

BAB 3 PELAKSANAAN KERJA MAGANG

3.1. Kedudukan dan Organisasi

Dalam proses pelaksanaan kerja magang, penulis menjabat sebagai web developer pada divisi Pemegang Layanan Polis. Tim terdiri dari 2 orang di mana setiap orang bertanggung jawab pada satu proyek. Setelah kedua proyek selesai dikerjakan maka akan dilakukan penggabungan sehingga menjadi satu web aplikasi. Proyek ini dipimpin dan dipantau oleh Bapak Bakhtiar Rachman selaku Assistant VP Policy Owner Services. Bapak Bakhtiar berperan dalam memberikan informasi, membimbing, mengawasi serta mengevaluasi pengerjaan proyek *tracking* dan *monitoring* klaim. Selain itu, penulis diberikan arahan terkait proses bisnis oleh Bapak Arifien yang ditunjuk oleh Bapak Bakhtiar sebagai mentor penulis. Koordinasi tentang perkembangan proyek, diskusi, serta bertukar ide dilakukan secara langsung ketika bekerja *Work From Office* (WFO), dan aplikasi Zoom dan WhatsApp ketika sedang *Work From Home* (WFH).

3.2. Tugas yang Dilakukan

Selama pelaksanaan kerja magang, penulis mengemban tanggung jawab sebagai berikut:

- Merancang flowchart dari proses bisnis yang telah dipaparkan oleh *supervisor*.
- Mencari dan mempelajari library yang dibutuhkan untuk mendukung pengembangan proyek.
- Merancang tampilan antar muka dari setiap halaman.
- Membuat table dan relasinya untuk digunakan pada fitur Inbox dan Profil Penerimaan.
- Membuat halaman Login, Dashboard, Inbox, Details, Profil Penerimaan, dan Tracking Klaim.
- Melakukan *Testing* dan *Debugging* pada aplikasi yang dibuat.

- Bekerja sama dengan anggota tim jika terdapat fitur yang saling berhubungan dan melakukan *merger* pada dua proyek menjadi satu aplikasi.

3.3. Uraian Pelaksanaan Magang

Pelaksanaan kerja magang dilaksanakan selama 100 (seratus) hari kerja atau 20 (dua puluh) minggu. berikut adalah *gantt chart timeline* kerja magang dan table detail kegiatan yang dilakukan setiap minggunya.

Nama Kegiatan	Minggu Ke-																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Briefing requirements	■																			
Mempelajari proses bisnis yang dipaparkan	■	■																		
Membuat flow aplikasi		■	■																	
Mempersiapkan software, tools dan library			■	■																
Pengenalan dan pembelajaran software				■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Mendesain tampilan antarmuka (UI/UX)					■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Coding																				
Konsultasi permasalahan	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Test & Debug																	■	■	■	■
Merger & Finishing Project																			■	■

Gambar 3.1. *Timeline* kerja magang.

Berikut merupakan penjelasan kegiatan yang tertera pada *Gantt Chart*

- **Briefing Requirement.** Pada tahap ini, penulis akan mendapatkan pengarahan langsung mengenai proses bisnis yang dilakukan, masalah yang dihadapi, dan apa yang dibutuhkan oleh divisi Layanan Pemegang Polis. Pengarahan diberikan oleh *supervisor*.
- **Mempelajari bisnis yang dipaparkan.** Penulis membutuhkan waktu untuk memahami proses bisnis, dan permasalahan yang dihadapi. Penulis mendapatkan bimbingan dari *supervisor* untuk menjelaskan hal-hal yang tidak dipahami penulis. Hal ini berguna untuk perancangan alur aplikasi pada kegiatan berikutnya.
- **Membuat alur aplikasi.** Penulis merancang diagram *site map*, *flowchart* aplikasi secara keseluruhan, dan membagi cakupan *flowchart* menjadi lebih kecil.
- **Mempersiapkan *softwar*, *tools*, dan *library*.** Penulis mengunduh dan melakukan instalasi perangkat lunak, peralatan, dan *libary* yang diperlukan.

- **Pengenalan dan pembelajaran *software*.** Penulis harus beradaptasi untuk penggunaan bahasa pemrograman PHP, penggunaan *framework* CodeIgniter 4, dan menggunakan Figma untuk mendesain UI/UX.
- **Mendesain tampilan UI/UX.** Penulis membuat desain *prototype* untuk gambaran bagaimana *output* yang diberikan dari pengerjaan proyek ini.
- ***Coding*.** Penulis memulai *setup* proyek dan melakukan pemrograman baik pada *front-end* maupun *back-end*.
- **Konsultasi permasalahan.** Jika terdapat kendala dan permasalahan, penulis akan berkonsultasi dan berdiskusi dengan *supervisi* untuk menemukan solusi terbaik.
- ***Test, dan Debug*.** Penulis melakukan test pada aplikasi untuk menemukan bug dan memperbaikinya supaya aplikasi berjalan dengan baik.
- ***Merger, dan Finishing Project*.** Penulis dan rekan penulis melakukan penggabungan dua proyek yang telah selesai kami kerjakan menjadi satu aplikasi

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Pengarahan kerja magang, pemaparan proses bisnis, dan pemaparan masalah apa yang dihadapi. Pengarahan dan pemaparan dilakukan oleh <i>supervisor</i>
2	Memahami proses bisnis yang dipaparkan, diskusi untuk membuat solusi dari masalah yang dihadapi, dan penentuan software, bahasa pemrograman dan <i>framework</i> apa yang akan digunakan. Selain itu, membuat <i>flowchart</i>
3	Membuat <i>site-map diagram</i> , <i>use-case diagram</i> , dan skema <i>database scheme</i> untuk table profil penerimaan dan inbox. Selain itu, Melakukan instalasi <i>software</i> PHPStrom, MAMP, Composer, CodeIgniter 4, Figma, serta mengunduh <i>library</i> IMAP
4	Mempelajari penggunaan <i>software</i> Figma, MAMP, Composer
5	Mempelajari penggunaan <i>framework</i> CodeIgniter 4, Mempelajari cara penggunaan <i>external library</i> pada CodeIgniter 4.
6	Melanjutkan mempelajari CodeIgniter 4. Mulai membuat UI/UX halaman <i>Login</i> , <i>Home</i> , dan <i>Inbox</i>
7	Melanjutkan membuat UI/UX <i>Inbox</i> dan <i>Tracking</i>
8	<i>Setup project</i> dan implementasi desain UI/UX ke <i>code</i> (halaman <i>Dashboard</i> dan <i>Tracking</i>)
9	Implementasi <i>code front-end</i> pada halaman <i>Inbox</i> dan Profil Penerimaan
10	Implementasi <i>code front-end</i> pada halaman <i>Message detail</i> dan <i>Tracking detail</i>
11	Membuat Database Schema dan tabel baru pada database untuk menyimpan <i>Inbox</i> , <i>User Account</i> , dan Profil Penerimaan
12	Implementasi PHP library IMAP dari Ddeboer[1] untuk mengambil data pesan masuk dari Gmail
13	Implementasi Model-View-Controller halaman <i>Inbox</i> dan halaman <i>Detail Message</i>
14	Implementasi Model-View-Controller halaman <i>Tracking</i> dan halaman <i>Tracking Detail</i>

Minggu ke-	Pekerjaan yang dilakukan
15	Implementasi Model-View-Controller pada halaman Profil Penerimaan
16	Membuat fitur CRUD untuk Create profil penerimaan yang akan disimpan ke database
17	Melakukan tes aplikasi dan memperbaiki <i>bug</i> yang ditemukan
18	Membantu proyek kartu dalam membuat <i>function</i> yang akan <i>generate</i> kode QR
19	Menggabungkan proyek aplikasi <i>Tracking Klaim</i> dan <i>Tracking Kartu</i>
20	Melakukan tes pada keseluruhan aplikasi dan memperbaiki <i>bug</i> yang ditemukan. Proyek Selesai

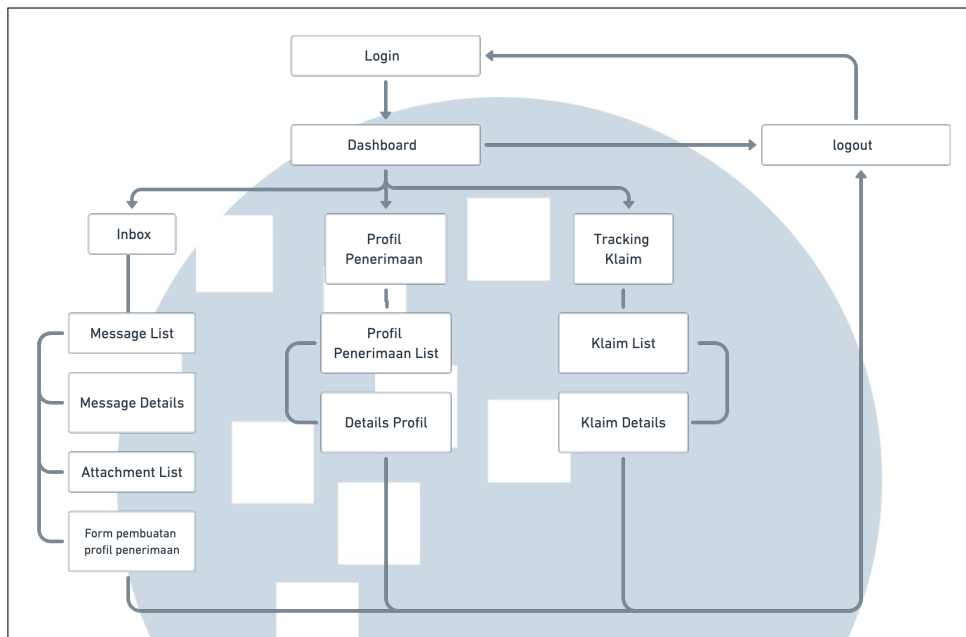
3.4. Perancangan Sistem

Dalam proses pelaksanaan magang, dirancanglah dokumen-dokumen yang akan menjadi acuan dalam pembangunan proyek, atau memberikan gambaran secara garis besar tentang bagaimana hasil dari proyek bekerja. Berikut merupakan dokumen-dokumen yang telah dirancang.

3.4.1. Diagram Site Map

Diagram *site map* merupakan sebuah diagram yang menggambarkan sebuah urutan atau alur dari sebuah aplikasi atau website yang akan digunakan oleh *user*. Dengan diagram ini, diharapkan user mendapatkan tentang bagaimana alur dari aplikasi yang akan digunakan [4].

UNIVERSITAS
MULTIMEDIA
NUSANTARA



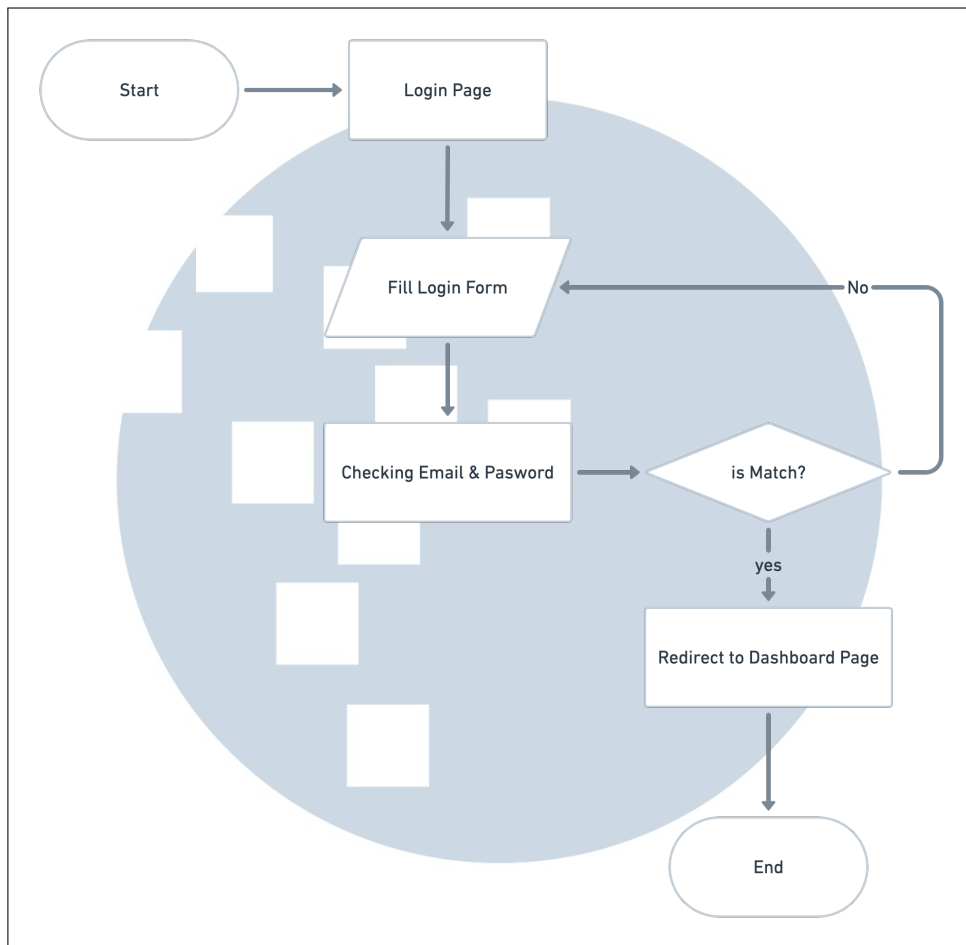
Gambar 3.2. Diagram *site map*

3.4.2. Flowchart

Untuk lebih mudah dalam memahami alur kerja aplikasi, flowchart dibagi menjadi beberapa bagian. Yaitu *login*, registrasi user baru, *Inbox*, detail dari pesan yang diterima, *Tracking* dan detailnya, serta *flowchart* profil penerimaan.

A. Login

Secara *Default*, hanya ada *role admin* dalam pengembangan website ini, sehingga penulis akan membuatkan satu akun yang dapat digunakan untuk *login*. Penambahan user baru (dengan *role admin*) tetap dapat dilakukan setelah *login*. Hal ini dilakukan agar orang luar yang menemukan website ini tidak dapat menambahkan akun baru dan masuk ke dalam aplikasi yang telah dibangun.

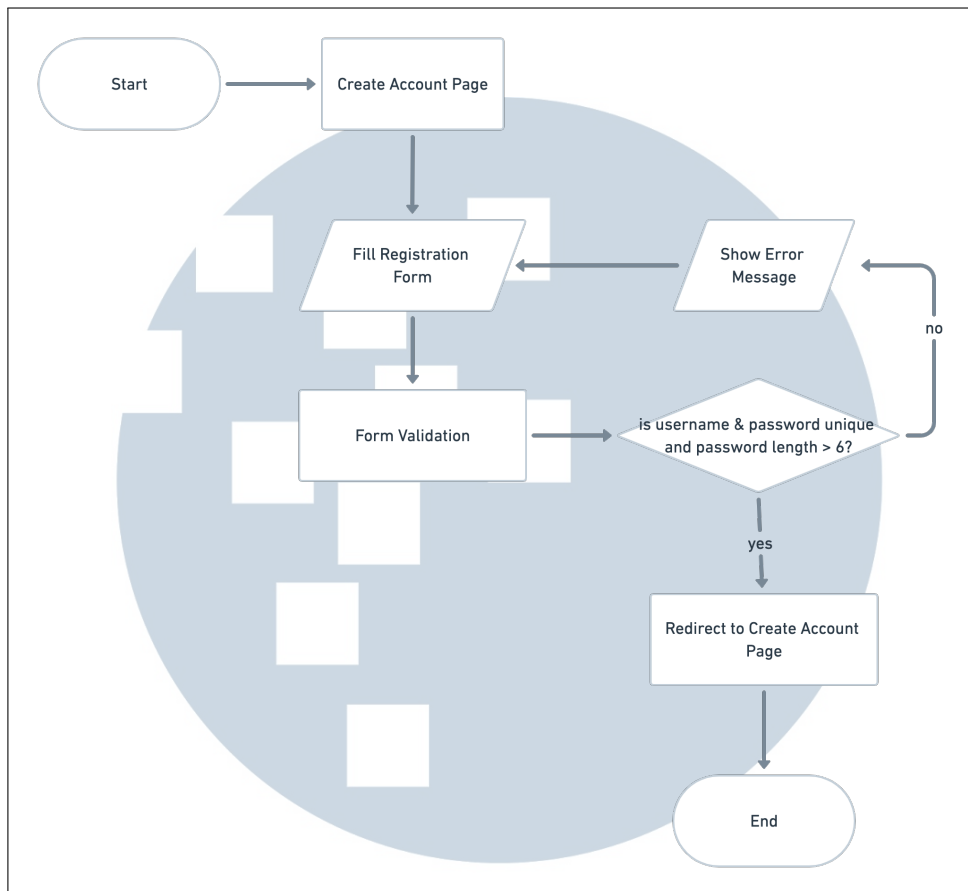


Gambar 3.3. Flowchart untuk halaman login

Saat pertama kali membuka website, user hanya diberikan *form* untuk autentikasi masuk ke dalam akun. Setelah berhasil masuk, *user* akan diarahkan ke halaman Dashboard.

B. Registration

User tetap dapat membuat akun baru dengan persyaratan akun baru hanya bisa dibuat oleh *user* yang telah login. *User* diberikan form input, lalu dilakukan *form validation* untuk mengecek apakah *username* atau *email* sudah digunakan, dan password harus minimal 6 (enam) karakter. Jika kondisi sudah cocok, maka data akan disimpan ke database dan user akan diarahkan kembali ke halaman registrasi

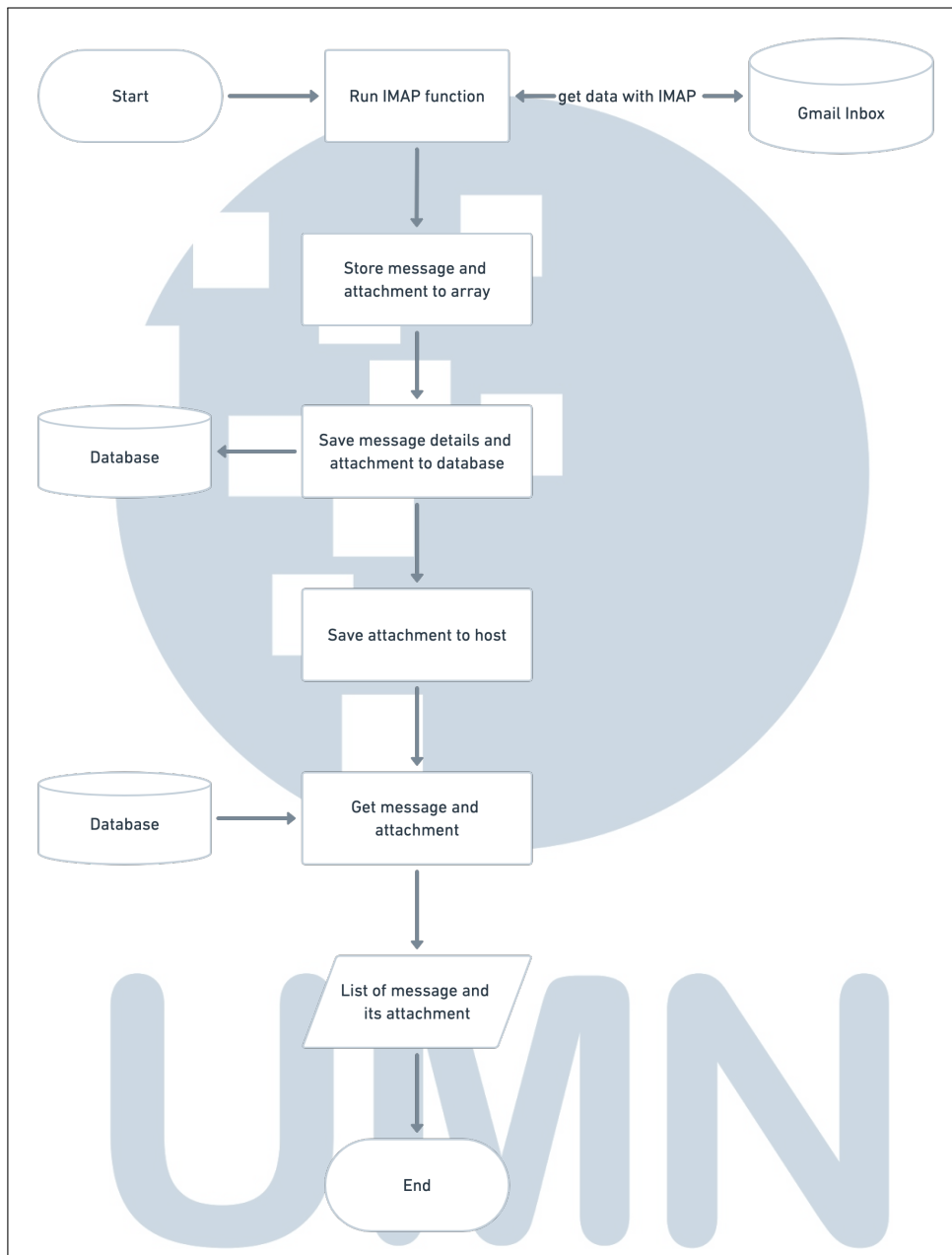


Gambar 3.4. *Flowchart* untuk halaman *Registration*

C. Inbox dan Message Details

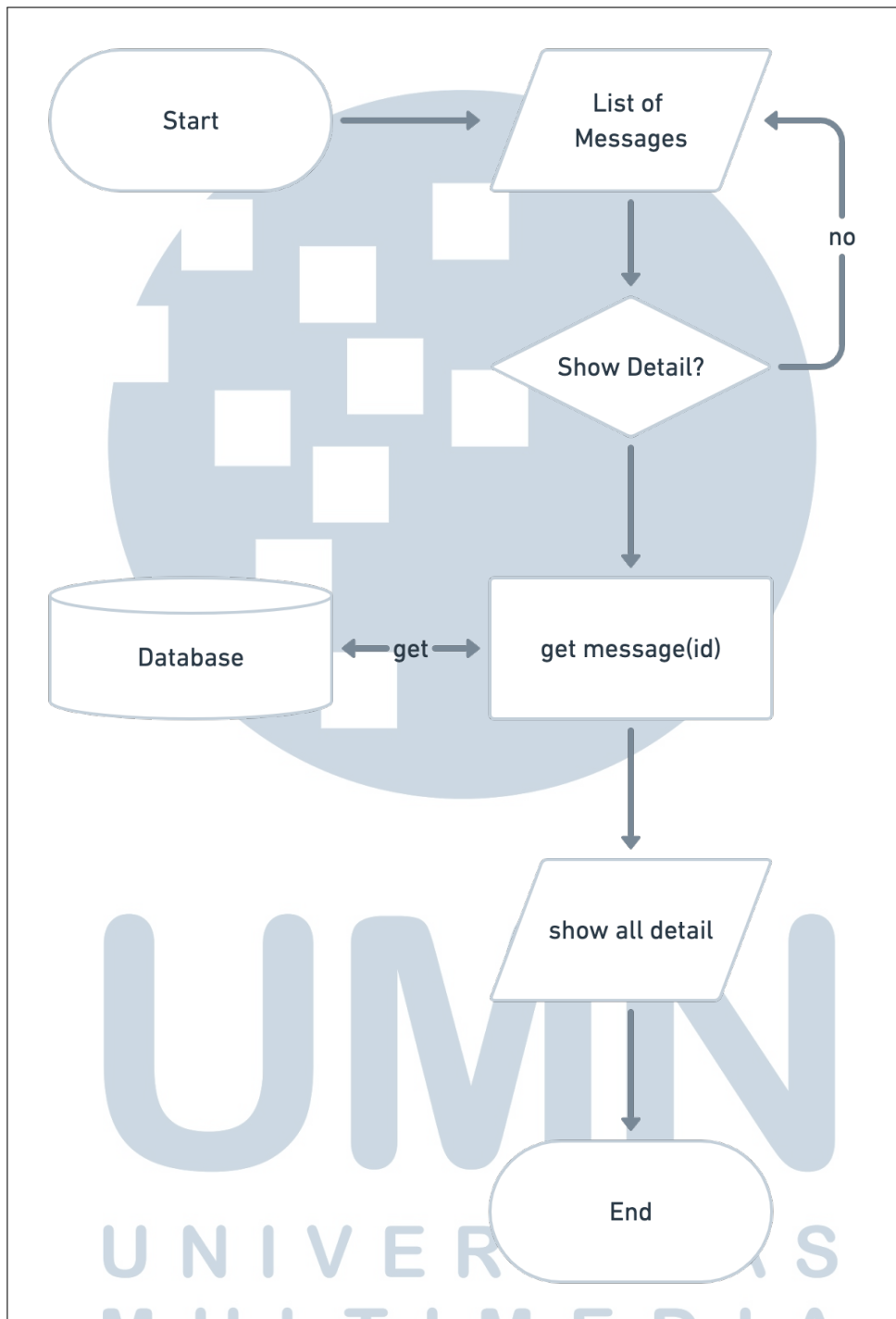
Saat halaman inbox dibuka, maka di bagian *back-end* akan menjalankan proses *get* data semua pesan yang masuk beserta dokumen lampiran jika ada. Selanjutnya, data yang sudah diambil akan disimpan ke dalam database dan akan di-*get* lalu ditampilkan pada halaman Inbox dalam bentuk tabel *list*

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.5. Flowchart untuk halaman *Inbox*

Pada tabel *list inbox* tidak ditampilkan isi atau *body* email. Untuk melihatnya, disediakan tombol action untuk melihat detail lebih lengkap dari setiap pesan.

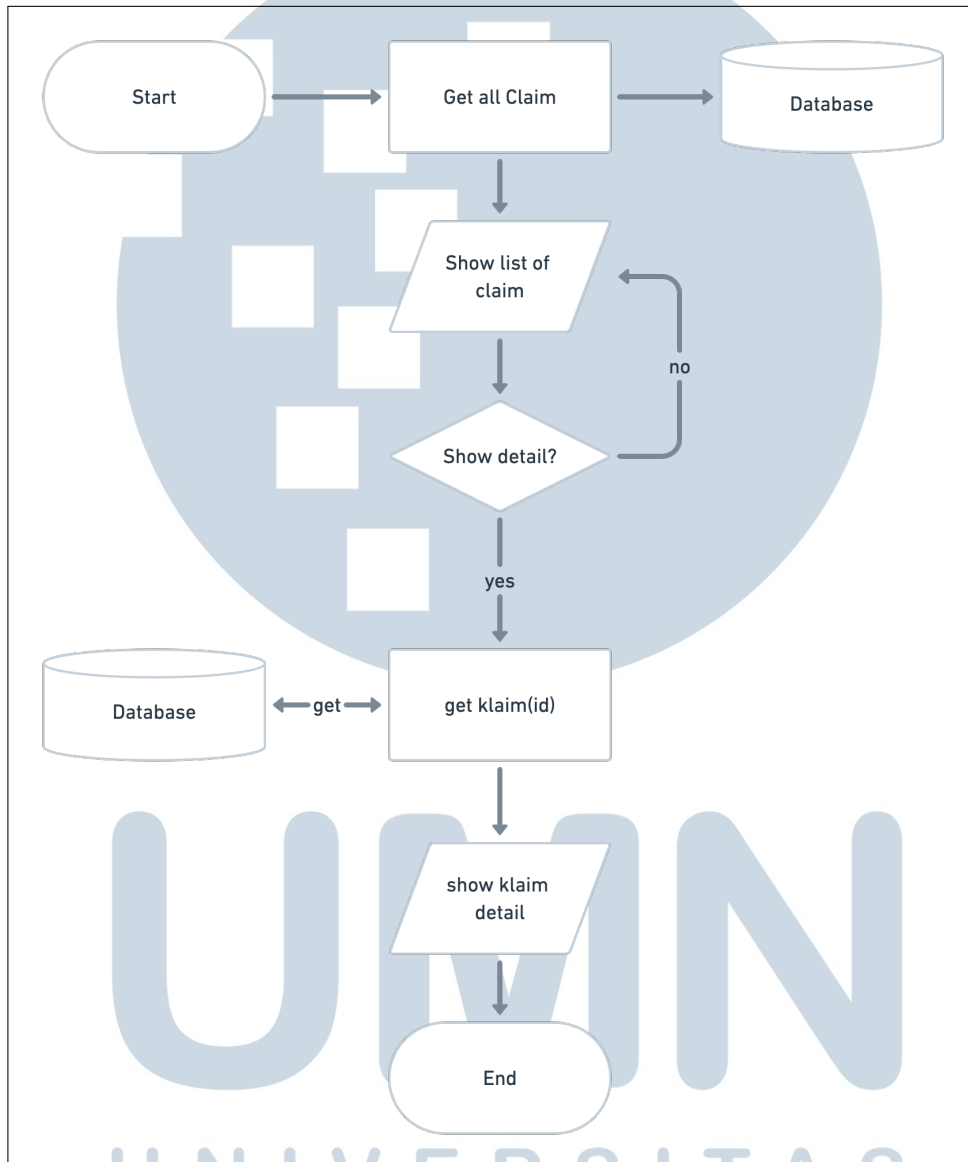


Gambar 3.6. Flowchart untuk halaman *Message Detail*

D. *Tracking dan Tracking Details*

Pada halaman *tracking*, data dari database diambil dan ditampilkan pada halaman tersebut. Karena *field* pada table sangat

banyak, hanya beberapa informasi saja yang diambil. Untuk detail lengkapnya, dibuatkan tombol action yang akan mengambil data dari *list* yang dipilih.

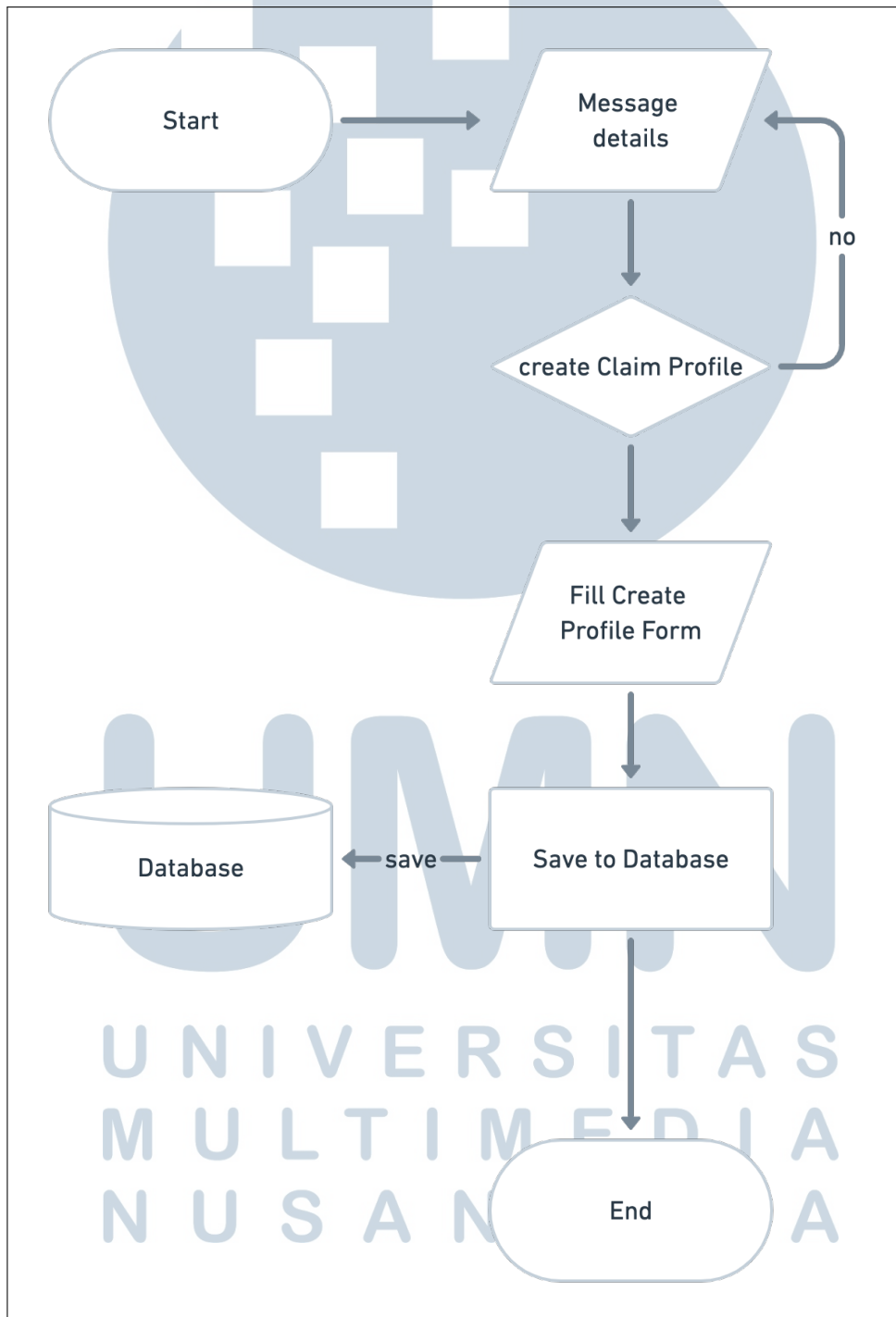


Gambar 3.7. Flowchart untuk halaman Track dan Detail

E. Profil Penerimaan dan Create New Profile

Pada halaman detail pesan, terdapat tombol yang berfungsi untuk *trigger modal form* untuk membuat profil penerimaan baru. User hanya perlu memasukkan beberapa kolom saja, sedangkan kolom lainnya akan terisi secara otomatis sesuai dengan data pada detail

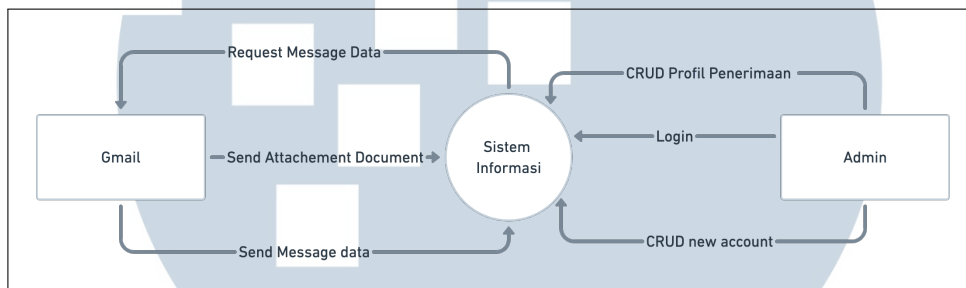
pesan. Setelah di submit, profil penerimaan baru akan disimpan ke dalam *database* dan ditampilkan pada halaman profil penerimaan. Untuk halaman *list* profil penerimaan, *flowchart* tidak ditampilkan karena proses yang dilakukan hanya *get data*.



Gambar 3.8. *Flowchart* untuk halaman *Create Profil Penerimaan*

3.4.3. Data Flow Diagram

Diagram alir data dibawah ini menunjukkan *flow* data yang berjalan pada aplikasi. Sistem akan meminta data *message* ke Gmail, lalu Gmail akan mengirimkan data *message* beserta lampirannya. Pada entitas Admin, admin akan mengirimkan data input ke sistem untuk melakukan autentikasi *login* dan membuat akun baru. Admin juga akan melakukan CRUD profil penerimaan ke dalam sistem.

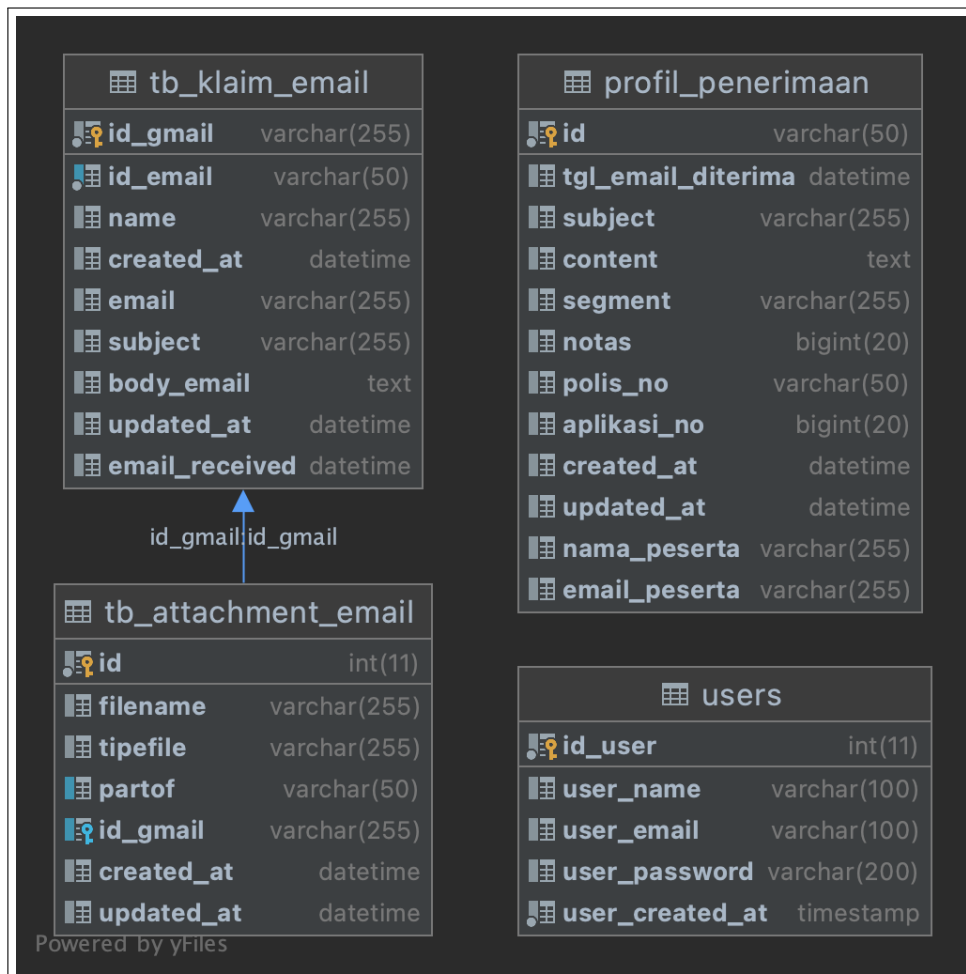


Gambar 3.9. *Data Flow Diagram*

3.4.4. Database Scheme

Di dalam database yang digunakan pada proyek ini, tabel yang digunakan hanya 1 tabel yang sudah ada dan sisanya penulis merancang sendiri tabel yang akan digunakan saat proses pembuatan aplikasi. Table dari perusahaan tidak ditampilkan penulis karena menyangkut rahasia perusahaan. Berikut ini merupakan skema *database* yang telah penulis rancang.

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 3.10. Skema Database

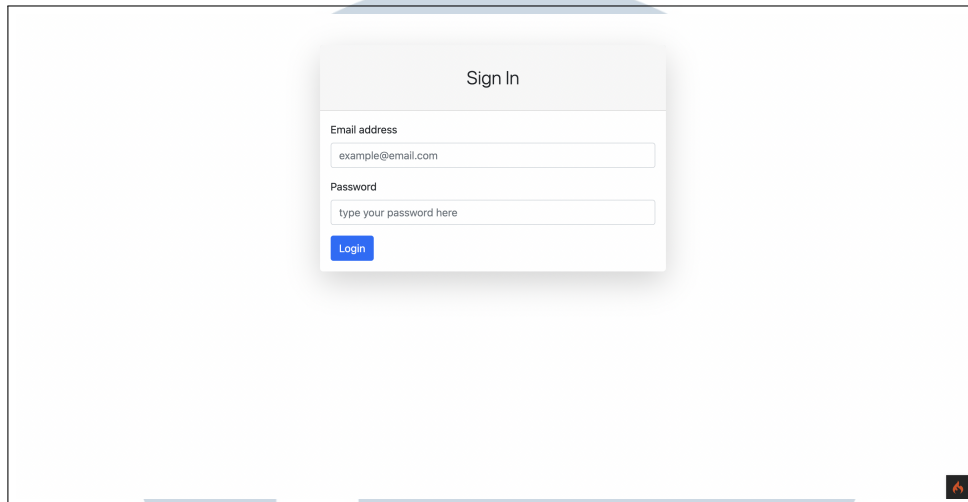
Pada skema database ini, dapat dilihat hanya ada 2 tabel yang saling berelasi yakni tavke klaim email dan tabel attachment email, dengan id gmail sebagai *foreign key*-nya. relasi ini dibuat untuk mempejelas suatu attachment merupakan bagian dari pesan yang mana.

3.5. Hasil Implementasi

Setelah merancang dokumen-dokumen yang dibutuhkan sebagai acuan pembuatan aplikasi, selanjutnya penulis mulai melakukan implementasi pembuatan aplikasi. Berikut ini merupakan *screen capture* dari hasil implementasi yang telah penulis lakukan.

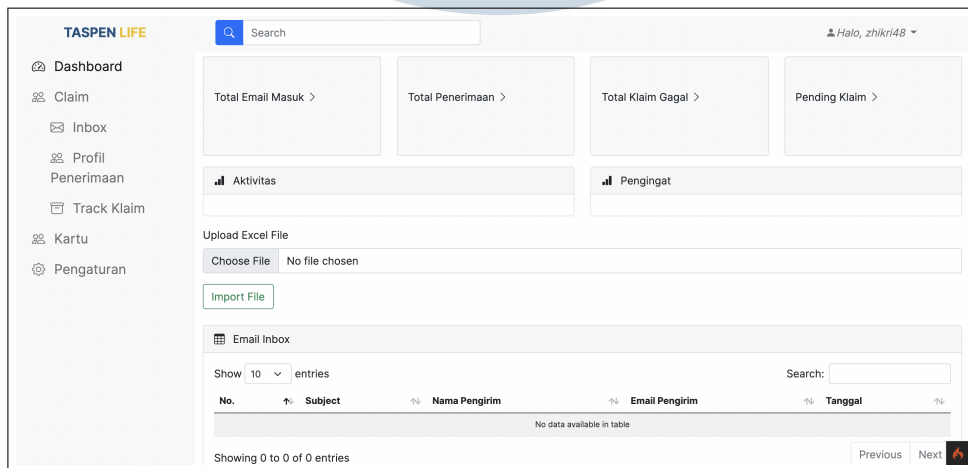
Saat user membuka website pertama kali, maka akan ada *form login* untuk autentikasi masuk ke dalam website. Akun untuk login telah dipersiapkan untuk

admin. untuk membuat akun lainnya maka dapat ke menu *setting* dan pilih ‘buat akun baru’.



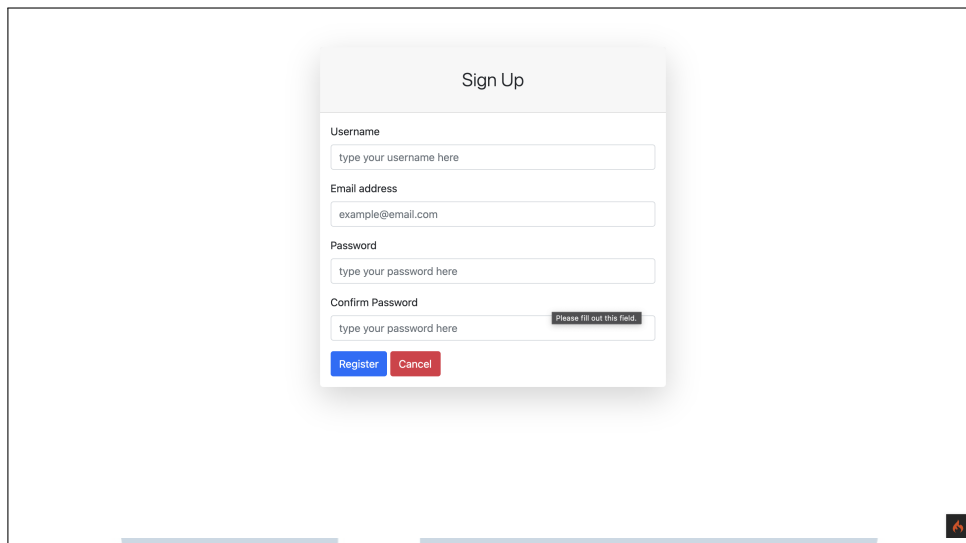
Gambar 3.11. Hasil implementasi halaman *login*

Jika berhasil masuk, maka user akan diarahkan ke halaman *Dashboard*, halaman ini digunakan oleh tim proyek kartu untuk mengunggah dokumen.



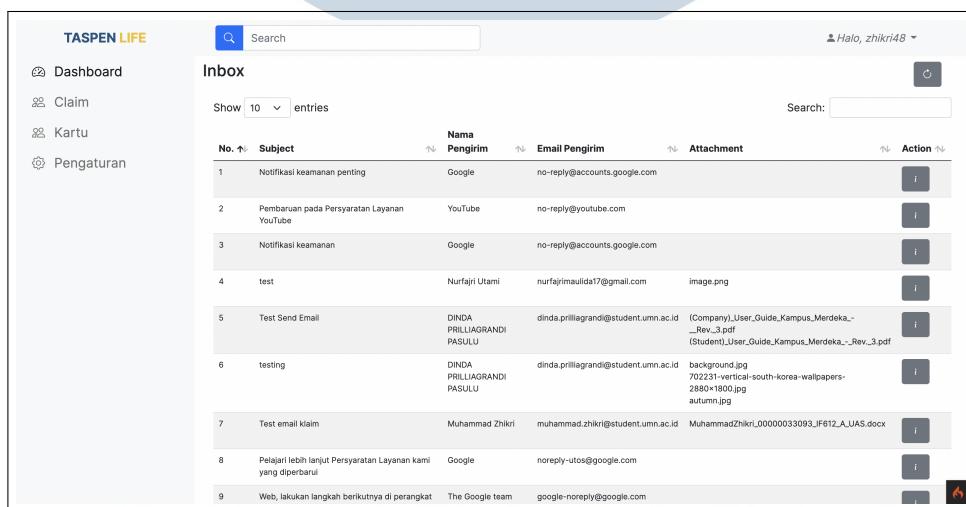
Gambar 3.12. Hasil implementasi halaman *Dashboar*

Setelah berhasil masuk, user diberikan *navigation bar* di sebelah kiri untuk berpindah-pindah halaman. untuk membuat akun baru, *user* dapat berpindah ke halaman Pengaturan dan akan diarahkan ke halaman registrasi akun.



Gambar 3.13. Hasil implementasi halaman *Registration*

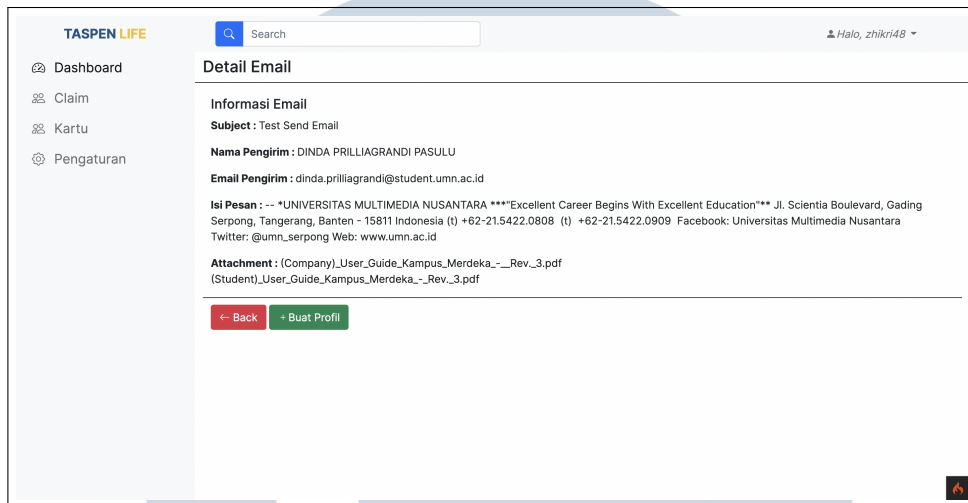
Kolom *username* dan *email* harus unique, jika nama pengguna atau alamat *email* telah digunakan maka akan ada pesan peringatan. kolom kata sandi harus terdiri dari minimal 6 (enam) karakter.



Gambar 3.14. Hasil implementasi halaman *Inbox*

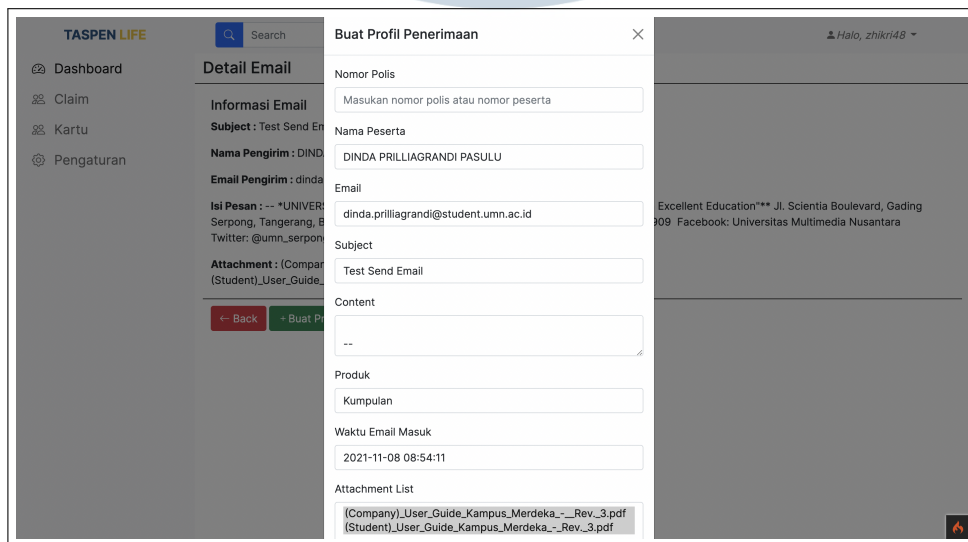
Halaman Inbox akan menampilkan *list* dari pesan yang di kirim ke alamat email yang sudah ditentukan. Untuk dapat mengambil data pesan, akun Gmail harus mengaktifkan pengaturan IMAP, jika tidak maka akan terjadi *error*. selanjutnya data *list* pesan yang sudah diambil akan dimasukkan ke dalam *database* dan akan diambil lagi untuk ditampilkan pada halaman inbox. beberapa informasi penting seperti *subject*, nama pengirim, email pengirim, dan *list attachment* jika ada.

untuk melihat selengkapnya, disediakan tombol action yang akan mengarahkan ke halaman detail sesuai dengan *id* yang dimiliki *message*.



Gambar 3.15. Hasil implementasi halaman *Detail Message*

Pada halaman detail pesan, akan ditampilkan isi pesan dari *email* dan terdapat tombol kembali dan tombol untuk membuat profil penerimaan.



Gambar 3.16. Form untuk membuat profil penerimaan baru

Jika user menekan tombol 'buat profil' maka akan muncul *pop-up* modal yang menampilkan form. user hanya perlu memasukkan kolom yang kosong saja karena kolom lainnya akan diisi secara otomatis sesuai dengan data detail pesan.

No	Notas	Nama Peserta	Email Peserta	Segment Produk	Subject Email	Tanggal Email Diterima	Tanggal Profil dibuat	Action
1	12123	Muhammad Zhikri	muhammad.zhikri@student.umh.ac.id	Kumpulan	Test email klaim	2021-11-04 11:00:25	2021-12-12 22:29:42	
2	0123	Google	no-reply@accounts.google.com	Kumpulan	Notifikasi keamanan penting	2021-11-04 03:49:49	2021-12-12 22:29:15	
3	12112	Nurfajri Utami	nurfajrimaulida17@gmail.com	Kumpulan	test	2021-12-11 11:43:15	2021-12-11 10:45:48	
4	001	DINDA PRILLIAGRANDI PASULU	dinda.prilliagranti@student.umh.ac.id	Individu	testing	2021-12-02 10:30:58	2021-12-07 04:18:25	
5	TSKUJAI700000999	DINDA PRILLIAGRANDI PASULU	dinda.prilliagranti@student.umh.ac.id	Kumpulan	Test Send Email	2021-11-08 08:54:11	2021-12-07 04:17:53	

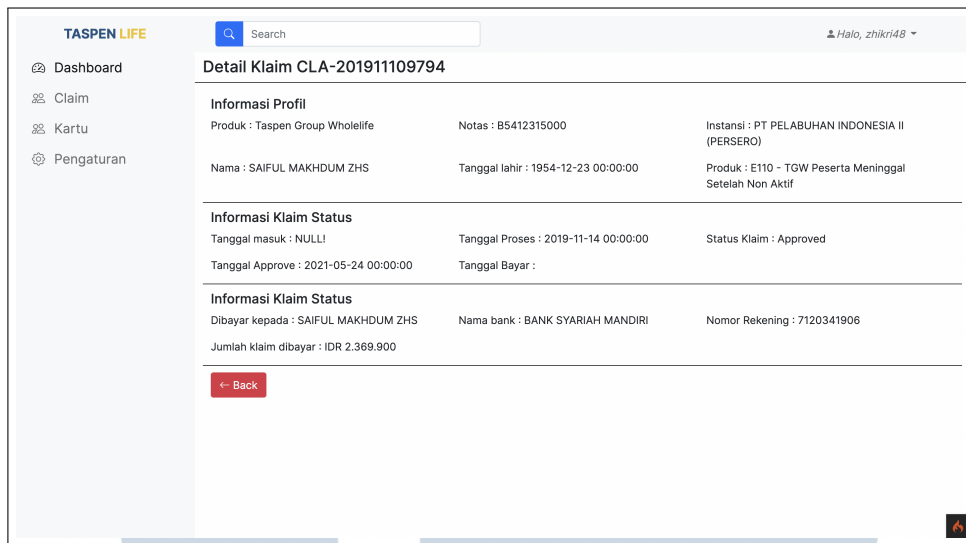
Gambar 3.17. Hasil implementasi halaman *Profil Penerimaan*

Setelah *user* menekan tombol ‘*submit*‘ pada *form* yang tertera pada Gambar 3.15, data yang sudah input akan dimasukkan ke dalam *database* dan akan diambil untuk ditampilkan pada halaman *Profil Penerimaan*.

ID	No. Klaim	Name	Jenis Produk	Tanggal Penerimaan	Klaim Status	Approved By	Action
20748	CLA-1992000824	LAMBERTUS MAMBRAKU	Taspen Group Endowment	1992-04-11 00:00:00	Approved		
20754	CLA-1992000830	BERNADUS HEIPON	Taspen Group Endowment	1992-04-13 00:00:00	Approved		
59972	CLA-2000003494	SUTIADJI	Taspen Group Wholelife	2000-09-07 00:00:00	Approved		
67195	CLA-2002000906	TJAHJONO BIN TAMANG	Taspen Group Wholelife	2002-02-18 00:00:00	Approved		
73843	CLA-2002007554	NI NYOMAN WIDIATI	Taspen Group Endowment	2002-12-20 00:00:00	Approved		
97651	CLA-2005002765	I KETUT NUJADA	Taspen Group Endowment	2005-05-13 00:00:00	Approved		
97653	CLA-2005002767	I WAYAN MARA	Taspen Group Endowment	2005-05-13 00:00:00	Approved		
286044	CLA-20181109794	SAIFUL MAKHDUM ZHS	Taspen Group Wholelife	2018-11-14 10:23:35	Approved	Syahrinal	
302752	CLA-20201126410	ABD. RAHMAN	Taspen Group Wholelife	2020-11-17 10:05:31	Canceled	Syahrinal	
306784	CLA-202102130447	INTAN CINTHIA MIRZADEVI	Taspen Save	2021-02-02 14:13:57	Approved	Syahrinal	

Gambar 3.18. Hasil implementasi halaman *Tracking*

Berikutnya terdapat halaman yang menampilkan list klaim. Struktur table pada tabel klaim yang terdapat di dalam *database* terdiri dari kolom yang sangat banyak, oleh karena itu pada tabel halaman *Tracking*, hanya ditambahkan data-data yang diperlukan saja. Namun, data pada tabel *view* tidak terlalu lengkap, maka dibuatlah tombol yang akan mengarahkan ke halaman detail klaim.



Gambar 3.19. Hasil implementasi halaman detail klaim

Pada halaman detail klaim ini, data-data yang spesifik diperlukan oleh *user* akan ditampilkan. Data yang ada pada halaman ini sesuai dengan *id* klaim yang ingin diakses. Tidak ada fungsi *CRUD* atau *action* apapun pada halaman ini, karena tujuannya hanya untuk melihat detail dari klaim.

3.6. Testing

Pengujian dilakukan secara manual tanpa *tools* pihak ketiga. penulis mencoba keseluruhan aplikasi dengan berbagai kondisi, seperti misalnya mencoba menginput *username* atau *password* yang salah apakah akan tetap masuk dalam kondisi *log in* atau akan memunculkan pesan kesalahan. Namun, sebelum pengujian keseluruhan, untuk beberapa fitur seperti mengambil data dari *email* dan *attachment*-nya, penulis membuat *function test* yang akan menjalankan beberapa *function*, hal ini dilakukan agar lebih mudah untuk menguji fungsi tertentu bekerja dengan sesuai yang diinginkan. Bug yang ditemukan saat pengujian akan segera diperbaiki oleh penulis.

```

public function testSaveMessageAndAttachment()
{
    $this->getTLKlaimMessages();
    $this->saveMessage();
    $this->saveAttachment();
    $this->moveEmailToNewMailbox();
}

```

Gambar 3.20. Fungsi *test* untuk menyimpan aplikasi

3.7. Kendala yang ditemukan

Dalam proses pembuatan proyek aplikasi *Tracking* dan *Monitoring* klaim ini terdapat beberapa kendala yang dihadapi, yaitu sebagai berikut:

1. Penulis tidak terlalu familiar dengan *framework* CodeIgniter 4. CodeIgniter 4 merupakan *framework* terbaru di mana cara penggunaannya berbeda dengan CodeIgniter 3.
2. Penulis kesulitan untuk mendapatkan data dari Inbox Gmail, hal ini disebabkan protokol yang digunakan untuk mengambil data email adalah IMAP (*Internet Message Access Protocol*), Bukan POP3, maupun SMTP.
3. Penulis kesulitan untuk membuat profil penerimaan di mana kolom yang tersedia akan terisi secara otomatis menyesuaikan dengan data yang ada pada pesan.

3.8. Solusi Atas Kendala yang Ditemukan

Setiap mendapatkan permasalahan, penulis akan berdiskusi bersama tim proyek aplikasi kartu dan *supervisor*. Berikut ini merupakan solusi yang penulis dapatkan setelah melakukan diskusi:

1. Penulis diberi waktu untuk beradaptasi dan mempelajari bagaimana cara menggunakan CodeIgniter 4. Penulis menjadikan dokumentasi dari website resmi CodeIgniter sebagai sumber belajar.
2. Penulis menggunakan *library* PHP IMAP yang dibuat oleh seorang pengguna Github dengan nama pengguna Ddeboer. Sebelumnya penulis sudah mencoba berbagai *library* lainnya namun tidak dapat berjalan.
3. Penulis memasukkan semua pesan yang diambil dari Gmail ke dalam database (ini juga atas saran *supervisi*). Sehingga akan mempermudah pengambilan data dan *binding message* dengan *attachment*. Waktu waktu yang digunakan untuk *me-load* halaman juga lebih cepat ketimbang menampilkan data yang diambil secara *direct* dari Gmail menggunakan IMAP.