BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Pengguna *email* di Indonesia sangat banyak dan webmail yang menyediakan layanan *email* gratispun sangat banyak, sehingga para pengguna *email* di Indonesia terdaftar di banyak webmail seperti: yahoo! mail, gmail, hotmail dan masih banyak lagi webmail yang menyediakan layanan *email* gratis. Akan tetapi dengan banyaknya webmail, tidak diikuti dengan layanan push mail seperti outlook express yang hanya menyediakan untuk satu *email* saja. Layanan push mail digunakan agar email yang masuk dan keluar dapat diakses secara *realtime*, tanpa harus membuka alamat webmail penyedia email, sehingga pengguna email dapat mengirim dan menerima email secara cepat dan efisien.

Oleh karena itu dibutuhkan sebuah aplikasi untuk dapat memenuhi banyak email dari banyak webmail serta dapat di *push* melalui yahoo messenger secara *realtime* dan dapat diakses dimana saja dan kapan saja. Dengan menggunakan yahoo messenger sebagai perantara dari aplikasi ini, email dapat di *push* secara *realtime.* Di samping itu yahoo messenger merupakan aplikasi IM (*instant messenger*) paling popular di Indonesia serta dapat menghemat pengeluaran pulsa dibandingkan apabila aplikasi ini berjalan pada *sms gateway*, karena sms gataway memberlakukan pulsa premium serta tidak dapat berjalan secara *realtime.*

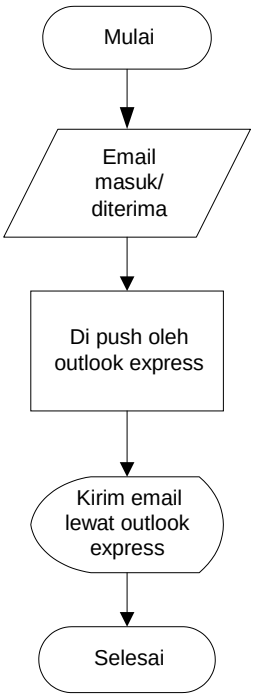
Dengan membangun aplikasi ini maka *email* yang masuk dapat dibuka langsung oleh yahoo messenger serta dapat menerima dan mengirim pesan ke banyak *email* yang bisa diakses melalui komputer, laptop, maupun perangkat mobile secara *realtime.*

Layanan push mail saat ini merupakan sebuah layanan yang disediakan oleh *developer* pihak ketiga, dimana sebuah email dapat di *push* agar dapat dikelola oleh penggunanya tanpa harus masuk ke webmail langsung. Sebagai
contoh *Microsoft office outlook express* merupakan sebuah perangkat lunak pihak

ketiga yang menyediakan layanan push mail ini. Berikut adalah bagaimana skenario push mail pada umumnya berjalan:

1. Email masuk dari webmail
2. Kemudian di push oleh outlook express
3. Diterima oleh user dan dikelola (*read, filter, delete, etc*)
4. Oleh user email dibalas atau kirim email melalui outlook express.

Untuk lebih jelasnya dapat dilihat pada *flowchart* di bawah ini:



Gambar 3.1 Flowchart Cara Kerja Push Mail

Adapun beberapa kelemahan layanan push mail yang ada saat ini adalah sebagai berikut:

1. Layanan push mail yang disediakan oleh outlook express hanya untuk satu email saja.
2. Pengaksesan email yang masuk melalui outlook express tidak bisa berjalan secara *realtime*.
3. Hanya bisa diakses dalam PC (*Personal Computer*), artinya tidak bisa diakses dalam perangkat *mobile,* kecuali perangkat *mobile* khusus seperti *blackberry*, *iphone*, dan *android* itupun bukan menggunakan outlook express melainkan aplikasi pihak ketiga dari masing-masing *developer* perangkat *mobile* tersebut.

3.1.1 Analisis Sistem Yang Sedang Berjalan

Dalam tahapan analisis prosedur yang sedang berjalan akan dijelaskan cara kerja email, instant messenger serta push mail agar nantinya diketahui kekurangan dan kelebihan cara kerjanya sehingga perangkat lunak yang dibangun dapat menutupi kekurangan yang ada.

3.1.1.1 Analisis Prosedur Email

Email adalah singkatan untuk Electronic Mail, atau dalam bahasa Indonesia berarti surat elektronik. Surat elektronik sudah mulai dipakai di tahun

1960-an. Pada saat itu Internet belum terbentuk, yang ada hanyalah kumpulan *mainframe* yang terbentuk sebagai jaringan. Mulai tahun 1980-an, surat elektronik sudah bisa dinikmati oleh khalayak umum. Sekarang ini banyak perusahaan pos di berbagai negara menurun penghasilannya disebabkan masyarakat sudah tidak memakai jasa pos lagi.

Etika dalam surat elektronik sama dengan etika dalam menulis surat biasa. Ada surat elektronik yang isinya formal ada yang informal. Beberapa poin penting yang harus diperhatikan dalam mengirim email:

1. Jangan mengirim surat elektronik dengan lampiran (*attachment*) yang terlalu besar (lebih dari 512 kB). Tidak semua orang mempunyai akses Internet yang cepat, dan ada kemungkinan lampiran tersebut melebihi kapasitas surat elektronik penerima, sehingga akan ditolak mailserver penerima. Selain itu, perhatikan juga bahwa beberapa penyedia surat elektronik juga menerapkan batasan tentang jumlah, jenis, dan ukuran surat elektronik yang dapat diterima (dan dikirim) penggunanya.
2. Jangan mengirim lanjut (forward) surat elektronik tanpa berpikir kegunaan bagi orang yang dituju. Selalu isi kolom subjek, jangan dibiarkan kosong. Dalam mengutip tulisan orang lain, selalu usahakan mengutip seperlunya, jangan mau gampangnya mengutip seluruh tulisan orang itu: dalam menjawab surat elektronik orang lain, kutip bagian yang kita tanggapi saja, selain lebih jelas juga tidak memakan waktu / jatah akses penerima, dalam mengutip tulisan orang ketiga, ingat hak cipta: kutip sesedikit mungkin dan rujuk ke tulisan aslinya.
3. Jangan menggunakan huruf kapital karena dapat menimbulkan kesan anda berteriak.

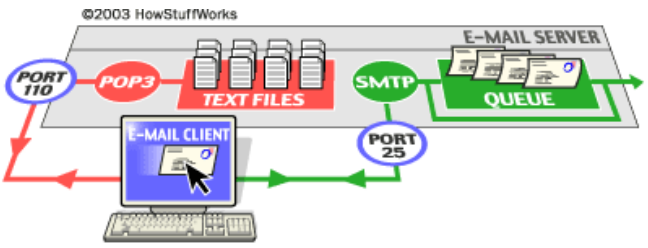
4. Gunakan kata-kata dengan santun. Adakalanya sesuatu yang kita tulis akan terkesan berbeda dengan apa yang sebetulnya kita maksudkan.

Ada 3 jenis tujuan pengiriman yaitu *To*, *Cc* dan *Bcc*. Penjelasannya sebagai berikut:

1. *To*: tujuan standar pengalamatan pesan.
2. *Cc*: singkatan dari *Carbon Copy*. Jika ingin mengirim pesan kepada seseorang, tetapi ingin juga memberikan *copy* pesan pada orang lain.
3. *Bcc*: singkatan dari *Blind Carbon Copy*. Jika ingin mengirim pesan pada 2 orang, pada kolom *To* dan pada kolom *Bcc*, keduanya akan menerima Email tetapi penerima yang ada di kolom *To* tidak akan mengetahui (*blind*) bahwa email juga dikirim kepihak ke 2 (*Bcc*).

Setelah menentukan bagaimana dan ke mana pesan akan dikirim, berikut adalah cara kerja email:

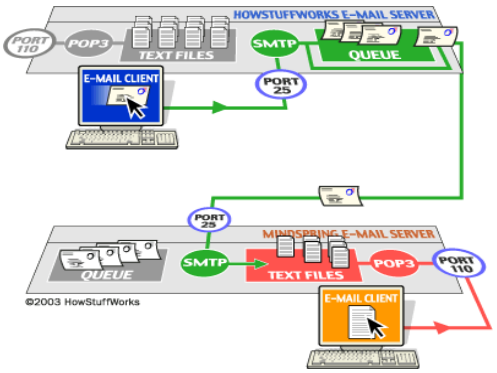
Pada *email server* terdapat dua *server* yang berbeda yaitu *incoming* dan *outgoing server*. *server* yang biasa menangani *outgoing email* adalah *server SMTP* (*Simple Mail Transfer Protocol*) pada *port 25* sedangkan untuk menangani *incoming email* adalah *POP3* (*Post Office Protcol*) pada *port 110* atau *IMAP* (*Internet Mail Access Protocol*) pada *port 143*. Untuk lebih jelasnya dapat dilihat pada gambar berikut:



Gambar 3.2 Cara Kerja Email Ketika dikirim Oleh *Client*

Saat mengirim *email* maka *email* akan ditangani oleh *SMTP server* dan akan dikirim ke *SMTP server* tujuan, baik secara langsung maupun melalui beberapa *SMTP server* dijalurnya. apabila *server* tujuan terkoneksi maka *email* akan dikirim, namun apabila tidak terjadi koneksi maka akan dimasukkan ke dalam *queue* (tempat penyimpanan sementara *server*) dan akan di *resend* setiap 15 menit. Apabila dalam 5 hari tidak ada perubahan maka akan diberikan *undeliver notice* ke *inbox* pengirim.

Apabila *email* terkirim, maka email akan masuk pada *POP3 server* atau *IMAP server*. jika menggunakan *POP3 server* maka apabila kita hendak membaca *email* maka *email* pada *server* di *download* sehingga *email* hanya akan ada pada mesin yang men-*download email* tersebut, dengan kata lain hanya bisa membaca *email* tersebut pada *device* yang mendownload email tersebut. berbeda dengan *POP3 IMAP server* mempertahankan *email* pada *server* sehingga *email* dapat dibuka kembali lewat *device* yang berbeda. Untuk lebih jelasnya dapat dilihat pada gambar di bawah ini:

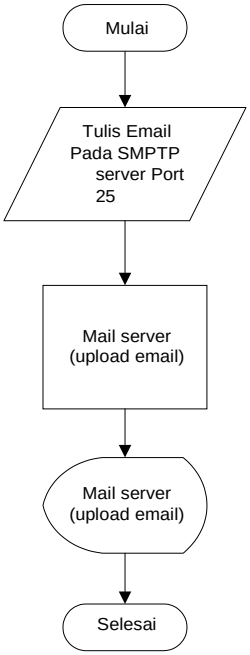


Gambar 3.3 Cara Kerja Email Ketika Dikirim Oleh Mail Server Dan Diterima Oleh Client.

Berikut skenario pengiriman email pada umumnya:

1. User pengirim mempersiapkan email yang akan dikirimkan kemudian email dikirim oleh user pengirim.
2. Email dikirim melalui SMTP server pada port 25.
3. Email diterima oleh mail server.
4. Email berhasil dikrim.

Untuk lebih jelasnya dapat dilihat dalam flowchart di bawah ini:

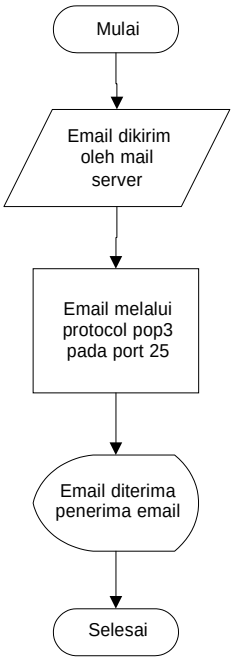


Gambar 3.4 Flowchart Pengiriman Email Pada Umumnya

Berikut adalah skenario penerimaan email:

1. Mail server mengirimkan email ke penerima.
2. Email melalui protocol pop3 pada port 25.
3. Email diterma oleh penerima email.

Untuk lebih jelasnya dapat dilihat pada flowchart di bawah ini:



Gambar 3.5 Flowchart Penerimaan Email Pada Umumnya

3.1.1.2 Analisis Prosedur Instant Messenger

Jaringan *instant messenger* terdiri dari *clients* dan *server*. *User* meng- *install* sebuah *client* yang menghubungkan ke *server* yang dioperasikan oleh vendor jaringan IM (*Instant Messenger*), seperti Yahoo Messenger, Windows Live Messenger atau AOL. *User* yang mendaftar ke *vendor* IM diberikan ID (Identitas Diri) yang unik, yang dapat berupa angka dan huruf. *User* lalu memberikan ID tersebut ke *user* lain yang ingin diajak untuk berkomunikasi melalui jaringan IM.

User memulai sebuah sesi IM dengan autentifikasi ke *server*. Ketika dua *user* yang sudah melakukan autentifikasi ingin berkomunikasi, rangkaian prosesnya adalah:

1. *User* A menginstruksikan IM *client* untuk mengirimkan pesan teks kepada *User* B. *Client* membuat sebuah paket yang berisi pesan dan mengirimkannya ke *server*.*Server* menerima isi paketnya dan melihat bahwa si penerima adalah *User* B.
2. *Server* lalu membuat sebuah paket baru yang mengandung pesn dari *User* A dan mengirimkannya ke *User* B.

Kebanyakan IM akan melanjutkan pengiriman pesan melalui *server* pusat. Bagaimanapun, beberapa IM membuat koneksi langsung antara user yang berkomunikasi setelah pesan pertama.Penggunaan *server* pusat lebih menguntungkan jika dilihat dari berbagai sudut pandang. Contohnya: User A hanya membutuhkan *unique* ID dari *User* B, selain itu, dia dapat mengirimkan pesan kepada *User* B meski dia tidak *online*. *Server* akan menyimpan pesan sampai *User* B melakukan autentifikasi.

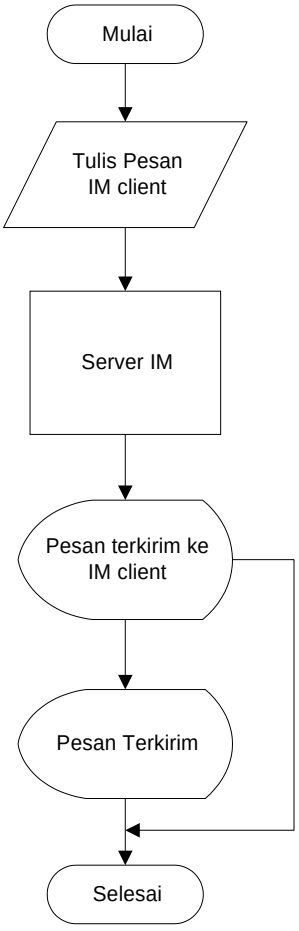
Kebanyakan IM *client* mempunyai kemampuan untuk membuat Buddy List, yaitu daftar unique ID teman-teman kita, sehingga akan mempermudah dalam berkomunikasi dengan mereka. Contohnya, ketika Bill mengirimkan Alice unique IDnya, Alice dapat menyimpannya dalam Buddy List. Sehingga, kapanpun Alice log in ke server IM, dia dapat melihat status dari Bill, apakah online atau tidak.

Berikut skenario pengiriman *instant messenger* pada umumnya:

1. Pengirim mengirimkan pesan singkat (*IM*).
2. Kemudian disimpan oleh *IM client* (*YM, AOL*, dll) dan diteruskan ke *server*

   IM.
3. Dari *server IM*, pesan diteruskan ke *IM client* penerima.
4. Setelah sampai pada *IM* clent penerima, pesan dapat dibaca oleh penerima pesan *IM*.

Untuk lebih jelasnya dapat dilihat dalam flowchart di bawah ini:



Gambar 3.6 Flowchart Pengiriman *Instant Messenger* (*IM*)

3.1.2 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional merupakan bagian untuk menspesifikasikan ukuran yang harus dipenuhi oleh perangkat lunak.

Untuk lebih jelasnya dapat dilihat dalam tabel 3.1 di bawah ini.

Table 3-1 Analisis Kebutuhan Non Fungsional

Kriteria Kebutuhan	Tuntutan Kebutuhan
Performansi	Perangkat PC/Laptop atau mobile untuk menjalankan aplikasi ini harus selalu terkoneksi ke internet
Batasan Memori	Perangkat PC/Laptop: 1. Prosessor minimum dual core 2. Hardddisk minimum 80GB 3. RAM Minimun 256MB memory Perangkat Mobile: 1. Memory internal minimum 10MB
Modus Operasi	Perangkat PC/Laptop: 1. windows XP, Vista atau windows 7 2. sudah terpasang aplikasi yahoo messenger versi 8 atau lebih baru Perangkat Mobile: 1. Minimal sudah mendukung aplikasi Java 2. Sudah terpasang aplikasi yahoo messenger atau sejenis aplikasi pemanggil YM, seperti: ebuddy, nimbuzz, dll
Kebutuhan Adaptasi Lokasi	-

3.1.2.1 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam menjalankan perangkat lunak ini adalah sebagai berikut:

1. System operasi windows xp or vista or 7
2. J2SDK (Java2 Software Development Kit) versi 1.6.
3. Xampp win32 v.1.7.1.
4. Yahoo Messenger 8.0 atau lebih baru.

3.1.2.2 Analisis Kebutuhan Perangkat Keras

Berikut adalah spesifikasi minimum untuk menjalankan perangkat lunak ini baik di computer maupun di perangkat mobile:

1. Intel dual core prosesor 1,5Ghz
2. Memory 512MB
3. VGA 256MB shared
4. Harddisk 10MB space
5. Monitor
6. Mouse and keyboard
7. Perangkat mobile yang sudah terpasang yahoo messenger

3.1.2.3 Analisis Kebutuhan Pengguna

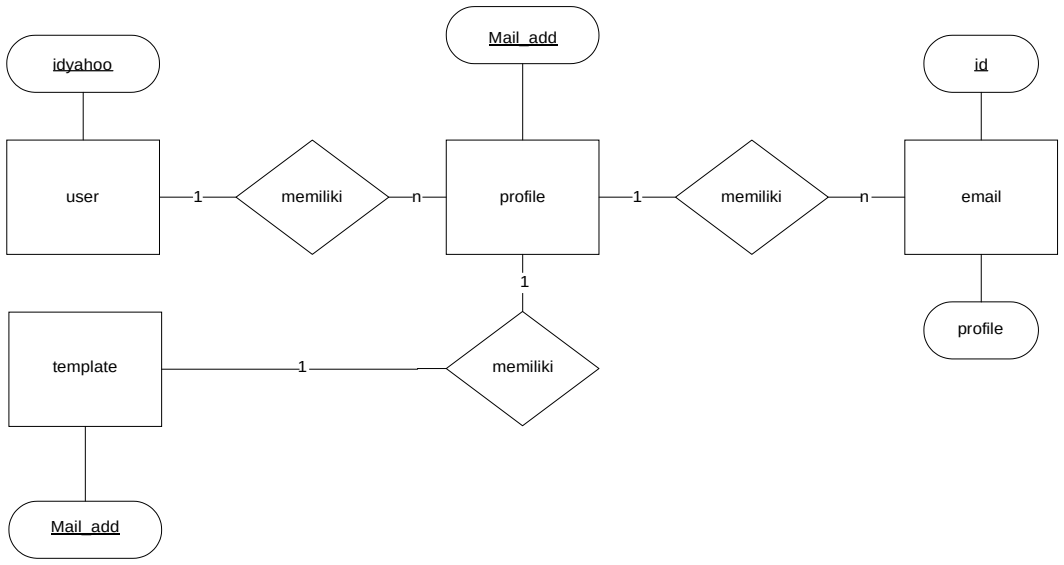
Untuk dapat menjalankan aplikasi ini, pengguna harus memenuhi persyaratan sebagai berikut:

1. Mengerti operasional perangkat mobile berbasis java.
2. Mengerti penggunaan yahoo messenger.
3. Mengerti secara umum penggunaan internet.
4. Memiliki perangkat mobile yang sudah terpasang YM.

3.1.2.4 Analisis Data

Analisis data meliputi *Entity relationship Diagram (ERD)* dan *Data Flow Diagram (DFD)*. *Entity relationship diagram* menjelaskan kardinalitas antara antara entitas-entitas dan menjelaskan pula relasi antar entitas-entitas yang berada pada perangkat yang dibangun. *Data flow diagram* merupakan alat yang dapat menggambarkan suatu proses di dalam system secara terstruktur dan jelas, serta dokumentasi dari perangkat lunak yang baik.

Entity relationship diagram (*ERD*) merupakan model konseptual yang mendeskripsikan hubungan antar penyimpanan. *ERD* digunakan untuk memodelkan struktur data dan hubungan antar data. Dengan *ERD*, model dapat diuji dengan mengabaikan proses yang dilakukan. Model data dari system yang akan dibangun dapat dilihat dari gambar di bawah ini:



Gambar 3.7 ERD Aplikasi Push Mail pada Yahoo Messenger

Adapun entitas-entitas lain yang terdapat dalam ERD diatas dapat dilihat pada Table 3-2:

Table 3-2 Entitas dan Atribut pada ERD

Entitas	Atribut	Keterangan
user	1. Idyahoo 2. interval_check 3. always_send 4. send_only_header	1. Nama ID yahoo 2. Jarak waktu cek email 3. Selalu kirim email 4. Pengiriman header email
profile	1. mail_add 2. owner 3. in_protocol 4. in_username 5. in_pass 6. in_server 7. in_port 8. in_ssl 9. out_server 10. out_port 11. out_ssl 12. out_auth 13. out_username 14. out_pass	1. Alamat email 2. Pemilik email 3. Nama protokol masuk email (SMTP) 4. Alamat email 5. Password email 6. Nama server email masuk 7. Nomor port protokol email masuk 8. Konfigurasi keamanan SSL email masuk 9. Nama server email keluar 10. Nomor port protokol email keluar 11. Konfigurasi keamanan SSL email keluar 12. Otentikasi / pembuktian protokol email keluar 13. Alamat email 14. Password email
email	1. id 2. profile 3. subject 4. from 5. to 6. send_date 7. mail_id 8. content 9. unread	1. Nama id yahoo 2. Nama profile 3. Subjek email 4. Dari siapa email masuk atau keluar 5. Untuk siapa email masuk atau keluar 6. Tanggal email terkirim 7. Alamat email 8. Isi dari email 9. Email yang belum terbaca

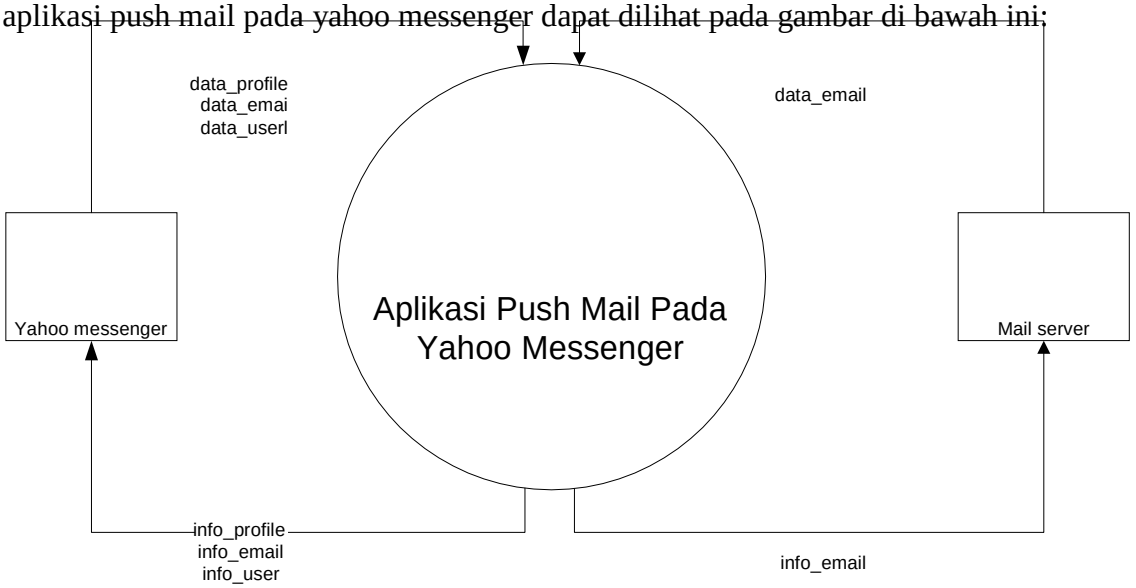
template	1. mail_addr	1. Alamat email
	2. in_protocol	2. Jenis incoming protocol
	3. in_server	3. Alat server incoming protocol
	4. in_port	4. Port number incoming protocol
	5. in_ssl	5. Keamanan incoming protocol
	6. out_server	6. Jenis outgoing protocol
	7. out_port	7. Port number outgoing protocol
	8. out_ssl	8. Keamanan outgoing protocol
	9. out_auth	9. Autentikasi keamanan

3.1.3 Analisis Kebutuhan Fungsional

Analsisi kebutuhan fungsional mendefinisikan aksi dasar yang ada dalam perangkat lunak yang dibangun untuk menerima dan memproses masukan dan menghasilkan keluaran.

3.1.3.1 Diagram Konteks

Diagram konteks adalah diagram yang menggambarkan masukan, proses dan keluaran secara umum yang terjadi pada sistem. Diagram konteks untuk
aplikasi push mail pada yahoo messenger dapat dilihat pada gambar di bawah ini;



Gambar 3.8 Diagram Konteks Aplikasi Push Mail Pada Yahoo Messenger

3.1.3.2 DFD Level 1

Pada DFD level 1 akan dibagi menjadi dua garis besar proses utama, yaitu pengolahan data profile dan pengolahan data email.

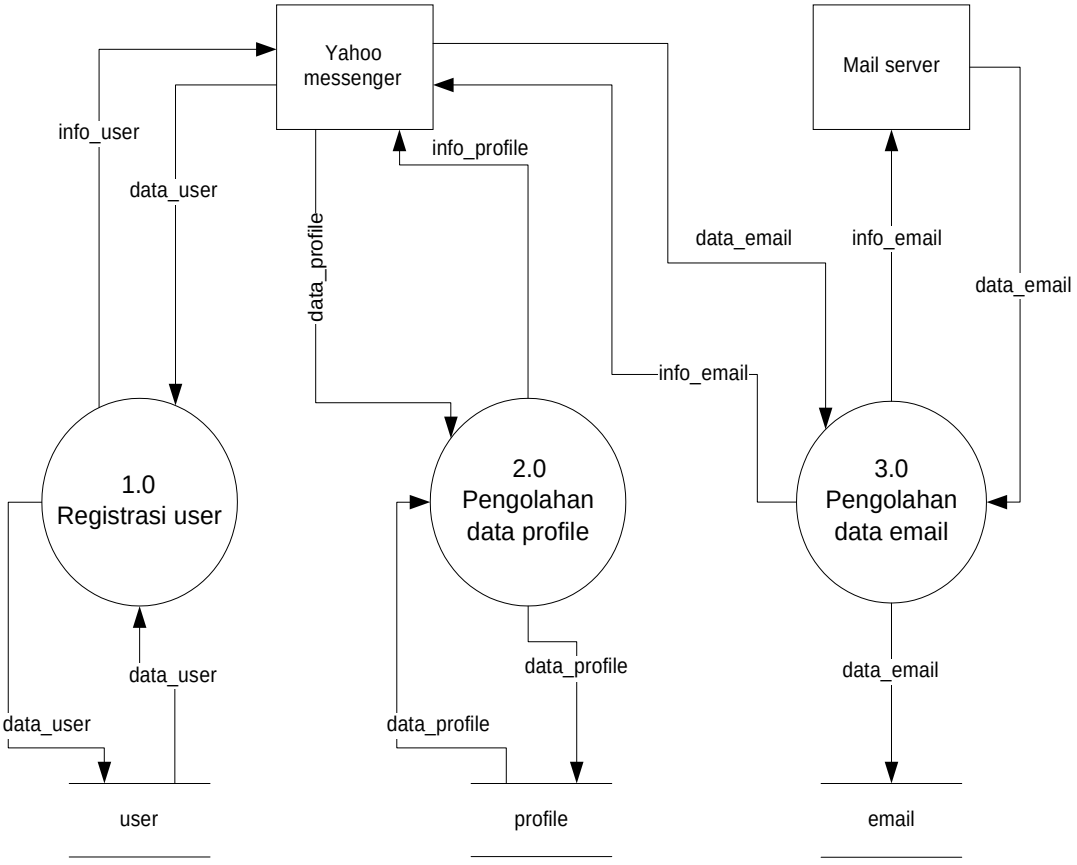
1. Pengolahan Data Profile

Pada proses pertama ini yahoo messenger dapat mengolah data profile seperti add profile, delete profile dan view profile. Data profile yang ditambahkan akan masuk ke tabel profile begitu pula data yang dilihat dan dihapus akan diapanggil dari tabel profile.

2. Pengolahan Data Email

Pengolahan data email memungkinkan yahoo messenger untuk mengirim dan menerima email dari mail server yang kemdian disimpan pada tabel email.

Untuk lebih jelasnya dapat dilihat dalam gambar di bawah ini:

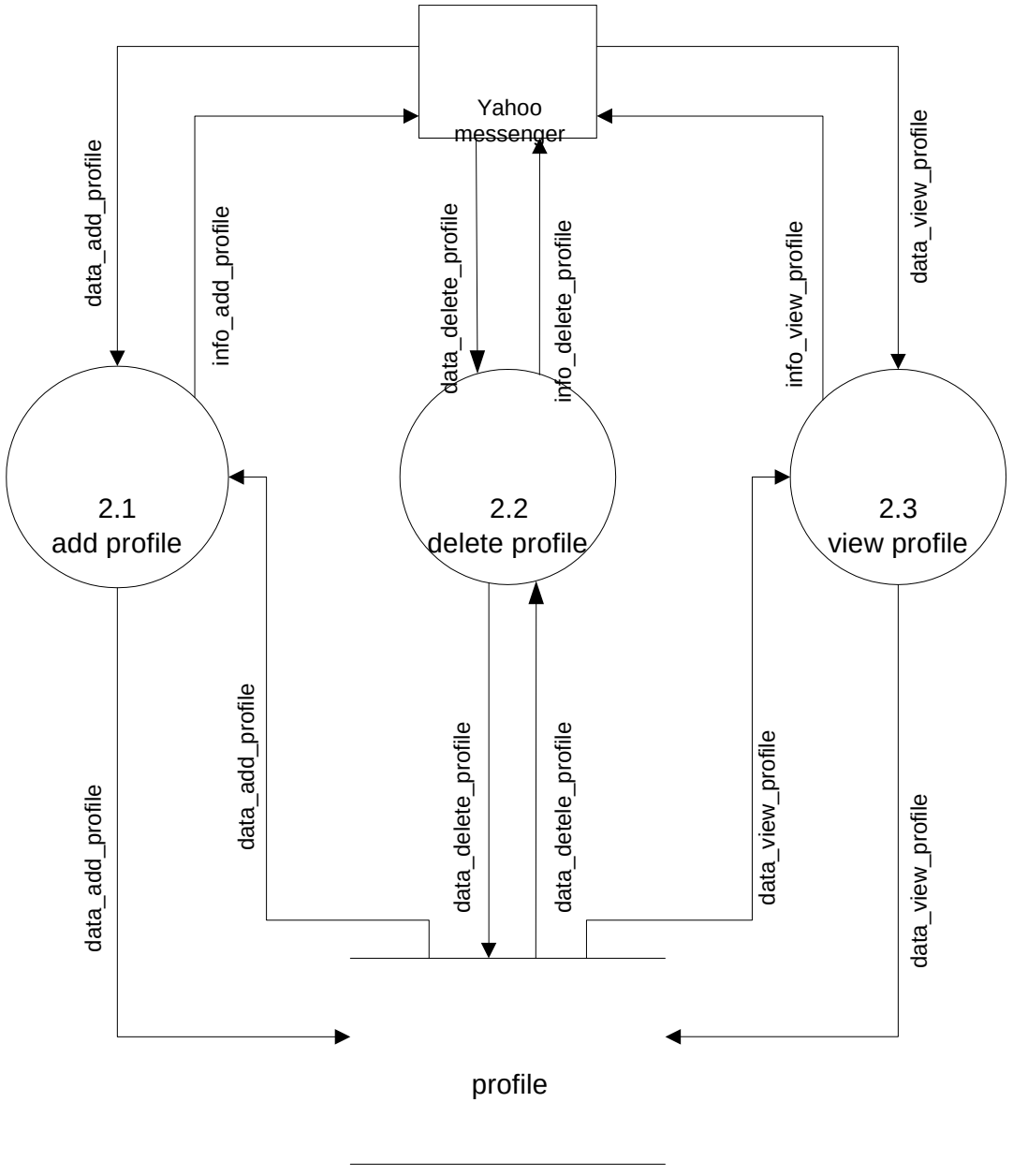


Gambar 3.9 DFD Level 1

3.1.3.3 DFD Level 2 Pengolahan Data Profile

Pada DFD level 2 pengolahan data profile yahoo messenger dapat mengolah data profile seperti add profile, delete profile dan view profile. Untuk

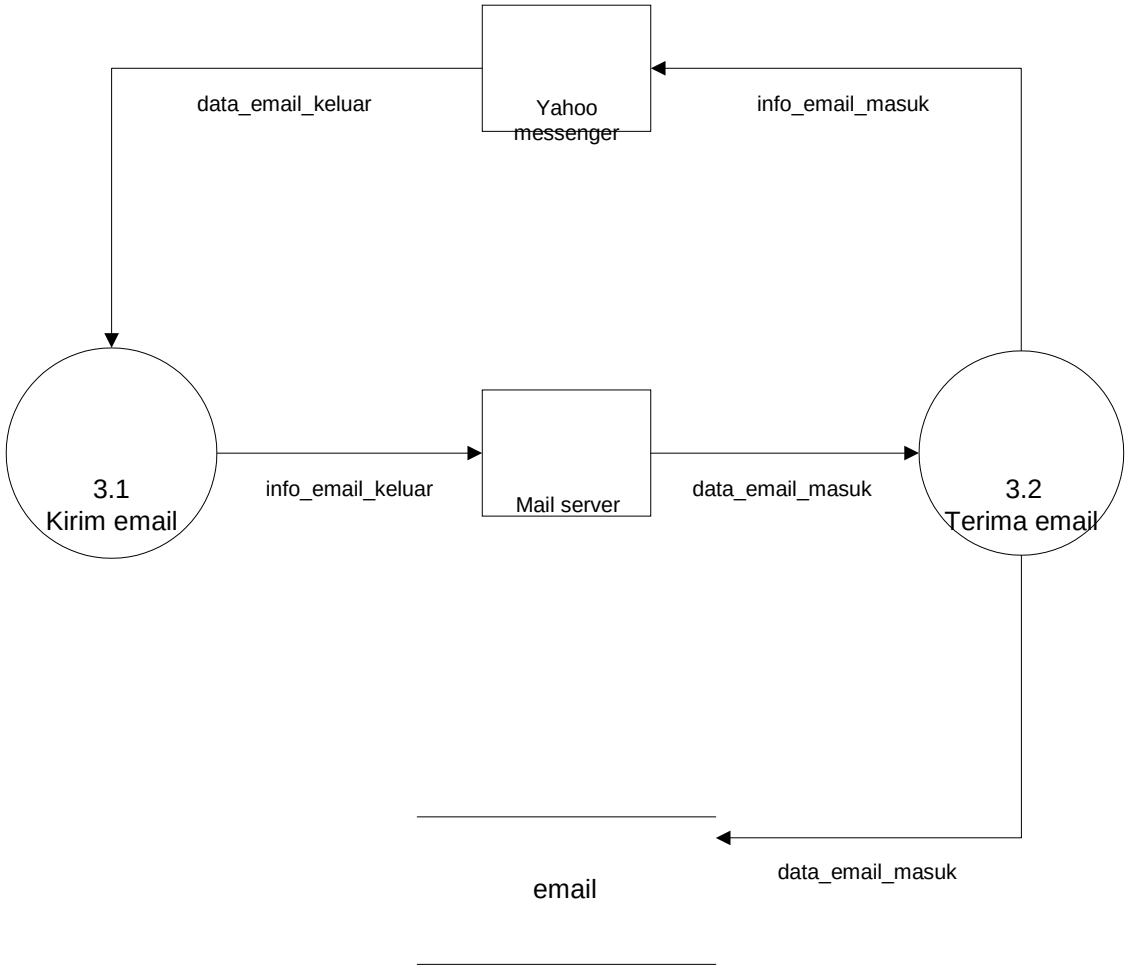
lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 3.10 DFD Level 2 Pengolahan Data Profile

3.1.3.4 DFD Level 2 Pengolahan Data Email

Pada DFD level 2 pengolahan data email yahoo messenger dapat mengolah data email seperti menerima email masuk dan mengirim email. Untuk
lebih jelasnya dapat dilihat pada gambar di bawah ini:



Gambar 3.11 DFD Level 2 Pengolahan Data Email

Berikut adalah spesifikasi proses dari DFD untuk menjelaskan perilaku-perilaku yang ada dalam DFD secara lebih rinci. Spesifikasi proses pada perangkat lunak push mail pada yahoo messenger dapat dilihat pada Table 3-3:

Table 3-3 Spesifikasi Proses DFD Level 1

NO	Spesifikasi	Keterangan
1.	No	1.0
	Nama	Registrasi user
	Input	data_user
	Output	info_profile
	Tujuan	Tabel user
	Logika proses	1. YM akan menambahkan user id YM aplikasi. 2. Sistem akan menyimpan id YM pada tabel user.
2.	No	2.0
	Nama	Pengolahan Data Profile
	Input	data_profile
	Output	info_profile
	Tujuan	Tabel profile
	Logika proses	3. YM akan memasukan profile baru dan protocol email sesuai web mail yang dimasukan. 4. Sistem akan menyimpan konfigurasi yang dilakukan oleh YM.
3.	No	3.0
	Nama	Pengolahan Data Email
	Input	data_email
	Output	data_email
	Tujuan	Tabel email
	Logika proses	1. YM menerima email dari webmail server. 2. Email dapat dikirim dan diterima melalui YM. 3. YM mengirimkan email ke webmail.
4.	No	2.1
	Nama	add profile
	Input	data_add_profile
	Output	info_profile
	Tujuan	Tabel profile

	Logika proses	1. YM akan melakukan konfigurasi dengan memasukan profile baru dan protocol email sesuai web mail yang dimasukan. 2. Sistem akan menyimpan konfigurasi yang dilakukan oleh YM.
5.	No	2.2
	Nama	Delete profile
	Input	Data_delete_profile
	Output	Info_delete_profile
	Tujuan	Tabel profile
	Logika proses	1. YM menghapus salah satu profile yang ada pada tabel profile 2. Sistem akan menghapus profile yang telah diperintahkan oleh YM
6.	No	2.3
	Nama	View profile
	Input	Data_view_profile
	Output	Info_view_profile
	Tujuan	Tabel profile
	Logika proses	1. YM melakukan perintah untuk melihat salah satu profile. 2. Sistem akan menampilkan semua detail dari profile yang diperintahkan oleh YM
7.	No	3.1
	Nama	Kirim email
	Input	Data_email_keluar
	Output	Info_email_keluar
	Tujuan	Mail server
	Logika proses	1. YM menulis pesan email 2. Sistem mengirimkan pesan tersebut ke mail server
8.	No	3.2

	Nama	Terima email
	Input	Data_email_masuk
	Output	Info_email_keluar
	Tujuan	Tabel email
	Logika proses	1. Mail server mengirimkan email. 2. Sistem menerima email masuk. 3. YM menerima email dari sistem.

3.1.3.5 Kamus Data

Kamus data membantu mendefinisikan dengan lengkap data yang mengalir diatara proses, penyimpanan data, dan entitas luar. Data yang mengalir tersebut dapat berupa masukan untuk sistem atau hasil dari proses sistem. Kamus data dibuat berdasarkan arus data yang mengalir pada konteks diagram dan DFD.

Table 3-4 Kamus Data

NO	Data	Keterangan
1.	**Nama**	data_profile
	Deskripsi	Berisi data profile
	Struktur data	email_add + password + in_protocol + in_username + in_pass + in_server + in_port + in_ssl + out_server + out_port + out_ssl + out_auth + out_username +
	email_add password	[A-Z|a-z|0-9] [A-Z|a-z|0-9]
	in_protocol	[A-Z|a-z]
	in_username	[A-Z|a-z|0-9]
	in_pass	[A-Z|a-z|0-9]
	in_server	[A-Z|a-z]
	in_port	[0-9]
	in_ssl	[true|false]
	out_server	[A-Z|a-z]

	out_port	[0-9]
	out_ssl	[true|false]
	out_auth	[true|false]
	out_username	[A-Z|a-z|0-9]
	out_pass	[A-Z|a-z|0-9]
2.	**Nama**	Data_email
	Deskripsi	Email yang masuk dan keluar
	Struktur data	Id + Profile + Subject + From + To + Send_date + Mail_id
	Id	[A-Z|a-z|0-9]
	Profile	[A-Z|a-z|0-9]
	Subject	[A-Z|a-z|0-9]
	From	[A-Z|a-z|0-9]
	To	[A-Z|a-z|0-9]
	Send_date	[datetime]
	Mail_id	[A-Z|a-z|0-9]

3.2 Perancangan Sistem

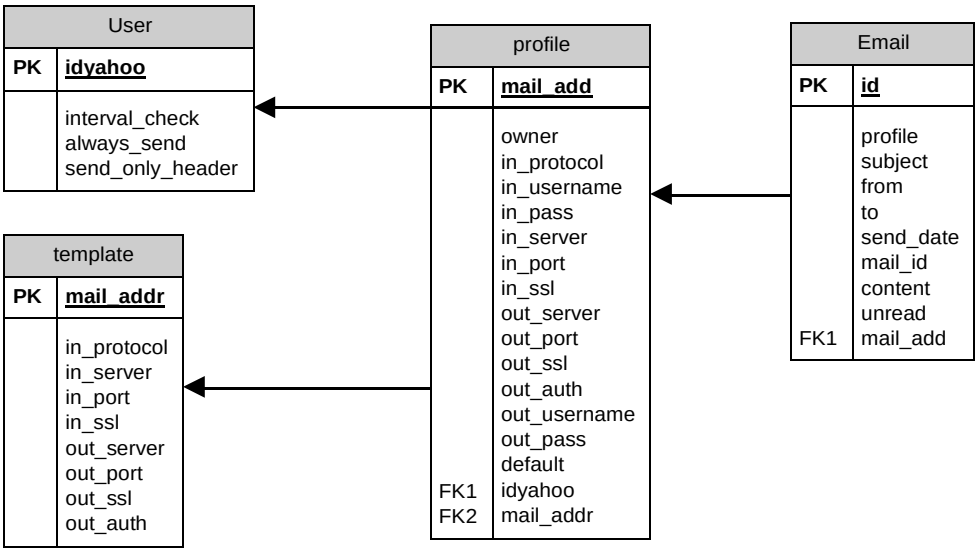
Perancangan merupakan tahap lanjutan setelah tahap analisis perangkat lunak dimana pada perancangan perangkat lunak digambarkan rancangan perangkat lunak yang akan dibangun sebelum dilakukan pengkodean kedalam bahasa pemrograman.

3.2.1 Perancangan Data

Perancangan data terdiri dari perancangan diagram relasi berdasarkan analisis basis data dan struktur table yang dibangun dalam sistem.

3.2.1.1 Diagram Relasi

Diagram relasi merupakan relasi khusus antar table yang berada dalam perangkat lunak yang dibangun. Berikut diagram relasi yang berada dalam perangkat lunak push mail pada yahoo messenger:



Gambar 3.12 Diagram Relasi

3.2.1.2 Struktur Tabel

Pada tabel-tabel di bawah ini adalah struktur tabel yang terdapat pada

perangkat lunak push mail pada yahoo messenger:

Table 3-5 Struktur Tabel User

Nama Field	Tipe	Panjang Karakter	Kunci	Keterangan
idyahoo	varchar	35	Primary key	nama id yahoo
interval_check	integer	1	-	-
always_send	integer	1	-	-
send_only_header	integer	1	-	-

Table 3-6 Struktur Tabel Profile

Nama Field	Tipe	Panjang Karakter	Kunci	Keterangan
mail_add	varchar	35	Primary key	alamat email
owner	varchar	35	-	nama pemilik mail
in_protocol	varchar	4	-	protokol email masuk
in_username	varchar	35	-	username protokol masuk

in_pass	text		-	password protokol masuk
in_server	varchar	35	-	nama server protokol mail masuk
in_port	integer	11	-	port protokol email masuk
in_ssl	integer	1	-	-
out_server	varchar	35	-	nama server protokol email keluar
out_port	integer	11	-	port number protokol email keluar
out_ssl	integer	1	-	-
out_auth	integer	1	-	-
out_username	varchar	35	-	username protokol email keluar
out_pass	text		-	password protokol email keluar

Table 3-7 Struktur Tabel Email

Nama Field	Tipe	Panjang Karakter	kunci	Keterangan
id	varchar	20	Primary key	Nama id yahoo
profile	varchar	50	-	Nama email add
subject	varchar	125	-	Subjek email
from	varchar	75	-	alamat email yg masuk
to	varchar	350	-	Alamat email tujuan
send_date	datetime		-	Waktu email masuk
mail_id	varchar	100	-	Mail address
content	text		-	-
unread	integer	1	-	-

Tabel 3-8 Struktur Tabel Template

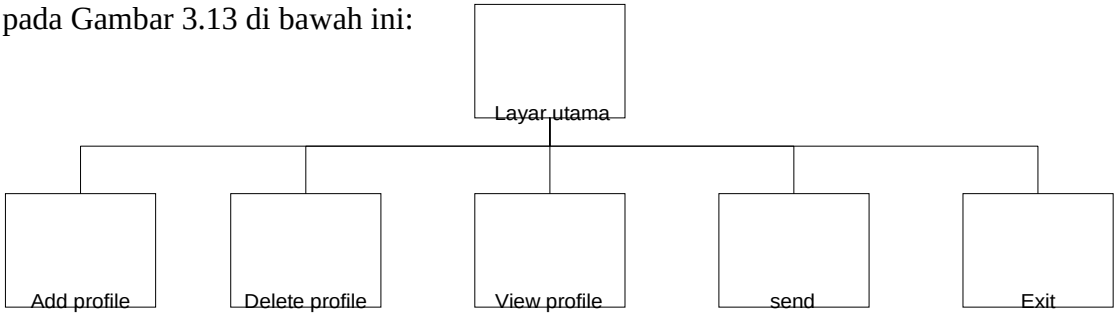
Nama Field	Tipe	Panjang Karakter	kunci	Keterangan
mail_addr	varchar	35	Primary key	Alamat email
in_protocol	varchar	4	-	Jenis incoming protocol
in_server	varchar	35	-	Alamat server incoming protocol
in_port	integer	11	-	Port number incoming protocol
in_ssl	integer	1	-	Keamanan incoming protocol
out_server	varchar	35	-	Alamat server outgoing protocol
out_port	integer	11	-	Port number outgoing protocol
out_ssl	Integer	1	-	Kemanan outgoing protocol
out_auth	integer	1	-	Keamanan autentikasi email

3.2.2 Perancangan Arsitektur Perangkat Lunak

Perancangan arsitektur terdiri dari perancangan arsitektur *data flow*, perancangan antarmuka, dan jaringan semantic pada sistem yang dirancang. Aplikasi yang dibangun merupakan aplikasi berbasis perintah (*commad*) yang disematkan (*embedded*) pada aplikasi yahoo messenger, oleh karena itu form yang terdapat pada perancangan antarmuka merupakan form yang terdapat pada antarmuka (*interface*) yahoo messenger.

3.2.2.1 Perancangan Arsitektur Data Flow

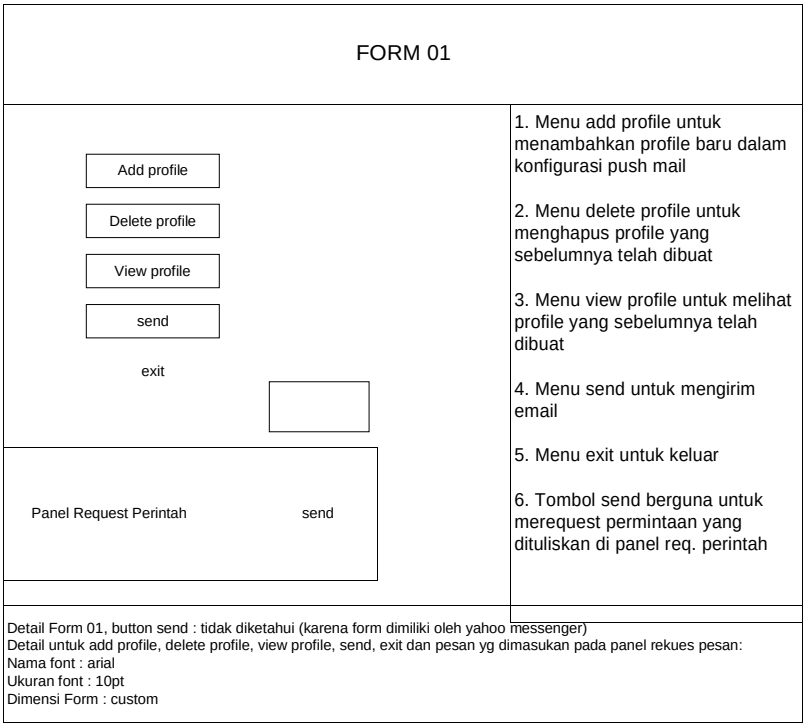
Perancangan arsitektur *data flow* di asumsikan hanya untuk arsitektur perangkat lunak yang dibangun saja, perancangan arsitektur ini tidak termasuk arsitektur *data flow* pada yahoo messenger. Untuk lebih jelasnya dapat dilihat
pada Gambar 3.13 di bawah ini:



Gambar 3.13 Perancangan Arsitektur Data Flow

3.2.2.2 Perancangan Antarmuka

Perancangan atarmuka terdapat pada form antarmuka yahoo messenger, di bawah ini hanya menampilkan perancangan antarmuka pada perangkat lunak yang dibangun saja. F01 antarmuka layar utama. Gambar Gambar 3.14 di bawah ini
adalah perancangan antarmuka layar utama.



Gambar 3.14 Antarmuka Layar Utama

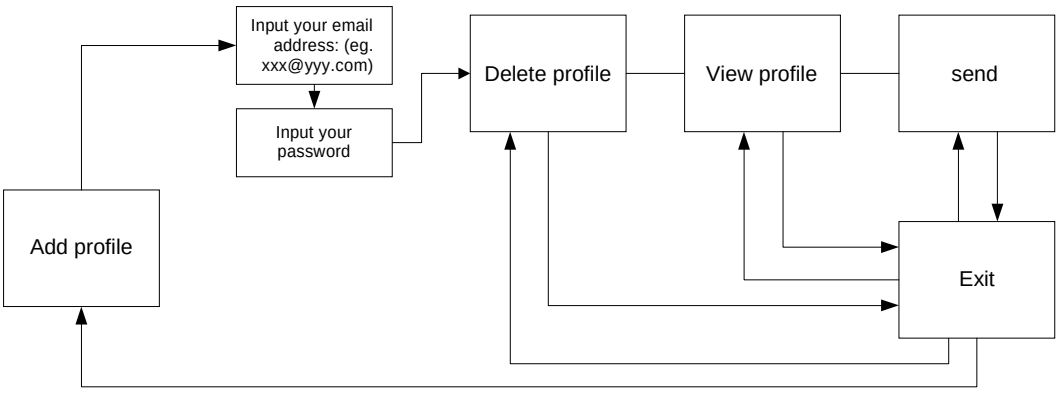
Untuk panel request pesan, dapat dilihat pada tabel perancangan pesan di bawah ini:

Table 3-9 Perancangan Pesan

No Pesan	Isi Pesan	Terdapat Pada Form
01	Add profile	F01
02	Delete profile	F01
03	View profile	F01
04	Send	F01
05	Exit	F01

3.2.2.3 Perancangan Perintah (*Command*)

Dari perancangan antarmuka diatas dapat dilihat bahwa yang dihasilkan dari aplikasi yang dibangun hanya perintah (*command*) yang terdiri dari: add profile, delete profile, view profile, send dan exit saja. Adapun form 01 dan panel request pesan merupakan menu yang terdapat pada tampilan (*interface*) yahoo messenger. Untuk mengetahui lebih jelas tentang perintah yang muncul dapat
dilihat pada gambar perancangan perintah di bawah ini:



Gambar 3.15 Perancangan Perintah (*command*)

3.2.2.4 Perancangan Arsitektur Jaringan

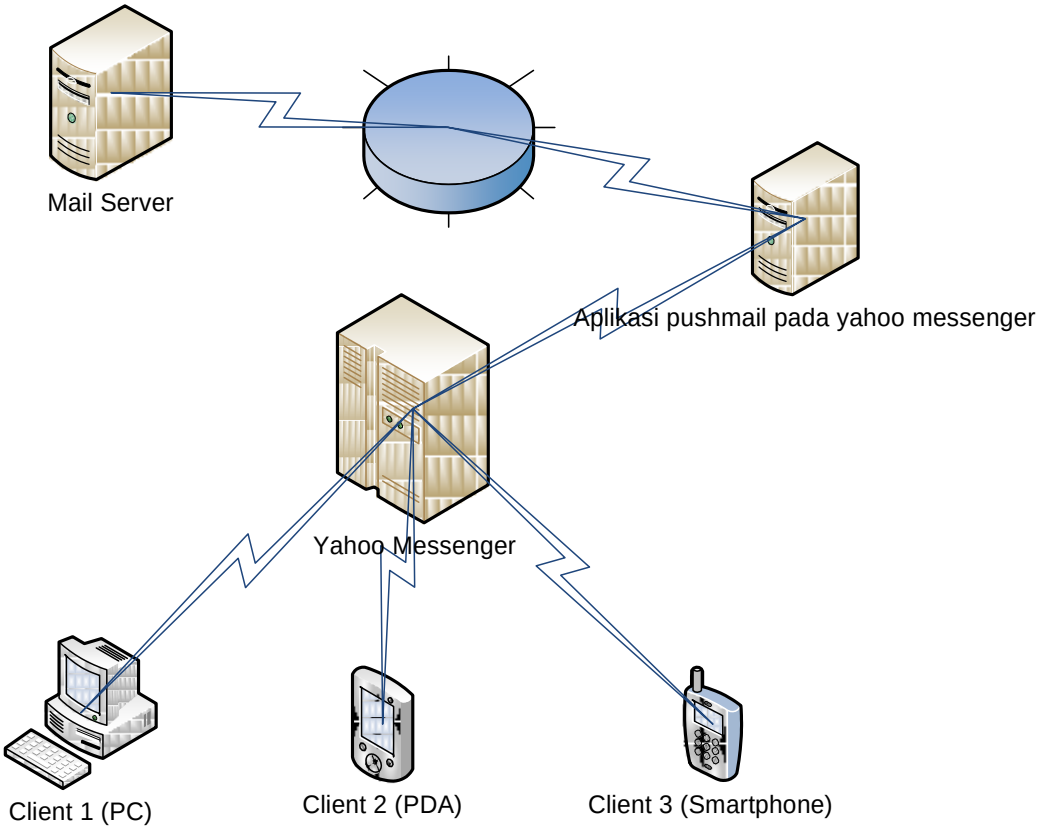
Perancangan arsitektur jaringan dibuat untuk menjelaskan bagaimana aplikasi yang dibangun berjalan. Dalam hal ini perangkat apa saja yang terhubung
di dalamnya dan apa saja yang terkait dengan aplikasi yang dibangun.

Berikut skenario perancangan arsitektur jaringannya:

1. Mail server mengirimkan email ke aplikasi yang dibangun.
2. Aplikasi yang dibangun mengirimkan email ke yahoo messenger (*direct messege*).
3. Yahoo messenger menampilkan email yang masuk ke client 1 dan client 2

serta mengirimkan email yang dikirim oleh client 1 dan 2. client 1 dan client 2 bisa berupa computer atau device lain yang sudah terpasang aplikasi yahoo messenger.

Untuk lebih jelasnya dapat dilihat pada gambar di bawah ini:



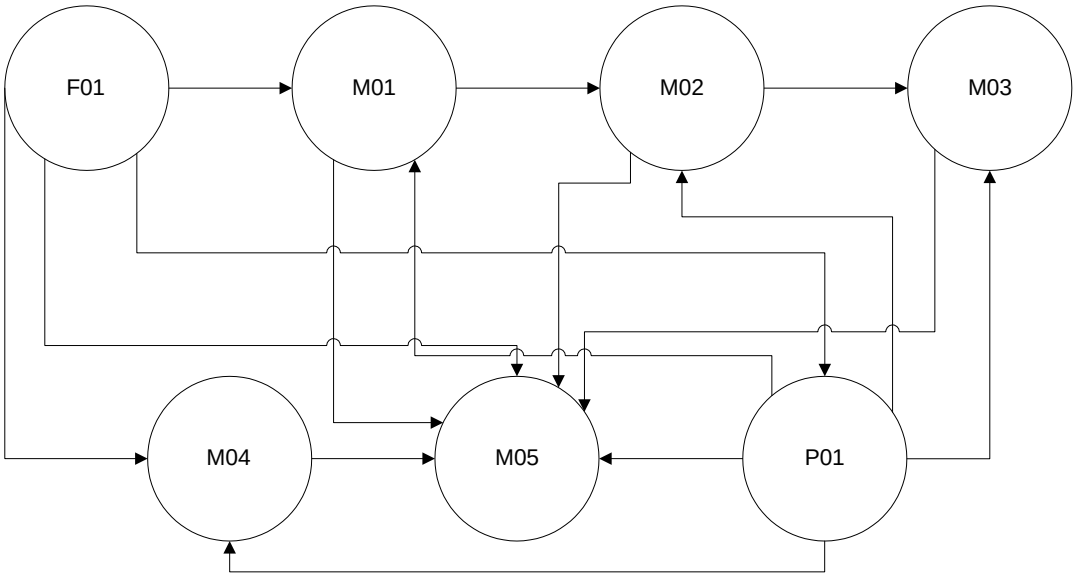
Gambar 3.16 Perancangan Arsitektur Jaringan

3.2.2.5 Jaringan Semantik

Table 3-10 Jaringan Semantik

No	Kode Perancangan	Keterangan
1.	F01	Layar Utama
2.	M01	Menu kesatu (add profile)
3.	M02	Menu kedua (delete profile)
4.	M03	Menu ketiga (view profile)
5.	M04	Menu keempat (send)
6.	M05	Menu kelima (exit)
7.	P01	Panel request pesan

Untuk lebih jelasnya dapat dilihat dalam Gambar 3.17 di bawah ini:



Gambar 3.17 Jaringan Semantik

3.2.3 Perancangan Prosedural

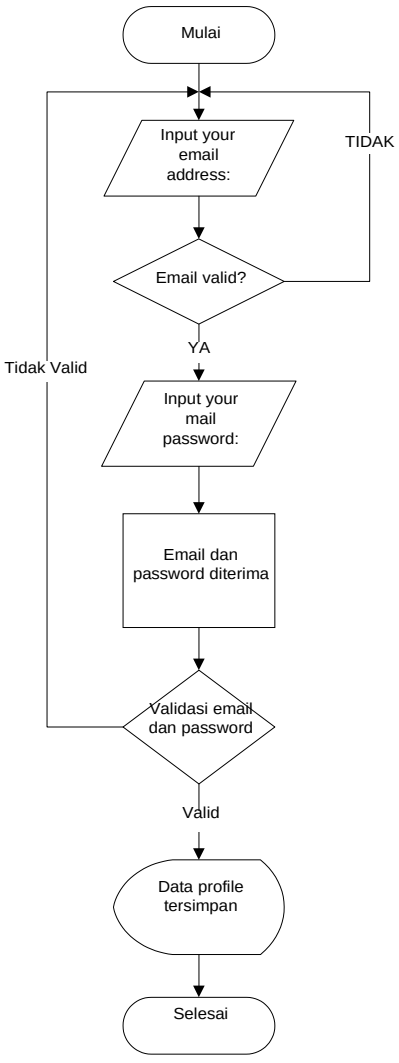
Perancangan prosedural merupakan perancangan yang dilakukan untuk menetapkan detail dari perangkat lunak yang dibangun. Adapun perancangan prosedural untuk perangkat lunak push mail pada yahoo messenger adalah sebagai berikut.

3.2.3.1 Prosedur Add Profile

Prosedur ini dilakukan oleh user untuk menambahkan profile baru kedalam konfigurasi layanan push mail. Skenarionya adalah sebagai berikut:

a. User memasukan data profile yang terdiri dari alamat email dan password.
b. Apabila semua data yang dimasukan valid, data profile akan tersimpan dan apabila tidak proses kembali ke awal.

Untuk lebih jelasnya dapat dilihat dalam flowchart di bawah ini:



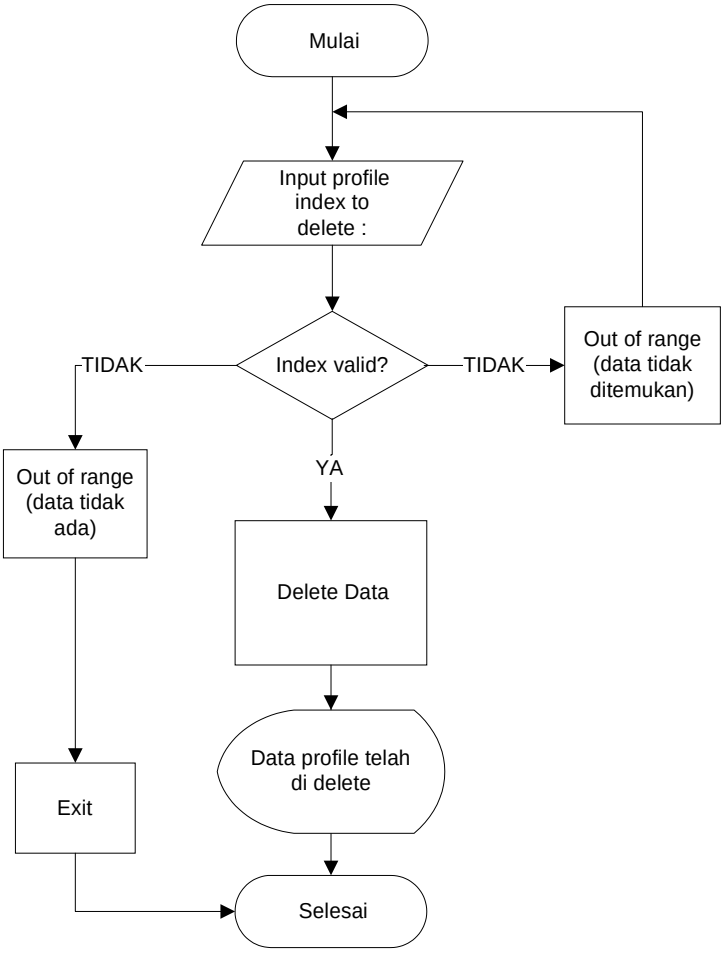
Gambar 3.18 Flowchart Prosedur Add Profile

3.2.3.2 Prosedur Delete Profile

Prosedur ini dilakukan uleh user untuk menghapus profile yang telah dibuat sebelumnya. Skenarionya adalah sebagai berikut:

a. User mencari data yang akan dihapus.
b. Apabila data profile sudah ditemukan, system akan menanyakan terlebih dahulu, yakin dihapus atau tidak.
c. Apabila sudah yakin data akan dihapus.

Untuk lebih jelasnya dapat dilihat pada flowchart di bawah ini:



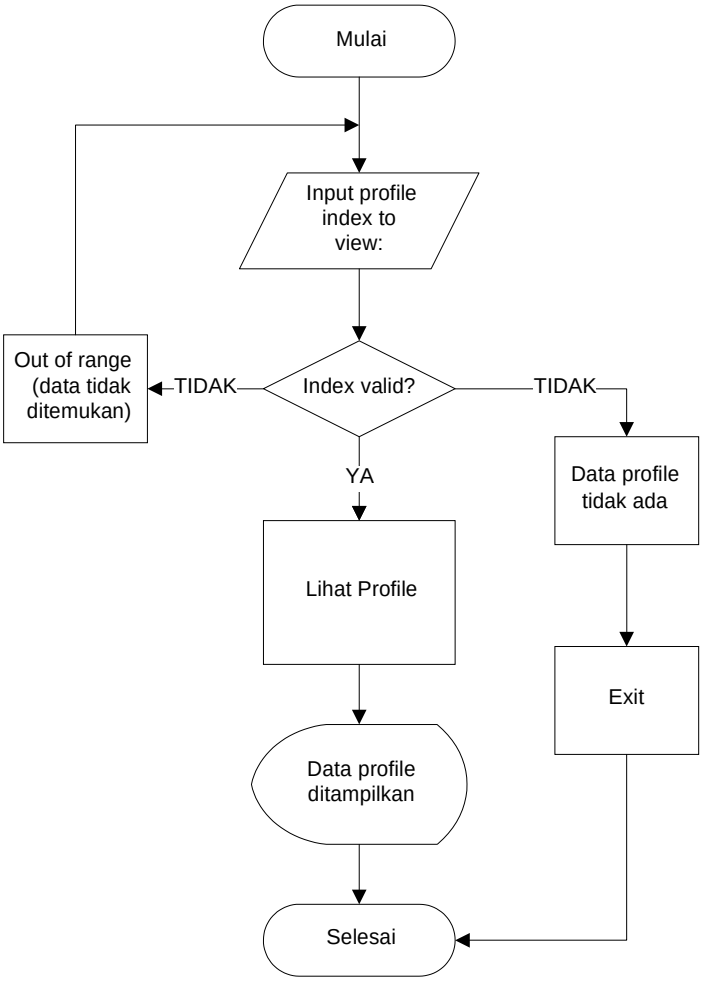
Gambar 3.19 Flowchart Delete Profile

3.2.3.3 Prosedur View Profile

Prosedur ini dilakukan oleh user untuk melihat profile yang telah dibuat sebelumnya. Skenarionya adalah sebagai berikut:

a. User akan mencari profile kemudian akan melihatnya.

Untuk lebih jelasnya dapat dilihat pada Gambar 3.20 berikut ini:



Gambar 3.20 Flowchart Prosedur View Profile

3.2.3.4 Prosedur Send

Prosedur ini dilakukan oleh user untuk mengirim email. Skenarionya adalah sebagai berikut:

a. User menulis pesan email dan memasukan alamat email tujuan.

b. Apabila aplikasi telah dijalankan dan internet sudah jalan email akan masuk ke aplikasi push mail yang telah dibangun.

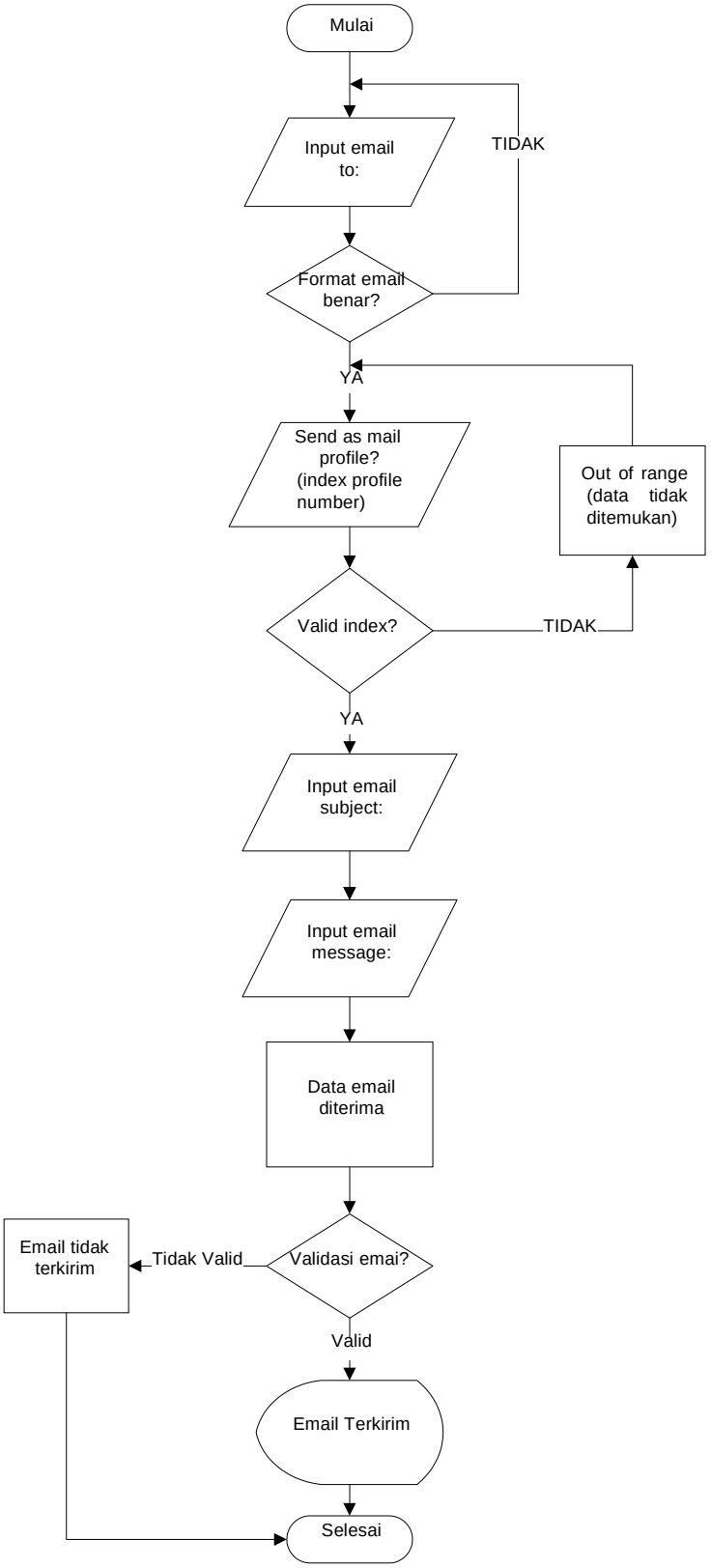
c. Kemudian email akan masuk ke web mail server melalui protokol SMTP

pada port 25.

d. Dari web mail server kemudian diteruskan pada alamat email tujuan melalui protokol POP3 pada port 110.

e. Email berhasil dikirim.

Untuk lebih jelasnya dapat dilihat pada flowchart di bawah ini:



Gambar 3.21 Flowchart Prosedur Send Email