



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

Tugas magang diberikan oleh *Project manager* dan dalam pelaksanaannya, seluruh pekerjaan yang dilakukan dipantau dan dibimbing langsung oleh Yoas selaku *Project Manager*. *Project Manager* terlebih dahulu menjelaskan proyek yang akan dikerjakan beserta spesifikasi proyek tersebut. Dalam pelaksanaan kerja magang di PT Indosterling Technomedia, terdapat dua orang dari jurusan Teknik Informatika Universitas Multimedia Nusantara yang dibagi menjadi dua tim dengan tugas yang berbeda.

Selama program kerja magang berlangsung, bimbingan, arahan pantauan dari pengawas lapangan terus dilakukan untuk meningkatkan pemahaman mengenai proyek yang dibangun, dari fitur hingga struktur database sistem. Selama bekerja, *progress* harus selalu dilaporkan kepada *project manager*, sehingga apabila terdapat kesalahan atau hal yang tidak sesuai dengan spesifikasi, dapat langsung diperbaiki. Begitu pula jika terdapat pertanyaan-pertanyaan ataupun terjadi kesulitan saat pengerjaan, dapat langsung bertanya kepada *Project Manager*.

3.2 Tugas yang Dilakukan

Berikut merupakan rincian tugas yang dilakukan selama pelaksanaan kerja magang dalam waktu 45 hari di PT Indosterling Technomedia.

Tabel 3.1 Rincian Tugas Magang

Minggu Ke-	Pekerjaan yang Dilakukan
1	<ul style="list-style-type: none"> • Belajar bahasa pemrograman Python untuk <i>web</i>, <i>framework</i> Flask, dan <i>framework</i> Django • Menghadiri rapat untuk membahas <i>system requirements</i> dari aplikasi • Memulai proyek pembuatan <i>website</i> dengan jQuery Formbuilder dan <i>framework</i> Django
2	<ul style="list-style-type: none"> • Menambahkan penggunaan Cookiecutter pada proyek • Mengubah struktur aplikasi menjadi Class Based View
3	<ul style="list-style-type: none"> • Membuat fitur untuk berpindah dari mode admin ke mode <i>user</i> • Membuat fitur untuk menambahkan <i>registrant</i> baru
4	<ul style="list-style-type: none"> • Membuat fitur untuk mengambil foto <i>registrant</i> • Membuat fungsi untuk menampilkan kuesioner yang telah dibuat
5	<ul style="list-style-type: none"> • Membuat fitur untuk mengolah data kuesioner <i>registrant</i> dan menampilkannya dalam bentuk tabel dan grafik • Memperbaiki tampilan kuesioner saat ditampilkan
6	<ul style="list-style-type: none"> • Memperbaiki kesalahan pada fitur pengolahan data dan mengganti tampilan grafik • Membuat fungsi <i>page redirecting</i> setelah mengisi kuesioner
7	<ul style="list-style-type: none"> • Membuat fitur untuk menampilkan seluruh <i>registrant</i> dalam sebuah <i>event</i> • Menambahkan fitur agar <i>registrant</i> yang pernah mendaftar tidak perlu mendaftar ulang • Belajar membuat <i>unit test</i> pada aplikasi dengan <i>framework</i> Django
8	<ul style="list-style-type: none"> • Melakukan <i>testing</i> aplikasi • Memperbaiki <i>user interface</i> pada beberapa <i>page</i>
9	<ul style="list-style-type: none"> • Membuat dokumentasi secara <i>inline</i> pada <i>source code</i> • Melakukan demo aplikasi kepada <i>project manager</i>

Pada minggu pertama, tugas yang diberikan adalah belajar bahasa Python untuk pemrograman *website*, kemudian dilanjutkan dengan belajar *framework* membangun web, berupa Flask dan Django. Setelah cukup memahami bagaimana membuat *website* dengan Python, tugas selanjutnya merupakan menghadiri rapat dengan *project manager* untuk membahas *system requirement* dari *website* yang akan dibangun. Setelah selesai membahas seluruh kebutuhan sistem, *project* mulai

dikerjakan. Tugas berikutnya yang dilakukan adalah membuat *project* dengan Django dan mempelajari cara kerja dari jQuery FormBuilder.

Pada minggu kedua, terjadi perubahan struktur *website*. Perubahan struktur dirasakan penting untuk meningkatkan *readability* dan pemahaman pada kode sehingga proyek dimulai ulang dengan menggunakan *project template* Django Cookiecutter yang berfungsi untuk membuat aplikasi langsung menjadi *production-ready*. Struktur lain yang berubah adalah adanya penggunaan Class Based View yang merupakan sekumpulan dari *abstract class* dengan fitur-fitur umum dalam pengembangan *website* yang dapat di-*inherit* sesuai kebutuhan.

Pada minggu ketiga, modul *user* mulai dikerjakan. Fungsi pertama yang dikerjakan merupakan fungsi untuk berpindah dari mode *admin* ke mode *user*. Fungsi ini juga dilengkapi fitur *logout* secara otomatis untuk mencegah *user* yang tidak bertanggung jawab masuk ke dalam mode *admin*. Setelah fungsi tersebut selesai, pengerjaan dilanjutkan dengan membuat fitur untuk menambahkan *registrant* yang mendaftar melalui *form* yang telah disediakan.

Pada minggu keempat, pengerjaan mode *user* dilanjutkan pada tahap kedua. Setelah pendaftaran *registrant*, tugas yang dilakukan berikutnya adalah membuat fungsi untuk mengambil foto *registrant* yang mendaftar dengan pertimbangan aplikasi *website* dibuka dengan *browser* pada *handphone* dan pengambilan foto dengan kamera depan *handphone* tersebut. Kemudian, tahap terakhir dari pembuatan mode *user* ialah menampilkan kuesioner yang telah dibuat sebelumnya.

Pada minggu kelima, dibuat penambahan fungsi pada tahapan ketiga dari mode *user*, yaitu mengambil data yang diisi *registrant* pada kuesioner dilanjutkan

dengan fungsi untuk mengolah semua data yang masuk dalam satu kuesioner kemudian menampilkannya dalam bentuk tabel dan grafik yang dapat dilihat pada mode *admin*. Perbaikan *user interface* kuesioner juga dilakukan pada minggu kelima.

Pada minggu keenam, ditemukan kesalahan dalam pengolahan data yang mengakibatkan kesalahan pada laporan sehingga tugas yang dikerjakan hanya membenarkan kesalahan pengolahan data dan juga mengganti grafik yang digunakan untuk menampilkan data. *Page redirecting* setelah *registrant* mengisi kuesioner juga ditambahkan pada minggu ini.

Pada minggu ketujuh, ditambahkan fitur pada mode *user*, agar *registrant* yang pernah mendaftar pada sebuah *event* dan telah mengisi kuesioner tidak perlu mendaftar ulang, hanya memerlukan memasukkan nomor ID-nya saja. Tugas selanjutnya ialah membuat fungsi untuk menampilkan detail *registrant* yang pernah mengisi kuesioner pada satu *event*. Pada minggu ini juga diberikan tugas untuk mempelajari *unit testing* pada aplikasi Django.

Pada minggu kedelapan, pengujian aplikasi dilakukan untuk mencari *bug* pada aplikasi yang dikembangkan. Tugas yang lainnya adalah memperbaiki *user interface* pada modul *user*. Pada minggu kesembilan, tugas terakhir yang dilakukan ialah memberikan dokumentasi pada *source code* secara *inline* (menuliskan fungsi dan kegunaan dengan *comment*) dan melakukan demo aplikasi secara keseluruhan kepada *project manager*.

3.3 Uraian Pelaksanaan Kerja Magang

Proses pelaksanaan kerja magang diuraikan menjadi tiga bagian yaitu proses pelaksanaan, kendala yang ditemukan, dan solusi atas kendala yang ditemukan. Berikut merupakan uraian dari setiap bagian pelaksanaan kerja magang.

3.3.1 Proses pelaksanaan

Dalam pelaksanaannya, modul *user* dibuat berdasarkan *system requirement* yang dibuat saat proses perancangan. Kemudian alur kerja submodul pendaftaran *registrant* baru, pengambilan gambar *registrant*, menampilkan dan mengolah data kuesioner, membuat *reporting* dari kuesioner dan menampilkan seluruh *registrant* dalam satu kuesioner yang merupakan bagian dari modul *user* akan dijabarkan dalam *Data Flow Diagram (DFD)*, *Sitemap*, *Flowchart*, *Database Schema*, struktur tabel, dan rancangan antarmuka. Setelah itu implementasi perancangan akan dijabarkan dalam implementasi sistem.

A. System Requirement

Kebutuhan sistem dari Modafatar modul *user* akan dijabarkan pada *system requirement* berikut.

- User Requirement:
 - Admin: *Admin* dapat melihat data-data kuesioner dan *registrant*.
 - Registrant: Melakukan pengisian kuesioner.
- System Requirement:
 - *Registrant* dapat melakukan pendaftaran yang dibagi dua tahap, pengisian data diri dan sesi foto.
 - *Registrant* dapat melihat pertanyaan kuesioner dan mengisi jawabannya.

- *Admin* dapat melihat data diri dari setiap *registrant* yang mendaftar, data ditampilkan dalam bentuk tabel.
- *Admin* dapat melihat hasil atau data dari kuesioner dengan format tabel atau grafik jika diperlukan.

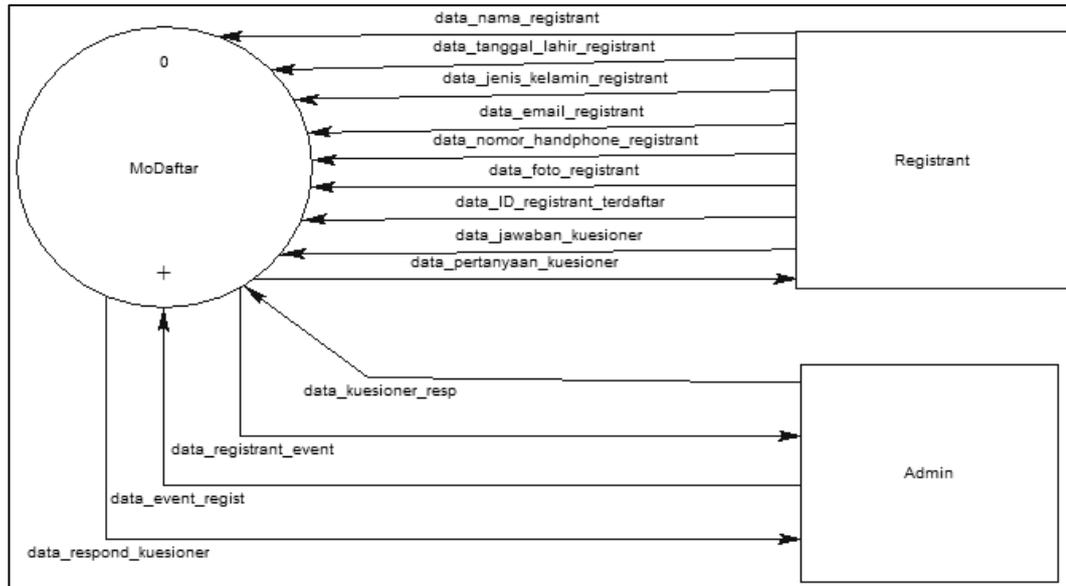
B. Data Flow Diagram

Data Flow Diagram adalah perangkat-perangkat analisis dan perancangan yang terstruktur sehingga memungkinkan penganalisis sistem memahami sistem dan subsistem secara visual sebagai suatu rangkaian aliran data yang saling berkaitan (Kendall & Kendall, 2008). Berikut merupakan Data Flow Diagram dari modul *user* Modaftar.

B.1 DFD Context Diagram

Gambar 3.1 menggambarkan DFD Context Diagram dari sistem Modaftar. Pada sistem ini terdapat dua entitas, yaitu Admin dan Registrant. Entitas Admin berperan sebagai *administrator* atau pengurus dari sebuah *event*. Admin dapat membuat sebuah *event* beserta kuesionernya. Selain itu, Admin juga dapat melihat daftar *registrant* dan jawaban kuesionernya. Entitas Registrant berperan sebagai pengunjung dari *event* yang hendak mengisi kuesioner pada sistem Modaftar.

U M N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



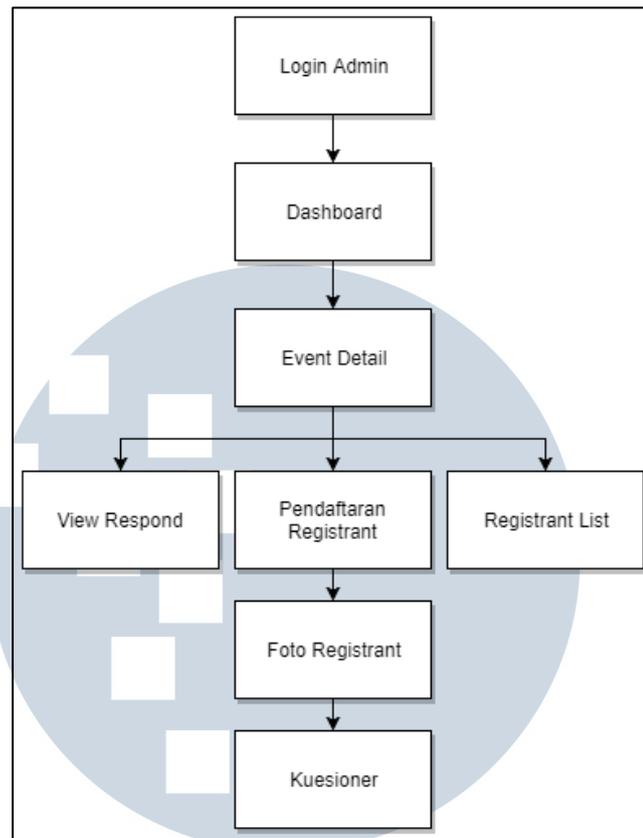
Gambar 3.1 DFD Context Diagram

B.2 DFD Level 1

Gambar 3.2 merupakan DFD level 1 dimana aliran data digambarkan secara rinci. Proses-proses dan tabel basis data yang digunakan juga digambarkan pada DFD level 1. Terdapat empat proses, yaitu *view_registrant_process*, *view_respond_process*, *add_new_registrant_process*, dan *answer_questionnaire_process*. Proses *view_registrant_process* mengambil data *registrant* dan data *event* dari tabel *event* dan *registrant* berdasarkan *event* yang dipilih *admin*. Lalu pada *view_respond_process*, *admin* memilih kuesioner dan *view_respond_process* mengambil data *respond* kuesioner tersebut dari tabel *Event*, *Registrant*, *Questionnaire*, *Choice*, dan *Answer*.

Pada *add_new_registrant_process*, proses mengambil data *event* dan *registrant*. Pada proses ini, *registrant* mengisi data diri berupa nama, nomor *handphone*, tanggal lahir, jenis kelamin, *e-mail*, dan foto. Jika *registrant* telah terdaftar sebelumnya, *registrant* hanya perlu memasukkan ID *registrant* miliknya.

C. Sitemap



Gambar 3.3 Sitemap Modafatar

Gambar 3.3 merupakan Sitemap dari Aplikasi Modafatar. Sitemap ini merupakan *sitemap* keseluruhan dari aplikasi Modafatar. Untuk modul *user*, hanya mencakup bagian View Respond, Registrant List, Pendaftaran Registrant, Foto Registrant, dan Kuesioner. *Sitemap* ini mengacu pada DFD Level 1 yang ada pada Gambar 3.2. View Respond mengacu pada proses 2. Untuk Registrant List, mengacu pada proses 1. Kemudian, Kuesioner mengacu proses 4, dan Pendaftaran Registrant, Foto Registrant mengacu pada proses 3.

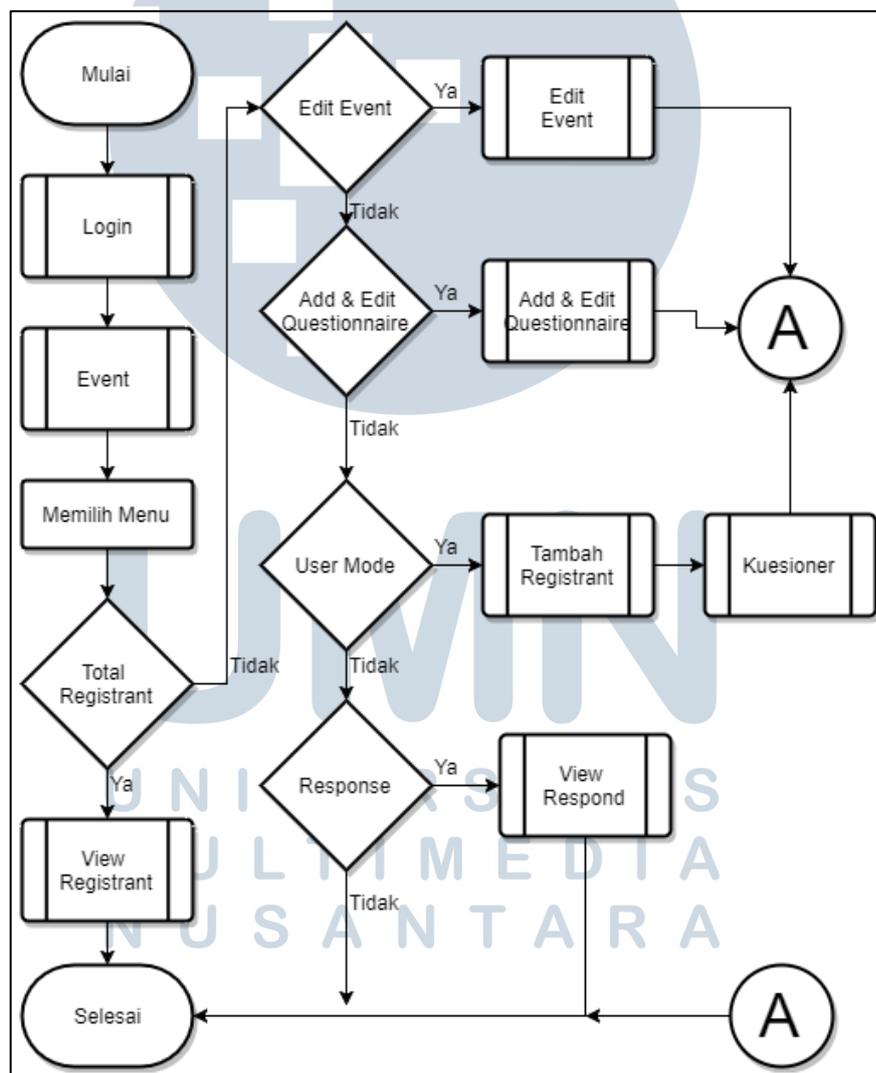
D. Flowchart

Flowchart atau bagan alur merupakan metode untuk menggambarkan tahap-tahap penyelesaian masalah (prosedur) data dengan simbol-simbol standar

yang mudah dipahami (Soeherman & Pinontoan, 2008). Berikut merupakan *flowchart* dari modul *user* Modaftar.

D.1 Flowchart Utama Aplikasi

Gambar 3.4 adalah Flowchart Utama dari aplikasi Modaftar. Pada *flowchart* ini, submodul Login, Event, Edit Event, Add & Edit Questionnaire tidak akan dijabarkan lebih rinci pada *flowchart* karena submodul tersebut berada diluar modul *user*.



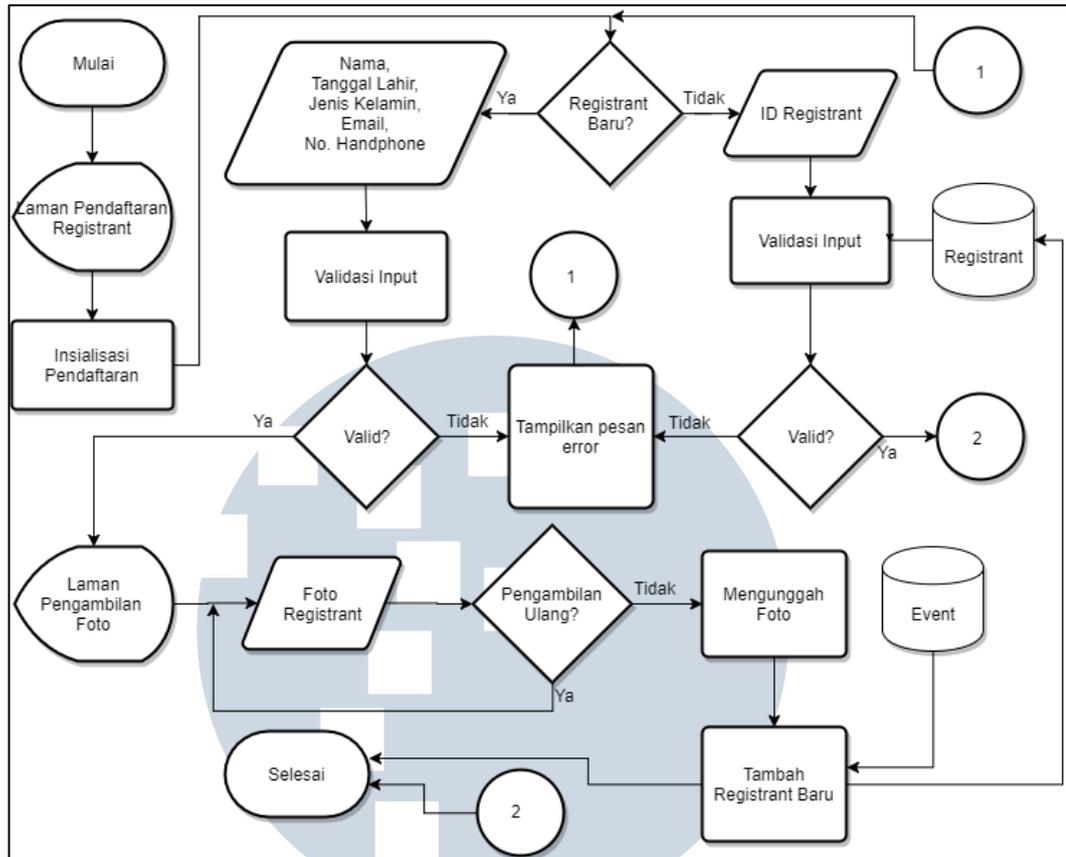
Gambar 3.4 Flowchart Utama Aplikasi

D.2 Flowchart Tambah Registrant

Gambar 3.5 adalah Flowchart Tambah Registrant. *Flowchart* ini mengacu ke proses 3 pada DFD Level 1. Pada saat membuka laman Pendaftaran Registrant, terdapat dua formulir yang harus diisi salah satunya oleh *registrant* yang bersangkutan. Jika *registrant* telah pernah mendaftarkan dirinya pada *event* tersebut dan hendak mengisi kuesioner, maka *registrant* akan mengisi ID-nya pada salah satu *form*. Sistem akan memvalidasi *input* dengan pengecekan ID telah terdaftar atau belum dan sudah pernah mengisi kuesioner atau belum. Jika ID telah terdaftar, *registrant* kemudian diarahkan pada laman Kuesioner. Sebaliknya, pesan *error* akan ditampilkan jika ID belum terdaftar atau sudah pernah mengisi kuesioner.

Namun, jika sebelumnya *registrant* tidak pernah mendaftar, pendaftaran dilakukan dengan memasukkan data diri berupa nama, tanggal lahir, jenis kelamin, *e-mail*, dan nomor *handphone*. Sistem akan melakukan validasi terhadap *input*, yaitu pengecekan tanggal lahir dan pengecekan format *e-mail*. Pendaftaran *registrant* baru akan dapat dilanjutkan ketika *input* tervalidasi, *registrant* akan diarahkan ke laman Pengambilan Foto jika sukses dan pesan *error* akan ditampilkan jika validasi gagal.

Pada laman Pengambilan Foto, sistem akan mengakses kamera dari *device* tersebut kemudian mengambil foto dari *registrant*. *Registrant* dapat mengambil ulang foto jika terdapat kesalahan pada foto. Setelah pengambilan foto, *registrant* akan didaftarkan pada sistem kemudian akan diarahkan pada laman Kuesioner.



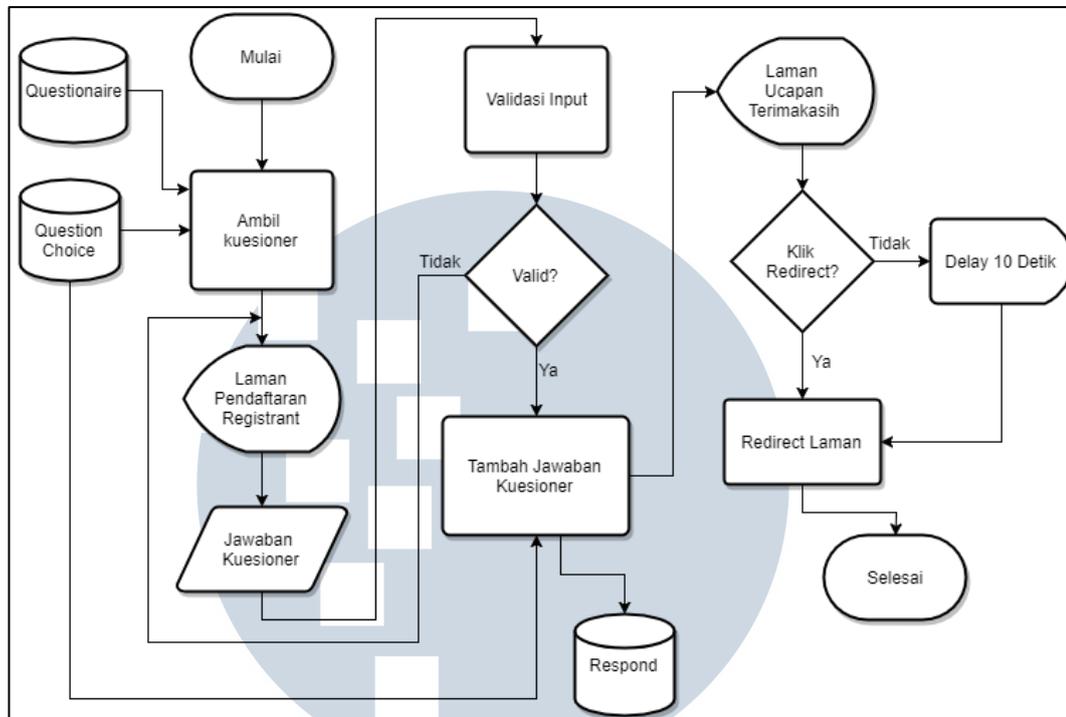
Gambar 3.5 Flowchart Tambah Registrant

D.3 Flowchart Kuesioner

Gambar 3.6 merupakan Flowchart Kuesioner. *Flowchart* ini mengacu pada proses 4 pada DFD Level 1. Laman kuesioner merupakan laman yang berisi kuesioner yang telah dibuat sebelumnya dan datanya diambil dari tabel Question dan Choice pada basis data. Setelah kuesioner ditampilkan, *registrant* kemudian menjawab semua pertanyaan-pertanyaan dari kuesioner tersebut.

Sistem akan melakukan validasi berupa pengecekan kesesuaian format jawaban dan pengecekan jawaban yang wajib diisi. Pesan *error* akan ditampilkan jika validasi gagal, namun jika validasi sukses, jawaban kuesioner akan disimpan ke basis data dan *registrant* diarahkan ke laman Ucapan Terima Kasih. Pada laman Ucapan Terima Kasih, *registrant* dapat menekan tombol *redirect* untuk kembali ke

laman Pendaftaran Registrant, atau dapat menunggu selama 10 detik kemudian sistem secara otomatis me-*redirect* registrant ke laman Pendaftaran Registrant.

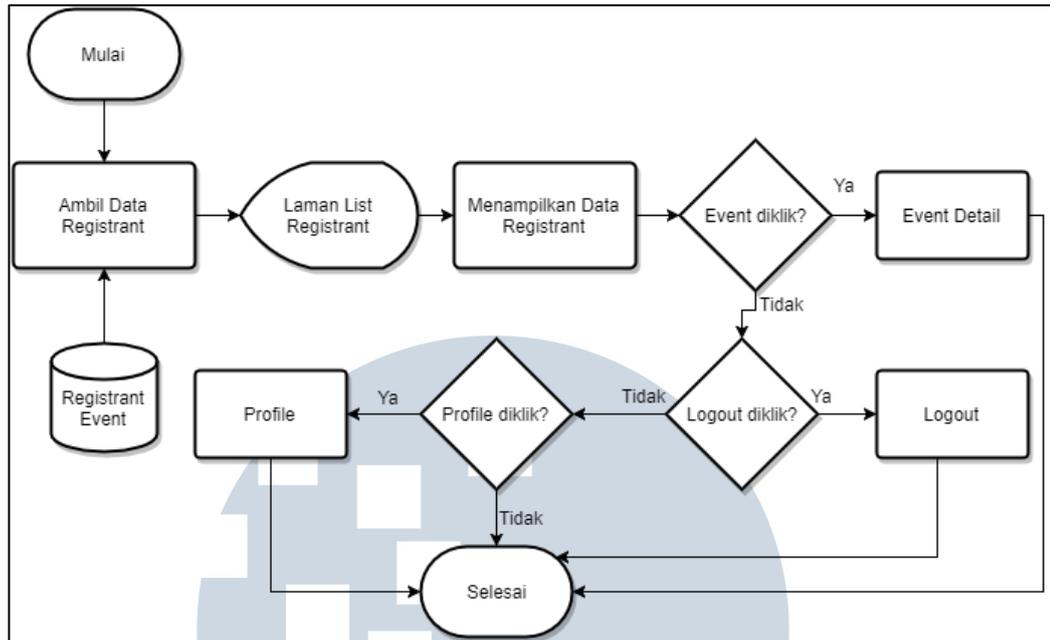


Gambar 3.6 Flowchart Kuesioner

D.4 Flowchart List Registrant

Gambar 3.7 merupakan Flowchart List Registrant. *Flowchart* ini mengacu pada proses 1 pada DFD Level 1. Pada laman List Registrant ini, data-data mengenai *registrant-registrant* yang terdaftar dalam sebuah *event* ditampilkan. Data-data ini merupakan data yang didaftarkan *registrant* kepada sistem saat proses pengisian kuesioner. Data-data dari tabel Event dan Registrant di basis data ditampilkan dalam bentuk tabel dengan ID-nya masing-masing.

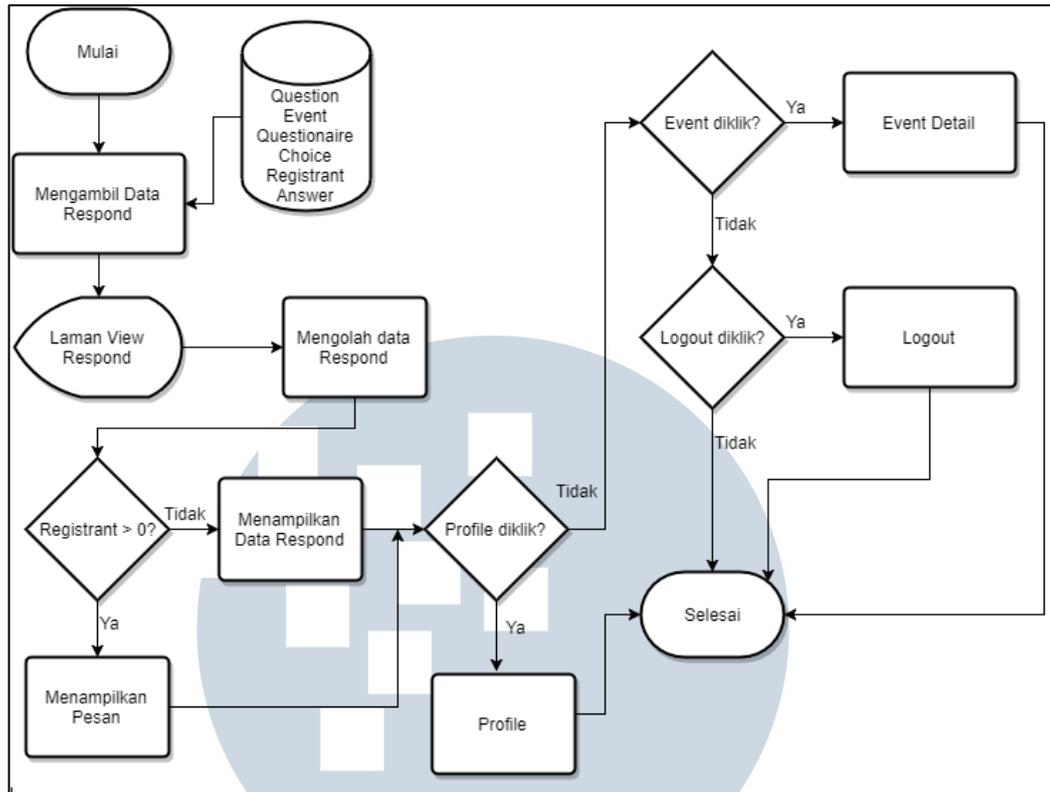
Pada laman List Registrant ini, terdapat beberapa navigasi seperti Event, Profile, dan Logout yang terhubung dengan proses-proses lain. Namun, pada laporan ini, proses Event Detail, Profile, dan Logout tidak dijelaskan karena merupakan entitas yang datang dari luar sistem.



Gambar 3.7 Flowchart List Registrant

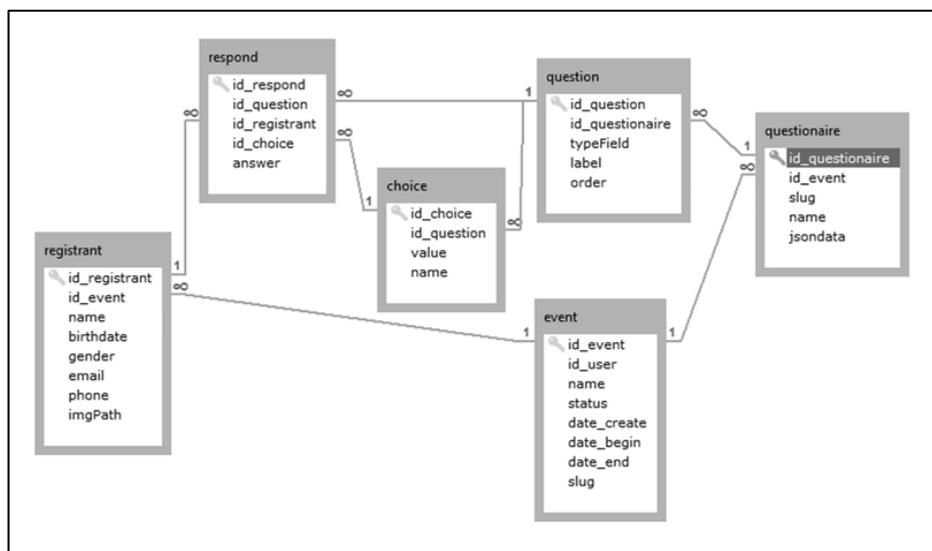
D.5 Flowchart View Respond

Gambar 3.8 merupakan Flowchart View Respond. *Flowchart* ini mengacu pada proses 2 pada DFD Level 1. Pada laman View Respond, ditampilkan data-data jawaban dari sebuah kuesioner atau yang disebut sebagai *respond*. Data *Respond* ini berasal dari enam tabel pada basis data, yaitu tabel Question, Event, Registrant, Choice, Respond, dan Questionnaire. Jika belum ada *registrant* yang telah menjawab kuesioner, maka ditampilkan pesan. Sebaliknya, data ditampilkan dalam bentuk tabel dan grafik jika sudah terdapat jawaban kuesioner. Pada laman View Respond ini, terdapat beberapa navigasi seperti Event, Profile, dan Logout yang terhubung dengan proses-proses lain. Namun, pada laporan ini, proses Event Detail, Profile, dan Logout tidak dijelaskan karena merupakan entitas yang datang dari luar sistem.



Gambar 3.8 Flowchart View Respond

E. Database Schema



Gambar 3.9 Database Schema

Gambar 3.9 merupakan Database Schema dari modul *user* Modafatar. Database yang digunakan untuk Modafatar ini telah dirancang terlebih dahulu oleh *project manager*. Pada modul *user* memiliki enam tabel, yaitu Tabel Event,

Questionnaire, Question, Choice, Respond, dan Registrant yang saling berelasi satu sama lain. Relasi tabel Event dan Questionnaire menjelaskan hubungan bahwa satu *event* dapat memiliki banyak kuesioner. Untuk hubungannya dengan tabel Registrant, memiliki arti bahwa dalam sebuah *event*, bisa terdapat banyak *registrant* yang mendaftar.

Kemudian, hubungan tabel Questionnaire dan Question merupakan hubungan *one to many* yang berarti bahwa satu kuesioner dapat terdiri dari banyak pertanyaan. Sementara hubungan tabel Question dengan Choice menyatakan sebuah pertanyaan pada kuesioner, dapat memiliki banyak jawaban (pilihan). Terakhir hubungan tabel Respond dengan Registrant dan Question. Hubungan ini menunjukkan jawaban dari satu pertanyaan kuesioner milik seorang *registrant* dan hubungan dengan tabel Choice menunjukkan bahwa jawaban kuesioner memiliki tipe pilihan dari tabel Choice.

F. Struktur Tabel

Modul *user* memiliki enam tabel, yang memiliki fungsi masing-masing. Berikut merupakan struktur tabel-tabel tersebut.

F.1 Tabel Event

Tabel Event merupakan sebuah tabel yang mengacu pada seorang *user* (*administrator*) dan menyimpan data *event-event* yang dibuat seorang *user*. Tabel Event akan dijabarkan pada Tabel 3.2.

Tabel 3.2 Struktur Tabel Event

Nama Kolom	Tipe Data	Keterangan
id_event	int(11)	Primary Key, Not Null, Auto Increment
id_user	int(11)	Foreign Key dari Tabel User
Slug	varchar(255)	Identitas unik sebuah <i>event</i>
name	varchar(255)	Nama dari sebuah <i>event</i>
status	varchar(255)	Status dari sebuah <i>event</i>

Tabel 3.2 Struktur Tabel Event (Lanjutan)

Nama Kolom	Tipe Data	Keterangan
date_create	date	Tanggal sebuah <i>event</i> dibuat pada sistem
date_begin	date	Tanggal sebuah <i>event</i> dimulai
date_end	date	Tanggal sebuah <i>event</i> berakhir

F.2 Tabel Questionnaire

Tabel Questionnaire merupakan tabel yang mengacu pada sebuah *event* dan menyimpan kuesioner. Tabel Questionnaire akan dijabarkan pada Tabel 3.3.

Tabel 3.3 Struktur Tabel Questionnaire

Nama Kolom	Tipe Data	Keterangan
id_questionnaire	int(11)	Primary Key, Not Null, Auto Increment
id_event	int(11)	Foreign Key dari Tabel Event
slug	varchar(255)	Identitas unik sebuah kuesioner
name	varchar(255)	Nama dari sebuah kuesioner
jsondata	text	Data kuesioner dalam bentuk JSON

F.3 Tabel Question

Tabel Question merupakan tabel yang mengacu pada tabel Questionnaire karena setiap baris data pada tabel menyimpan satu pertanyaan dari sebuah kuesioner. Tabel Question akan dijabarkan pada Tabel 3.4.

Tabel 3.4 Struktur Tabel Question

Nama Kolom	Tipe Data	Keterangan
id_question	int(11)	Primary Key, Not Null, Auto Increment
id_questionnaire	int(11)	Foreign Key dari Tabel Questionnaire
typeField	varchar(255)	Jenis <i>field form</i> pada kuesioner
label	varchar(255)	Satu pertanyaan pada kuesioner
order	int(11)	Urutan pertanyaan dalam sebuah kuesioner

F.4 Tabel Choice

Tabel Choice merupakan tabel yang mengacu pada tabel Question menyimpan pilihan-pilihan jawaban dari setiap pertanyaan yang memiliki tipe

radio button, *multiple choice*, dan *combo box*. Tabel Choice akan dijabarkan pada Tabel 3.5.

Tabel 3.5 Struktur Tabel Choice

Nama Kolom	Tipe Data	Keterangan
id_choice	int(11)	Primary Key, Not Null, Auto Increment
id_question	int(11)	Foreign Key dari Tabel Question
value	varchar(255)	Nilai/ <i>value</i> dari sebuah pilihan
name	varchar(255)	Nama dari sebuah pilihan

F.5 Tabel Respond

Tabel Respond merupakan tabel yang mengacu pada tabel Registrant, tabel Question, dan tabel Choice karena setiap baris data pada tabel Respond menyimpan jawaban *registrant* dari setiap pertanyaan dalam sebuah kuesioner. Jika tipe pertanyaan adalah *radio button*, *multiple choice*, dan *combo box*, maka hanya menyimpan referensi ke tabel Choice. Tabel Respond akan dijabarkan pada Tabel 3.6.

Tabel 3.6 Struktur Tabel Respond

Nama Kolom	Tipe Data	Keterangan
id_respond	int(11)	<i>Primary Key, Not Null, Auto Increment</i>
id_question	int(11)	<i>Foreign Key</i> dari Tabel Question
id_registrant	int(11)	<i>Foreign Key</i> dari Tabel Registrant
id_choice	int(11)	<i>Foreign Key</i> dari Tabel Choice
answer	Text	Jawaban <i>registrant</i> dari sebuah pertanyaan

F.6 Tabel Registrant

Tabel Registrant merupakan tabel yang menyimpan data pribadi dari semua *registrant* yang telah terdaftar dalam sebuah *event*. Tabel Registrant akan dijabarkan pada Tabel 3.7.

Tabel 3.7 Struktur Tabel Registrant

Nama Kolom	Tipe Data	Keterangan
id_registrant	int(11)	Primary Key, Not Null, Auto Increment
id_event	int(11)	Foreign Key dari Tabel event
name	varchar(255)	Nama registrant
birthdate	date	Tanggal Lahir registrant
gender	varchar(5)	Jenis kelamin registrant
email	varchar(255)	Email registrant
phone	varchar(255)	Nomor telepon registrant
imgPath	varchar(255)	Path foto registrant

G. Perancangan Antarmuka Sistem

G.1 Rancangan Laman Tambah Registrant

New Registrant

Name

Birthdate
 / /

Gender

Email

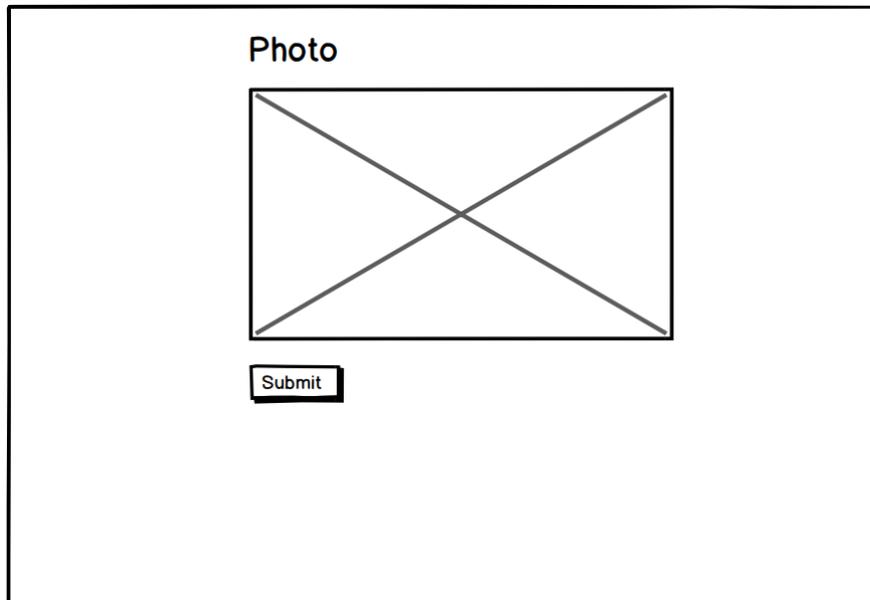
Phone

Registrant

Gambar 3.10 Rancangan Laman New Registrant (Data Diri)

Gambar 3.10 merupakan rancangan laman New Registrant yang harus diisi oleh *registrant*. Terdapat dua *form* pada laman ini, *form* pada bagian atas diperuntukkan untuk *registrant* baru, terdapat *textbox* untuk nama, *email*, dan telepon, sebuah *combo box* untuk jenis kelamin, dan *date picker* untuk tanggal lahir yang harus diisi *registrant* sebelum melanjutkan ke tahap berikutnya. Pada *form*

kedua, terdapat satu *textbox* untuk mengisi ID bagi *registrant* yang pernah mendaftar sebelumnya.



Gambar 3.11 Rancangan Laman New Registrant (Foto)

Gambar 3.11 merupakan Rancangan Laman New Registrant saat pengambilan foto. Laman ini merupakan kelanjutan dari pendaftaran *registrant* baru yang telah mengisi data diri sebelumnya. Pada laman ini terdapat sebuah *photo viewer* yang menampilkan foto yang telah diambil. Kedua perancangan laman ini dibuat berdasarkan Flowchart Tambah Registrant pada Gambar 3.5.

G.2 Rancangan Laman Kuesioner

Gambar 3.12 merupakan rancangan Laman Kuesioner. Rancangan laman ini mengikuti Flowchart Kuesioner pada Gambar 3.6.

Gambar 3.12 Rancangan Laman Kuesioner

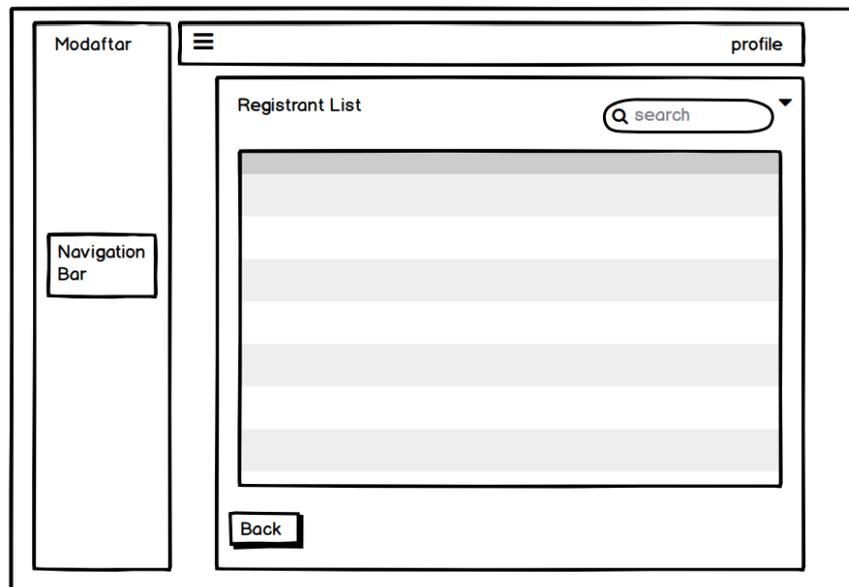
Pada laman ini terdapat pertanyaan-pertanyaan yang harus diisi oleh *registrant*. Jenis-jenis pertanyaan dapat bervariasi, pertanyaan dengan *input* jawaban berupa *textbox*, *date picker*, *combo box*, *radio button*, dan *multiple select*. Terdapat juga sebuah *button* untuk *submit* jawaban kuesioner.

G.3 Rancangan Laman Ucapan Terima Kasih

Gambar 3.13 Rancangan Laman Ucapan Terima Kasih

Gambar 3.13 merupakan rancangan Laman Ucapan Terima Kasih. Laman ini dirancang berdasarkan Flowchart Kuesioner pada Gambar 3.6. Pada laman ini hanya terdapat ucapan terima kasih atas ketersediaan *registrant* untuk menjawab kuesioner. Terdapat *hyperlink* pada kata *click here* untuk melakukan *redirect* langsung ke laman Tambah Registrant.

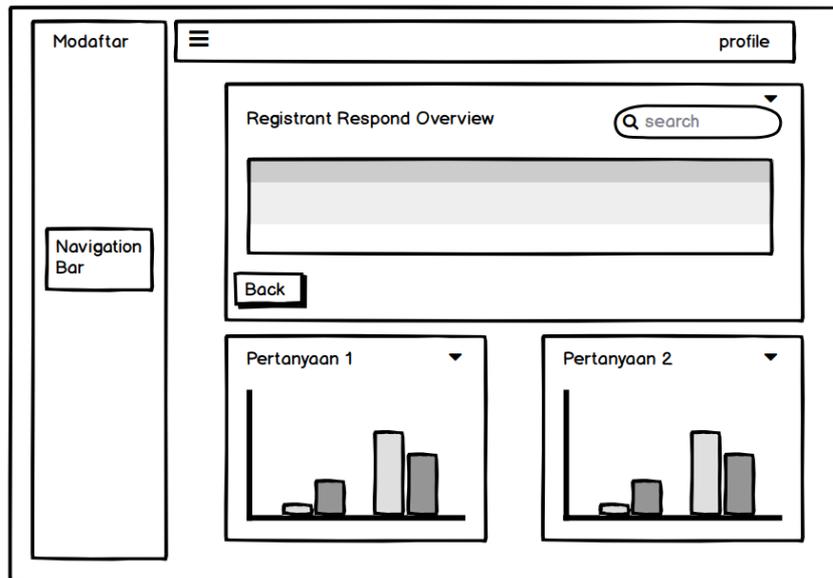
G.4 Rancangan Laman View Registrant



Gambar 3.14 Rancangan Laman View Registrant

Gambar 3.14 merupakan Rancangan Laman View Registrant yang dirancang menurut Flowchart List Registrant pada Gambar 3.7. Laman ini menampilkan daftar-daftar *registrant* yang telah terdaftar dalam sebuah *event*. Pada laman ini, *navigation bar* terdapat di sisi kiri halaman, sementara pada bagian atas laman terdapat *icon menu* untuk memunculkan atau menyembunyikan *navigation bar* dan pada kanan atas ditampilkan foto *admin* dan *username*-nya yang jika di klik dapat memunculkan *pop-up menu* untuk tautan *logout* dan kustomisasi profil.

G.5 Rancangan Laman View Respond



Gambar 3.15 Rancangan Laman View Respond

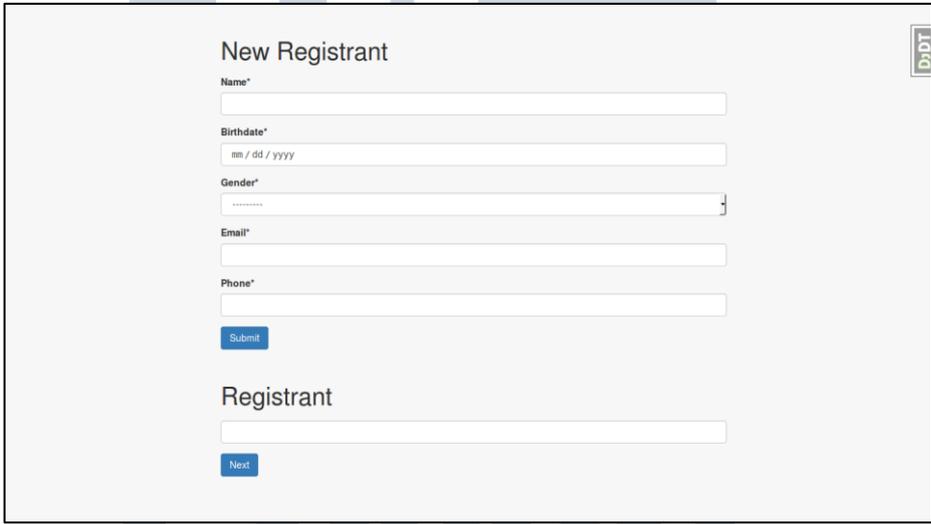
Gambar 3.15 merupakan Rancangan Laman View Respond. Perancangan laman ini menyesuaikan dan mengacu pada Flowchart View Respond pada Gambar 3.8. Pada laman ini menampilkan data *respond* kuesioner tertentu. Data ditampilkan pada tabel yang berada di tengah halaman. Pada tabel terdapat kolom pencarian untuk mencari data tertentu di dalam tabel. Kemudian terdapat tombol *back* untuk kembali pada halaman sebelumnya.

Pada setiap *box/container*, terdapat sebuah *button* kecil pada kanan atas untuk *minimize container* tersebut. Pada bagian bawah tengah halaman, terdapat *container* yang berisi grafik, grafik tersebut menampilkan presentase dari pertanyaan yang memiliki tipe *combo box*, *radio button*, dan *multiple choice*.

H. Implementasi Sistem

Pada tahap implementasi sistem, digunakan beberapa *library* dan *plugin* tambahan. Terdapat Chart.js untuk membuat grafik pada implementasi laman View Respond pada Gambar 3.21. Selain itu, digunakan juga Datatables untuk implementasi tabel pada laman View Registrant pada Gambar 3.20 dan tabel pada laman View Respond pada Gambar 3.21. Terakhir, digunakan *library* untuk *front-end* berupa Bootstrap untuk implementasi pada semua laman.

H.1 Laman Tambah Registrant



Gambar 3.16 Laman Tambah Registrant (Data Diri)

Gambar 3.16 merupakan implementasi dari laman Tambah Registrant. Pada laman ini, *registrant* baru akan mengisi data dirinya pada *form* yang berada di atas. Sementara itu, *registrant* yang telah mendaftar pada *event* ini, akan mengisi ID-nya pada *form* di bawah. Tombol *submit* akan membawa *registrant* pada laman Tambah Registrant bagian foto. Sementara, tombol *next* akan membawa *registrant* pada laman Kuesioner.



Gambar 3.17 Laman Tambah Registrant (Foto)

Gambar 3.17 merupakan implementasi dari Laman Tambah Registrant bagian foto. Laman ini merupakan laman tahap kedua dari pendaftaran *registrant* baru. Registrant akan diambil fotonya dengan kamera pada *device* yang bersangkutan dan hasilnya akan ditampilkan pada *image view* di tengah halaman. Tombol *submit* berfungsi untuk melanjutkan ke tahapan selanjutnya seperti pada Gambar 3.18.

H.2 Laman Kuesioner

Survey Kepuasan Customer

Nama Lengkap
Viktor

Jenis Kelamin
 Laki-Laki
 Perempuan

Tanggal Lahir
08 / 05 / 1996

Bagaimana tingkat kepuasan anda terhadap acara?
Sangat Puas

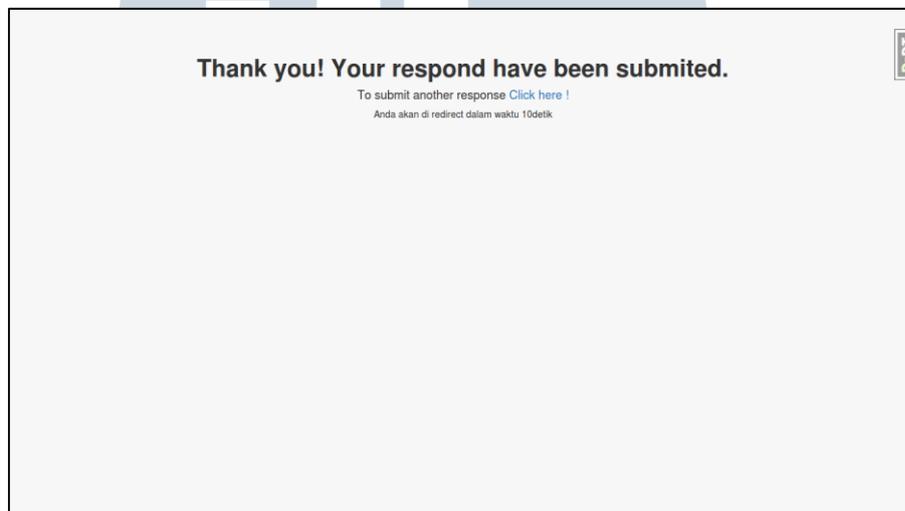
Menurut anda, dalam hal apa yang perlu ditingkatkan kedepannya untuk acara ini?
 Keamanan
 Kebersihan
 Keteraturan

Submit Query

Gambar 3.18 Laman Kuesioner

Gambar 3.18 merupakan implementasi dari rancangan laman Kuesioner. Pada laman ini, *registrant* akan menjawab pertanyaan-pertanyaan di kuesioner tersebut. Sebagai contoh, pada kuesioner ini *text-box* digunakan untuk menanyakan nama, sementara *date picker* digunakan untuk mengisi tanggal lahir. Kemudian terdapat *radio button* yang digunakan untuk pengisian jenis kelamin, *check box* untuk memberikan saran peningkatan dan *dropdown/select* diimplementasikan untuk menanyakan kepuasan *customer*.

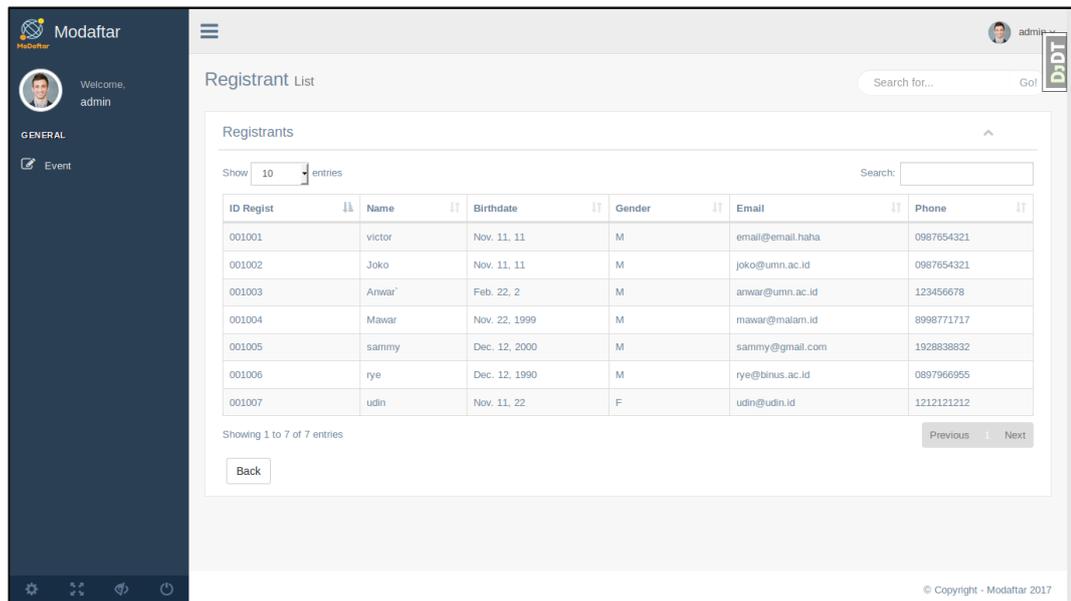
H.3 Laman Ucapan Terima Kasih



Gambar 3.19 Laman Ucapan Terima Kasih

Gambar 3.19 merupakan implementasi dari rancangan laman Ucapan Terima Kasih. Laman ini akan otomatis melakukan *redirect* ke laman Tambah Registrant dalam 10 detik. Selain itu, *hyperlink* pada kata *click here* dapat digunakan untuk *redirect* secara langsung.

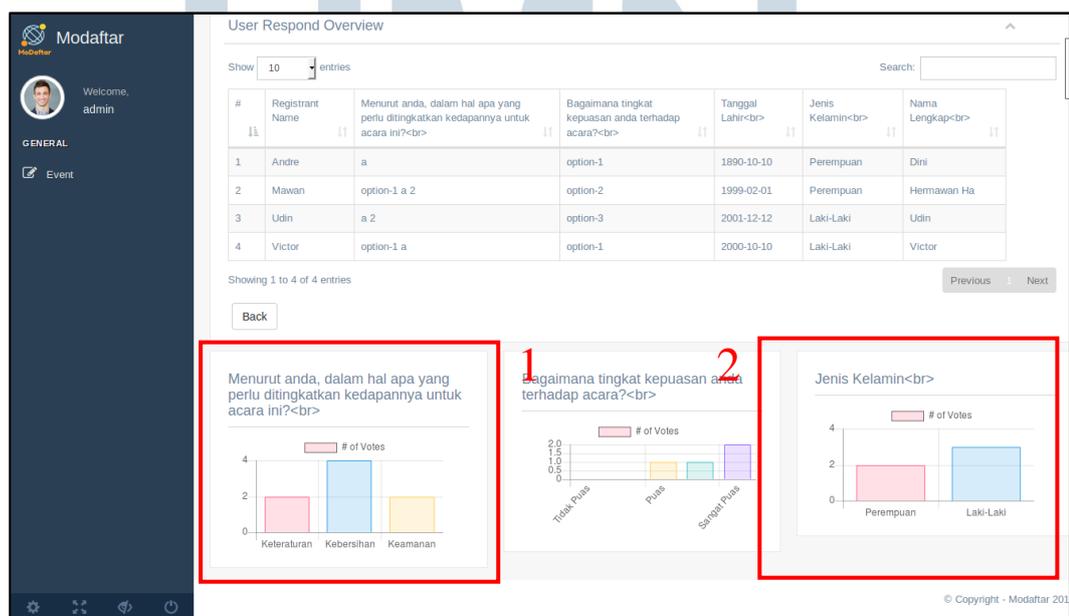
H.4 Laman View Registrant



Gambar 3.20 Laman View Registrant

Gambar 3.20 merupakan implementasi dari laman View Registrant. Pada laman ini, semua *registrant* dalam sebuah *event* akan ditampilkan. Data-data *registrant* ditampilkan dalam bentuk tabel beserta dengan ID dari setiap *registrant*.

H.5 Laman View Respond



Gambar 3.21 Laman View Respond

Gambar 3.21 merupakan implementasi dari rancangan laman View Respond. Pada laman ini, *admin* dari sebuah *event* yang bersangkutan dapat melihat data-data yang telah diisi oleh *registrant*. Setiap pertanyaan pada kuesioner akan menjadi sebuah kolom pada tabel yang berada di tengah laman beserta dengan jawabannya. Implementasi grafik dapat dilihat pada Gambar 3.21 bagian 1 yang menunjukkan statistik berupa *bar chart* jenis kelamin dari *registrant* yang menjawab kuesioner dan pada Gambar 3.21 bagian 2 menunjukkan statistik untuk pengembangan ke depan dari acara tersebut.

3.3.2 Kendala yang ditemukan

Kendala yang ditemukan selama pelaksanaan kerja magang di PT Indosterling Technomedia adalah pembuatan sistem dengan bahasa pemrograman dan *framework* yang belum pernah dipelajari sebelumnya.

3.3.3 Solusi atas kendala yang ditemukan

Solusi atas kendala yang ditemukan selama pelaksanaan kerja magang di PT Indosterling Technomedia adalah dengan belajar. Pada minggu pertama magang, tidak langsung melakukan pekerjaan, namun belajar bahasa pemrograman Python dan *framework* Django untuk membangun *website* dari berbagai sumber seperti internet, buku, dokumentasi bahasa pemrograman dan *framework*-nya.