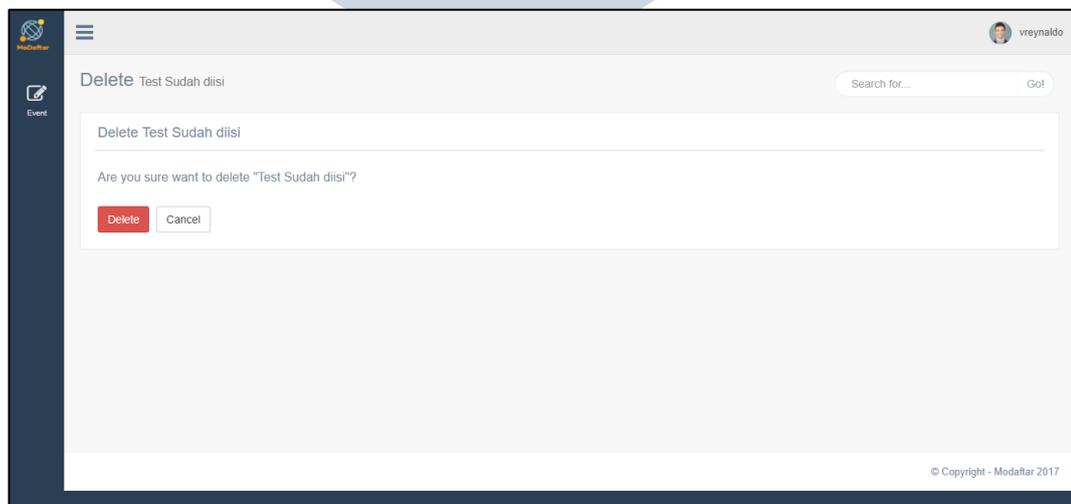
Gambar 3.25 Tampilan Add Event

Pada Gambar 3.25 merupakan halaman Add Event yang berguna untuk menambahkan *event* baru. Admin diminta untuk mengisi semua *field* yang tersedia. Jika sudah selesai admin dapat menekan tombol *create* untuk menyimpannya.



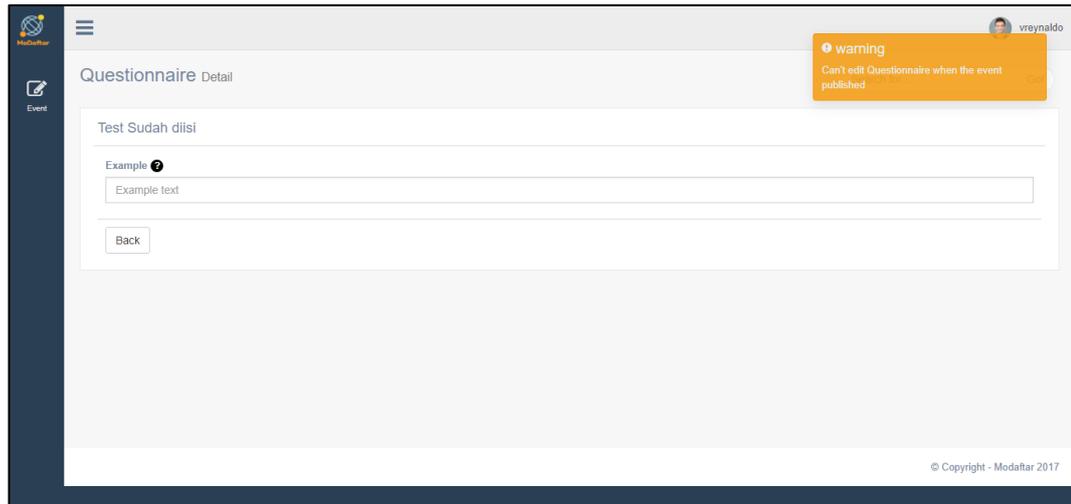
Gambar 3.26 Tampilan Add Questionnaire

Pada Gambar 3.26 menggambarkan tampilan pada saat admin ingin menambahkan *questionnaire*. Formulir dibuat menggunakan metode *drag-n-drop* dan hasilnya akan berupa objek JSON. Objek JSON tersebut yang nantinya akan disimpan di dalam *database*.



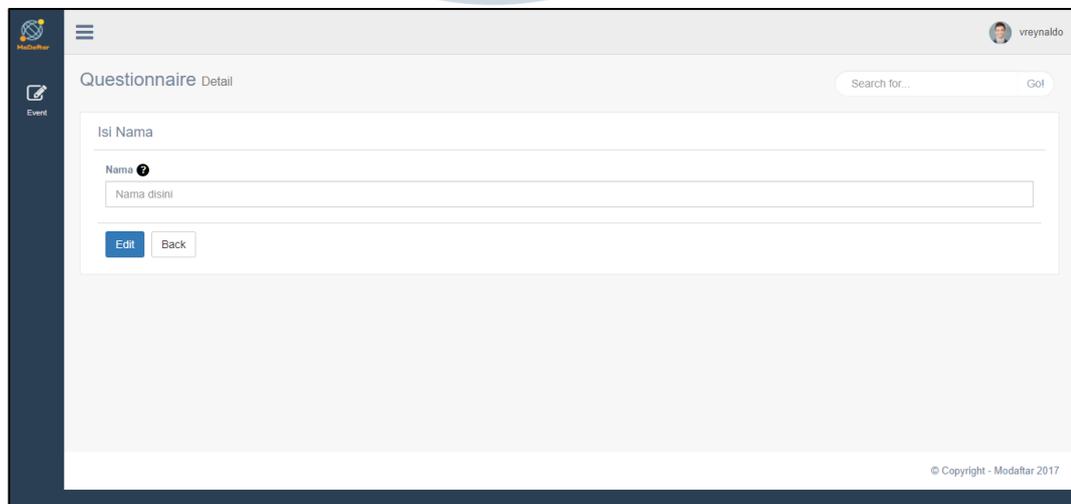
Gambar 3.27 Tampilan Delete Questionnaire

Gambar 3.27 merupakan halaman saat admin hendak menghapus sebuah *questionnaire*. Di halaman tersebut terdapat informasi nama *questionnaire* yang ingin dihapus. Admin diharuskan melakukan konfirmasi dengan menekan tombol *delete* untuk menghapus *questionnaire* tersebut dari *database*.



Gambar 3.28 Tampilan Questionnaire Detail

Pada Gambar 3.28 menggambarkan tampilan pada saat admin melihat *detail* dari *questionnaire* yang dibuat sebelumnya. Kotak peringatan seperti pada gambar akan keluar jika *questionnaire* sudah diisi oleh *registrant* dan admin tidak dapat mengedit *questionnaire*.



Gambar 3.29 Tampilan Questionnaire Detail dengan *edit*

Pada Gambar 3.29 menggambarkan tampilan saat admin melihat *detail* dari *questionnaire*. Tombol *edit* akan keluar jika belum ada *registrant* yang mengisi *questionnaire* tersebut.



Gambar 3.30 Tampilan Edit Questionnaire

Gambar 3.30 menggambarkan tampilan admin untuk mengubah informasi *questionnaire* yang telah dibuat sebelumnya. Pada halaman Edit Questionnaire, aplikasi menampilkan data *questionnaire* yang akan diubah. Setelah admin selesai mengubah data *questionnaire*, admin harus melakukan konfirmasi dengan menekan tombol *update* untuk menyimpan data *questionnaire* yang telah diubah ke dalam *database*.

3.3.3 Kendala dan Solusi yang Ditemukan

Selama menjalani praktek kerja magang di PT Indosterling Technomedia, kendala yang dialami berupa keterbatasan pengetahuan pada bahasa pemrograman Python pada pembuatan aplikasi Modaftar serta *style guide* yang biasa digunakan pada bahasa pemrograman Python. Bahasa pemrograman yang digunakan adalah Python dan *web framework* Django. Kendala di atas membuat pembangunan aplikasi menjadi sedikit terhambat. Solusi dari kendala yang terjadi adalah melakukan pencarian dan belajar mengenai informasi-informasi terkait serta memperbanyak diskusi dengan pembimbing lapangan di PT Indosterling Technomedia selama proses pembangunan aplikasi Modaftar.