BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisis

Salah satu bentuk komunikasi berbasis teks banyak dilakukan menggunakan aplikasi *messenger*. Proses pengiriman pesan rahasia yang dilakukan menggunakan algoritma *messenger* selama ini tidak melalui proses pengamanan terlebih dahulu. Akibatnya isi dari pesan rahasia yang dikirimkan melalui sebuah aplikasi *messenger* dapat dengan mudah diketahui maknanya oleh orang lain yang tidak diinginkan. Untuk mengatasi masalah tersebut pada penelitian ini akan dibangun sebuah aplikasi *messenger* dengan implementasi kombinasi algoritma ROT 13 degnan algoritma *One Time Pad* (OTP). Dengan menggunakan implementasi algoritma ke dalam sebuah aplikasi *messenger*, pesan rahasia yang akan dikirim dapat disandikan terlebih dahulu sebelum pesan tersebut dikirim. Sehingga pada perangkat penerima pesan yang diterima adalah dalam bentuk *ciphertext*. Aplikasi ini akan dibangun untuk digunakan pada smartphone android sehingga dapat digunakan secara *mobile*. Proses pengiriman pesan juga bersifat *real time* sehingga kegiatan bertukar pesan dapat dilakukan secara cepat.

III.2. Strategi Pemecahan Masalah

Beberapa strategi pemecahan masalah dalam perancangan aplikasi implementasi algoritma ROT 13 dan *One Time Pad* (OTP) dalam aplikasi *real time mesengger* berbasis android ini adalah sebagai berikut:

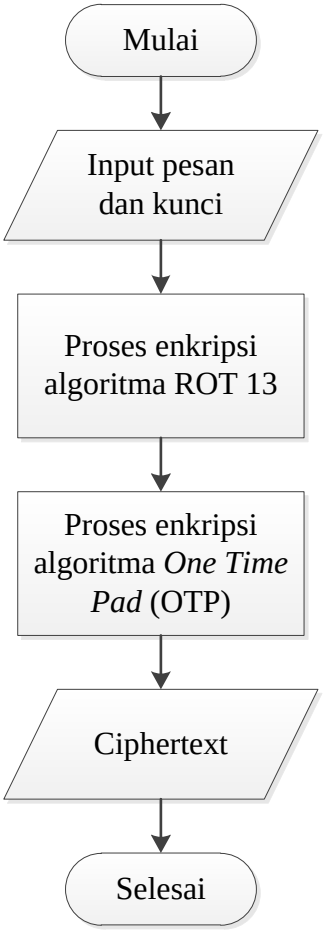
1. Aplikasi ini dibangun menggunakan perangkat lunak Android Studio untuk digunakan pada *smartphone* android.
2. Aplikasi ini dibangun untuk digunakan dalam proses bertukar pesan dalam bentuk teks.
3. Pada aplikasi akan diterapkan algoritma ROT 13 dan algoritma *One Time Pad* (OTP) untuk menyandikan pesan yang bersifat rahasia.

III.3. Penerapan Algoritma

Dalam pembuatan skripsi ini penulis menggunakan dua algoritma kriptografi yaitu algoritma ROT 13 dan algoritma *One Time Pad* (OTP). Untuk proses enkripsi dan dekripsi dari kedua algoritma tersebut dapat dilihat sebagai berikut :

III.3.1. *Flowchart* Enkripsi

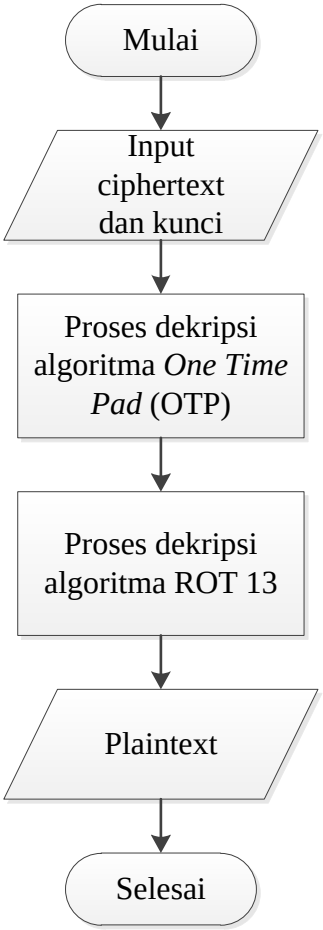
Berikut ini merupakan *flowchart* enkripsi dari kombinasi algoritma ROT 13 dan *One Time Pad* (OTP) dalam mengamankan pesan teks. *Flowchart* enkripsi dapat dilihat pada gambar III.1.



Gambar III.1. *Flowchart* Enkripsi

III.3.2. *Flowchart* Dekripsi

Berikut ini merupakan *flowchart* dekripsi dari kombinasi algoritma ROT 13 dan *One Time Pad* (OTP) dalam mengembalikan *ciphertext* ke bentuk aslinya. *Flowchart* dekripsi dapat dilihat pada gambar III.2.



Gambar III.2. *Flowchart* Dekripsi

III.3.3. Studi Kasus

Dalam penelitian ini digunakan dua buah algoritma yang di gabungkan untuk menghasilkan tingkat keamanan pesan yang lebih baik. Algoritma yang digabungkan adalah algoritma ROT 13 dan algoritma *One Time Pad* (OTP). Misalnya akan melakukan enkripsi terhadap pesan dan kunci sebagai berikut :

Pesan (*plaintext*) : RAHMAN

Kunci : KAMPUS

1. Proses Enkripsi

Proses enkripsi pertama kali dilakukan menggunakan algoritma ROT 13 sebagai berikut :

Pada sistem enkripsi ROT13 sebuah huruf digantikan dengan huruf yang letaknya di atas 13 posisi darinya. Algoritma ini hanya menggunakan basis 26 karakter dari A-Z. penggunaan algortima ROT13 ini dapat dilihat sebagai berikut :

Tabel III.1. Alfabet dan Algoritma ROT-13

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	**Alfabet**
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	**ROT-13**

Berdasarkan tabel di atas proses enkripsi algoritma ROT 13 terhadap pesan "**RAHMAN**" akan menghasilkan *ciphertext* "**ENUZNA**". Hasil tersebut di enkripsi kembali menggunakan algoritma *One Time Pad* (OTP) sebagai berikut :

Pada proses enkripsi dalam studi kasus ini penulis menggunakan anggota himpunan 26 karakter, sehingga persamaan yang digunakan dalam proses enkripsi adalah sebagai berikut :

Ci= (Pi + Ki) *mod* 26 ………………………………………….(1)

Maka perhatikan langkahnya seperti di bawah ini :

1. Konversikan *plaintext* dan kunci kedalam bentuk desimal dalam tabel ASCII.
2. Jumlahkan nilai dari setiap karakter *plaintext* dan kunci dimulai dari yang paling kiri hingga ke kanan.
3. Cari hasil sisa pembagian dari penjumlahan setiap karakter.
4. Konversikan kembali nilai desimal kedalam bentuk karakter.

Untuk lebih jelasnya dapat dilihat sebagai berikut :

Melakukan konversi setiap karakter *plaintext* dan juga kunci. Pada penelitian ini akan digunakan himpunan 26 karakter sehingga didapatkan nilai masing-masing karakter adalah :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Plaintext : E = 4, N = 13, U = 20, Z = 25, N = 13, A = 0

Kunci : K = 10, A = 0, M = 12, P = 15, U = 20, S = 18

Setelah mendapatkan nilai konversi, selanjutnya dilakukan perhitungan sehingga di dapat hasil sebagai berikut :

C1 = (E + K) mod 26
= (4 + 10) mod 26
= (14) mod 26
= 14
= O

C2 = (N + A) mod 26
= (13 + 0) mod 26
= (13) mod 26
= 13
= N

C3 = (U + M) mod 26
= (20 + 12) mod 26
= (32) mod 26
= 6
= G

C4 = (Z + P) mod 26
= (25+15) mod 26
= (40) mod 26
= 14
= O

C5 = (N + U) mod 26
= (13 + 20) mod 26
= (33) mod 26
= 7
= H

C6 = (A + S) mod 26
= (0 + 18) mod 26
= (18) mod 26
= 18
= S

Ciphertext yang dihasilkan dari proses enkripsi menggunakan algoritma ROT 13 dan algoritma *One Time Pad* (OTP) "**ONGOHS**".

2. Proses Dekripsi

Berikut ini adalah proses dekripsi yang dilakukan menggunakan algoritma ROT 13 dan algoritma *One Time Pad* (OTP) menggunakan *ciphertext* dan kunci sebagai berikut :

Ciphertext : **ONGOHS**

Kunci : **KAMPUS**

Dalam proses dekripsi pertama kali dilakukan menggunakan algoritma One Time Pad (OTP). Untuk mendekripsikan karakter-karakter *ciphertext* menjadi karakter-karakter *plaintext* dapat dilakukan menggunakan persamaan berikut :

Pi= (Ci - Ki) *mod* 26 ……(2)

Untuk proses perhitungannya adalah

P1 = (O - K) mod 26
= (14 - 10) mod 26
= (4) mod 26
= 4
= E

P2 = (N - A) mod 26
= (13 - 0) mod 26
= (13) mod 26
= 13
= N

P3 = (G - M) mod 26
= (6 - 12) mod 26
= (-6) mod 26
= 20
= U

P4 = (O - P) mod 26
= (14-15) mod 26
= (-1) mod 26
= 25
= Z

P5 = (H - U) mod 26
= (7 - 20) mod 26
= (-13) mod 26
= 13
= N

P6 = (S - S) mod 26
= (18 - 18) mod 26
= (0) mod 26
= 0
= Z

Hasil dekripsi menggunakan algoritma One Time Pad (OTP) adalah "**ENUZNA**". Setelah proses dekripsi menggunakan algoritma OTP selanjutnya dilakukan dekripsi menggunakan algoritma ROT 13 menggunakan tabel berikut :

Tabel III.1. Alfabet dan Algoritma ROT-13

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M

Alfabet

ROT-13

Dengan menggunakan tabel tersebut dalam proses dekripsi sehingga dari *ciphertext* "**ENUZNA**" akan dihasilkan *plaintext* "**RAHMAN**".

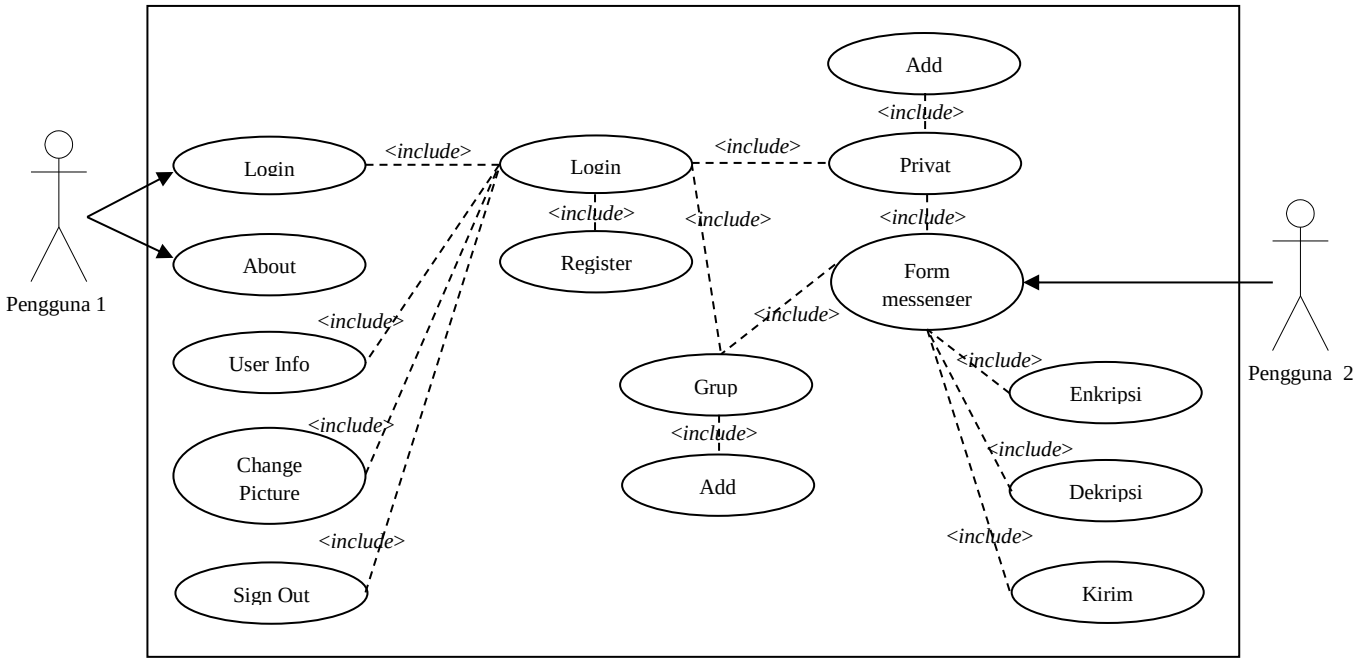
III.4. Desain Sistem

Perancangan aplikasi implementasi algoritma ROT 13 dan *One Time Pad* (OTP) dalam aplikasi *real time mesengger* berbasis android dirancang dengan menggunakan perangkat lunak Android Studio. Perancangan sistem yang

dirancang terdiri dari *use case*, *activity diagram*, *sequence diagram* serta desain dan penjelasan dari sistem yang dirancang. Berikut adalah perancangannya :

III.4.1. *Use Case Diagram*

Use case mendiskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi yang ada didalam sistem informasi tersebut. Berikut adalah *use case diagram* dari sistem yang dirancang :



Gambar III.3. *Use Case Diagram*

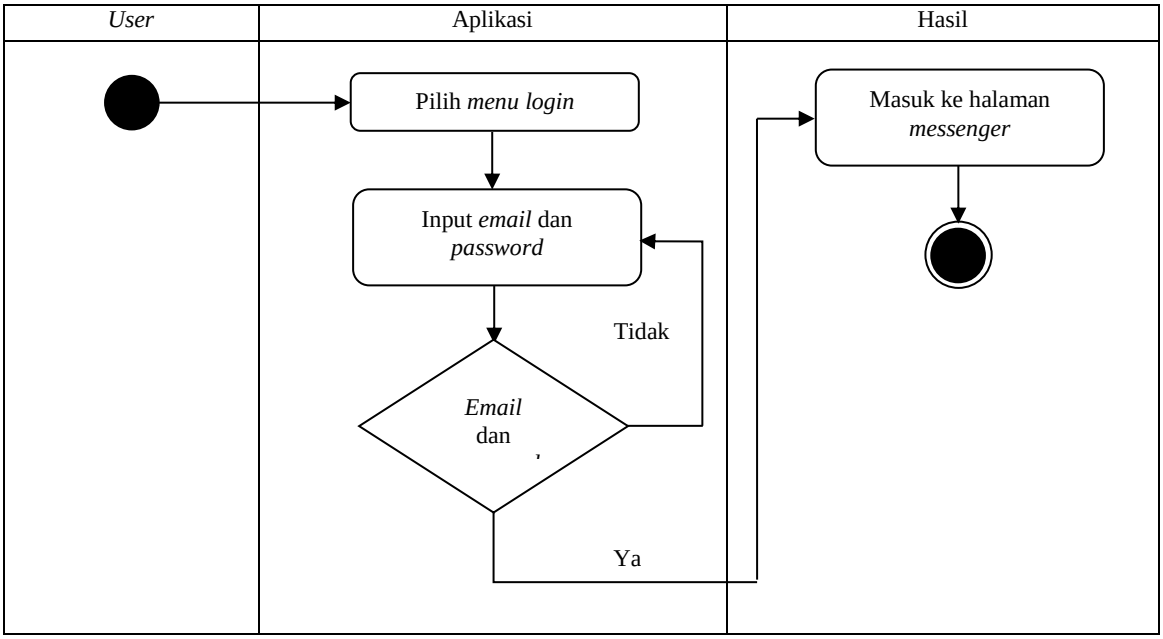
III.4.2. *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang

mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* yang terdapat pada aplikasi yaitu sebagai berikut :

III.4.2.1. *Activity Diagram Login*

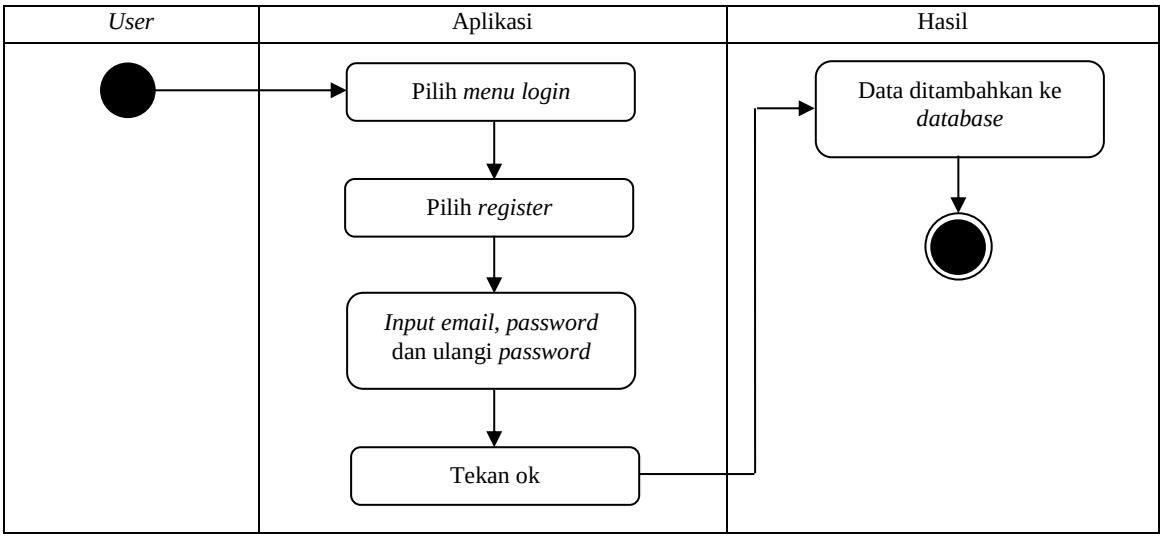
Activity diagram login menggambarkan alir aktifitas untuk melakukan proses *login* untuk dapat memulai bertukar pesan. Proses *login* dapat dilihat pada gambar III.4.



Gambar III.4. *Activity Diagram Login*

III.4.2.2. *Activity Diagram Register*

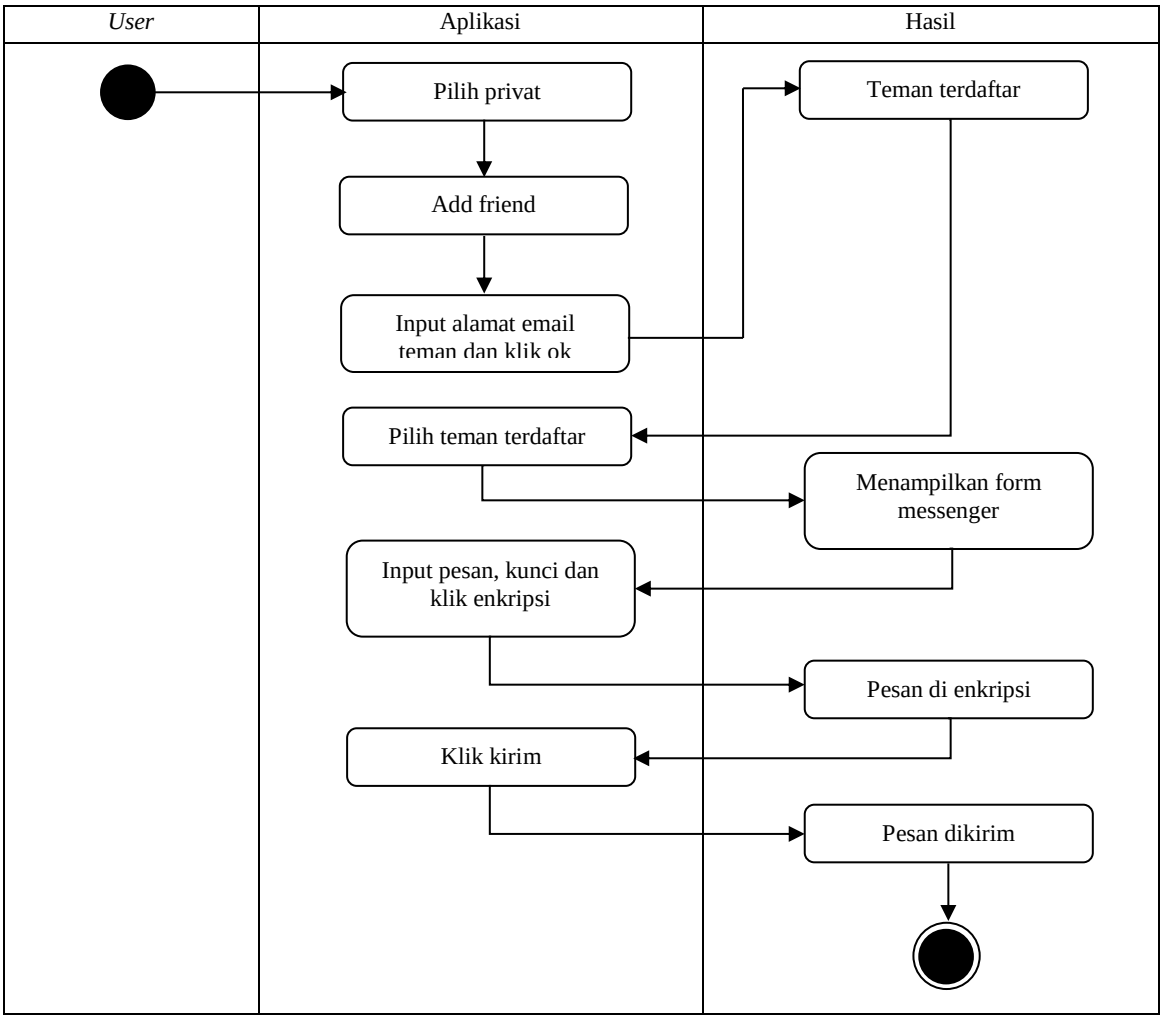
Activity diagram register menggambarkan alir aktifitas untuk melakukan *register* pada aplikasi. Proses *register* dapat dilihat pada gambar III.5.



Gambar III.5. *Activity Diagram Register*

III.4.4.3. *Activity Diagram Privat*

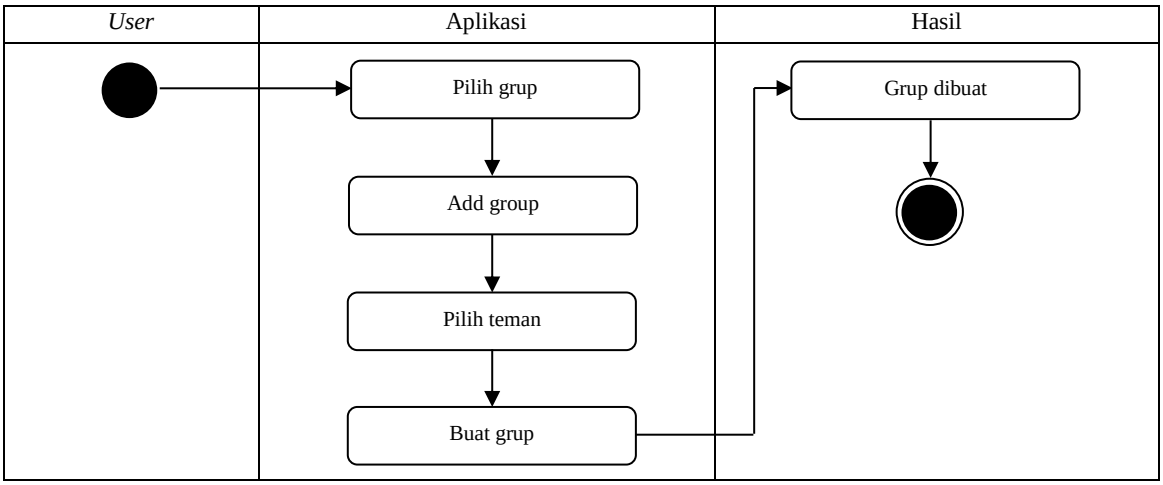
Activity diagram privat menggambarkan alir aktifitas untuk melakukan pengiriman pesan antar *user*. *Activity diagram privat* dapat dilihat pada gambar III.6.



Gambar III.6. ***Activity Diagram Privat***

III.4.2.4. *Activity Diagram* Grup

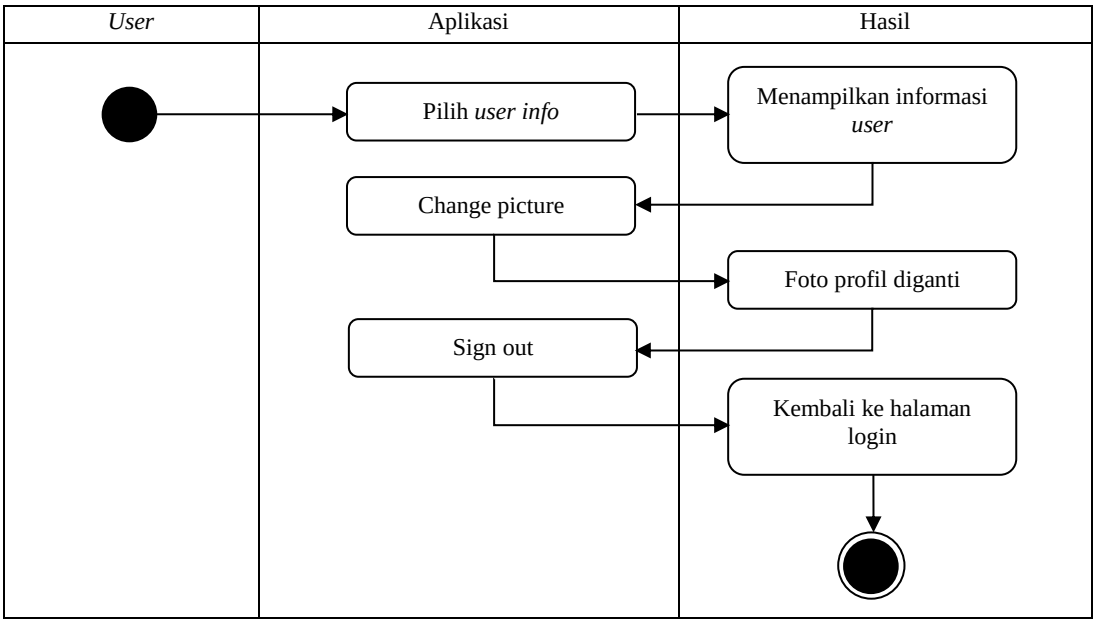
Activity diagram grup menggambarkan alir aktifitas dalam melakukan proses mengirim pesan secara grup yang berisi lebih dari 3 *user* dalam 1 grup. *Activity diagram* grup dapat dilihat pada gambar III.7.



Gambar III.7. *Activity Diagram Grup*

III.4.2.5. *Activity Diagram User Info*

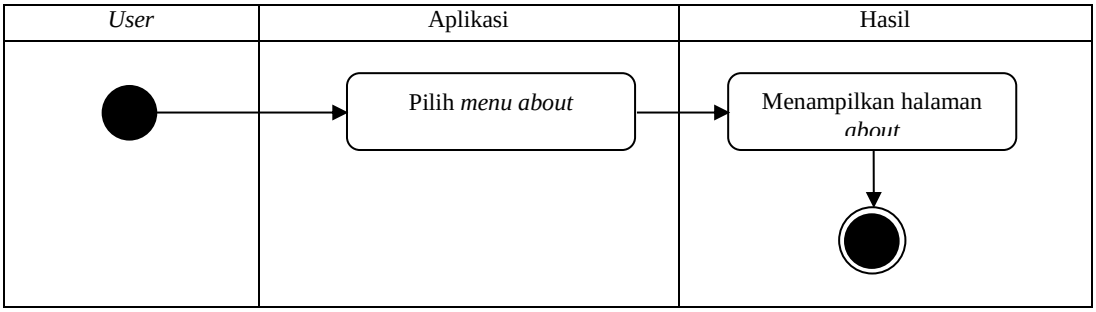
Activity diagram user info merupakan *activity diagram* pada saat memilih menu *user info*. Dalam activity ini pengguna dapat merubah *password* dan juga melakukan *sign out* dari aplikasi. *Activity diagram user info* ditunjukkan pada gambar III.8.



Gambar III.8. *Activity Diagram User Info*

III.4.2.6. *Activity Diagram About*

Activity diagram about merupakan *activity diagram* pada saat memilih menu *about*. Dalam activity ini akan ditampilkan informasi dari pembuat aplikasi. *Activity diagram about* ditunjukkan pada gambar III.9.



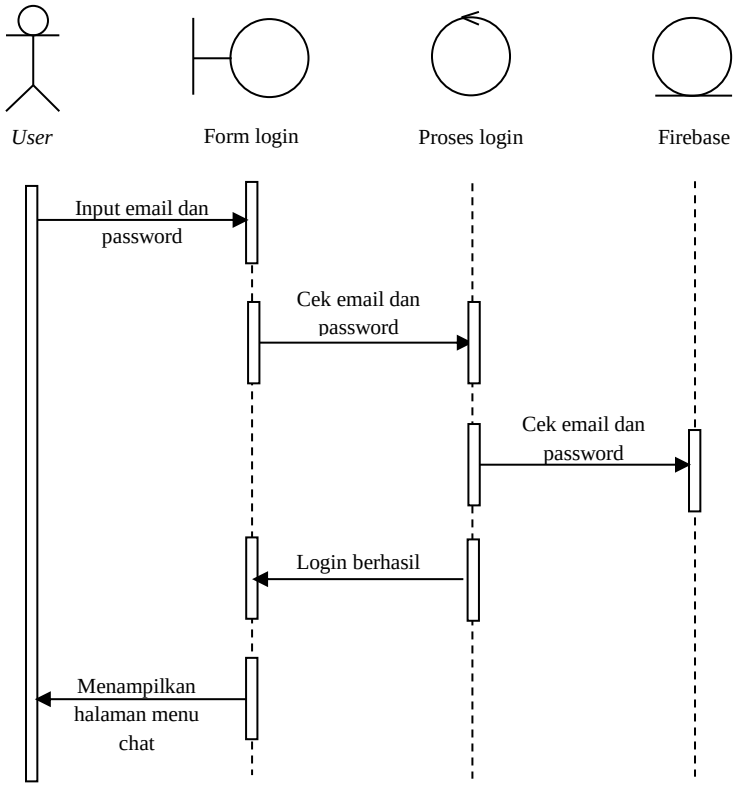
Gambar III.9. *Activity Diagram About*

III.4.3. *Sequence Diagram*

Sequence diagram pada aplikasi yang akan dibuat yaitu : *Sequence diagram login, register, privat, grup, user info* dan *about*.

III.4.3.1. *Sequence Diagram Login*

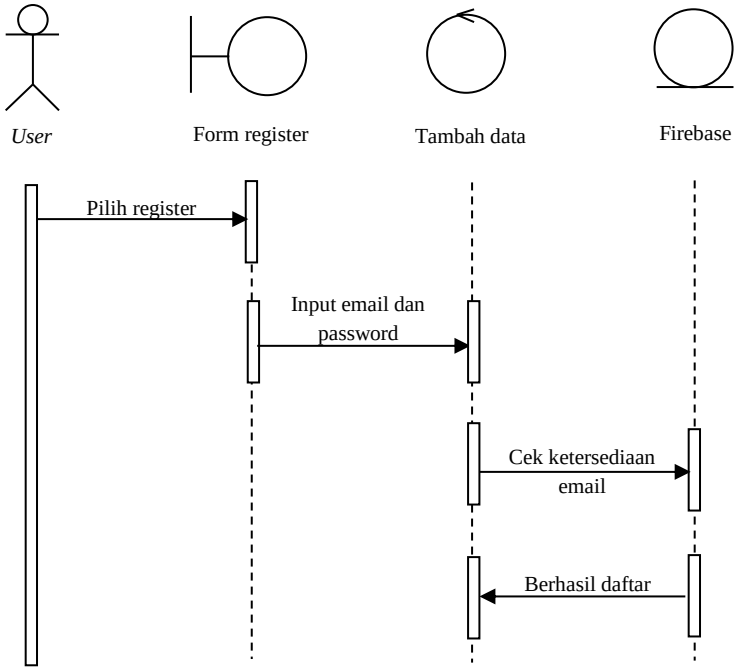
Sequence diagram login menggambarkan interaksi yang terjadi pada saat melakukan proses *login*. *Sequence diagram login* ditunjukkan pada gambar III.10.



Gambar III.10. *Sequence Diagram Login*

III.4.3.2. *Sequence Diagram Register*

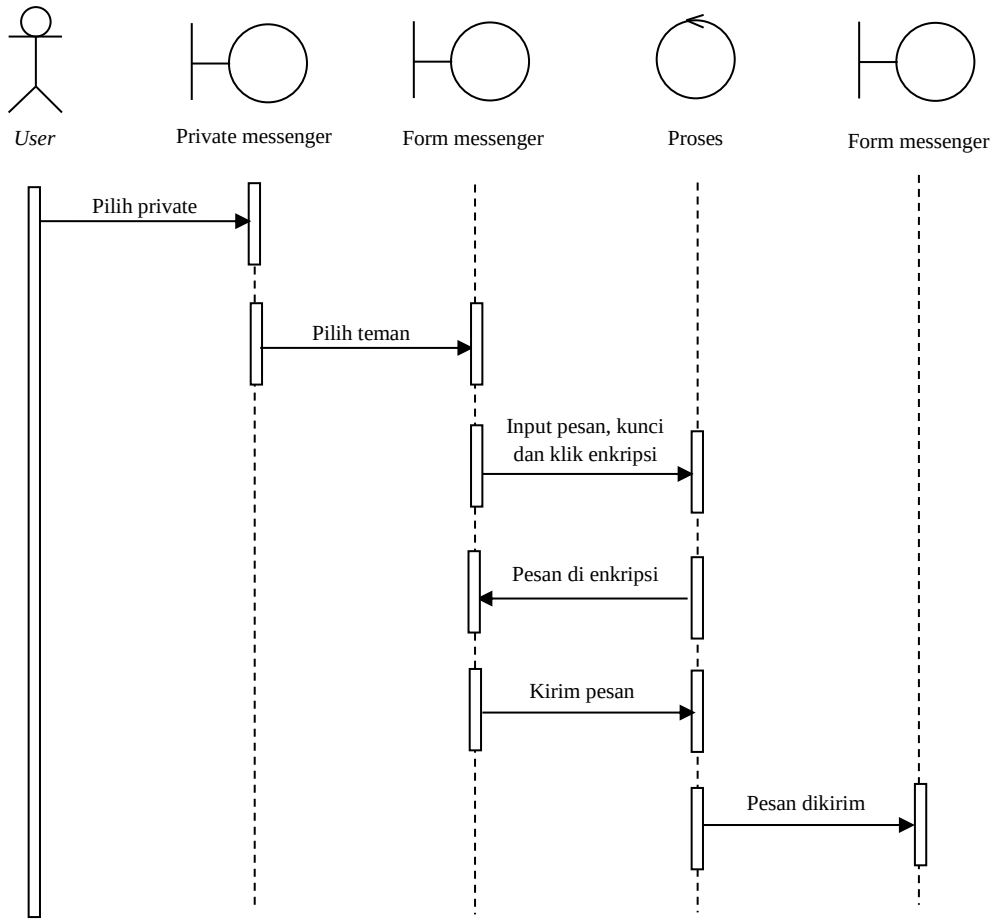
Sequence diagram register menggambarkan interaksi yang terjadi pada saat melakukan proses *register*. *Sequence diagram register* ditunjukkan pada gambar III.11.



Gambar III.11. *Sequence Diagram Register*

III.4.3.3. *Sequence Diagram Privat*

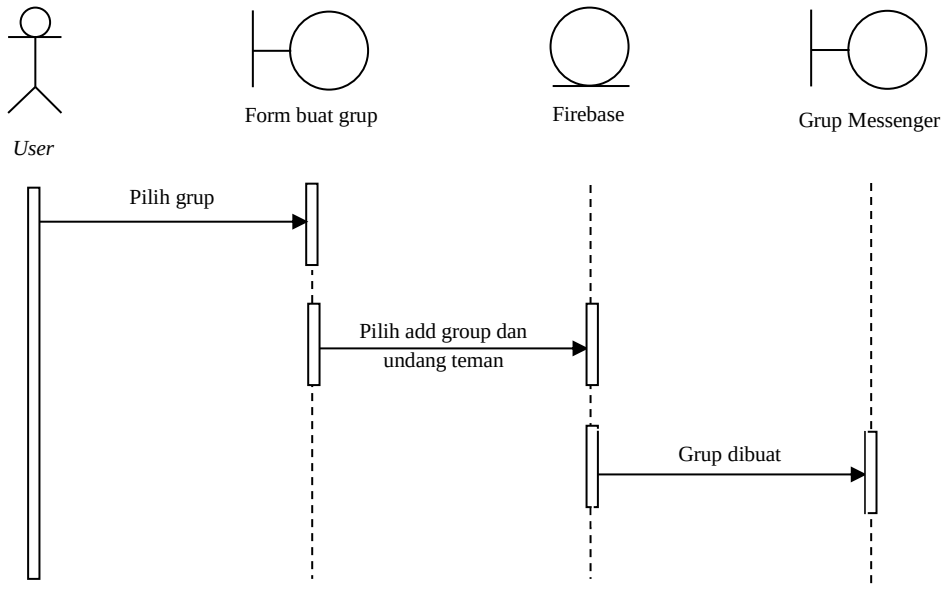
Sequence diagram privat menggambarkan interaksi yang terjadi pada saat melakukan proses *privat*. *Sequence diagram privat* ditunjukkan pada gambar III.12.



Gambar III.12. *Sequence Diagram Privat*

III.4.3.4. *Sequence Diagram Grup*

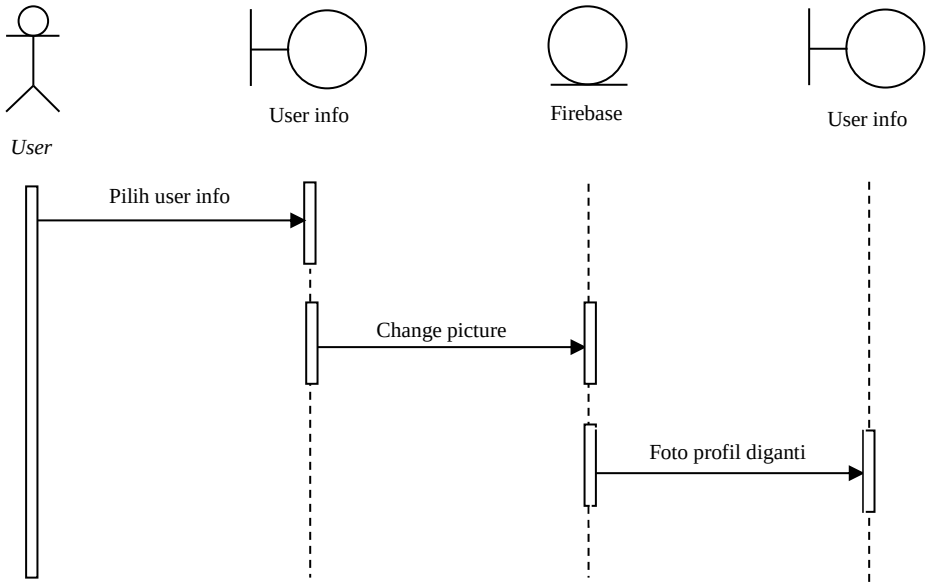
Sequence diagram grup menggambarkan interaksi yang terjadi pada saat melakukan proses *grup*. *Sequence diagram grup* ditunjukkan pada gambar III.13.



Gambar III.13. *Sequence Diagram Grup*

III.4.3.5. *Sequence Diagram User Info*

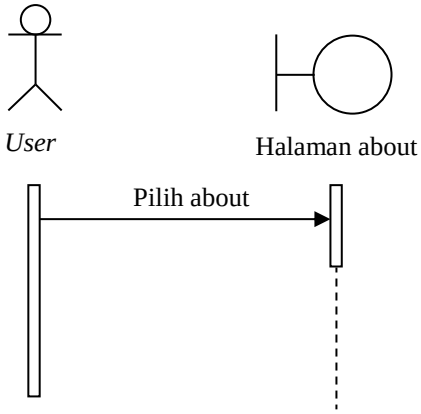
Sequence diagram user info menampilkan informasi dari *user* yang melakukan *login*. *Sequence diagram user info* ditunjukkan pada gambar III.14.



Gambar III.14. *Sequence Diagram User Info*

III.4.3.6. *Sequence Diagram About*

Sequence diagram about menampilkan informasi dari pembuat aplikasi. *Sequence diagram about* ditunjukkan pada gambar III.15.

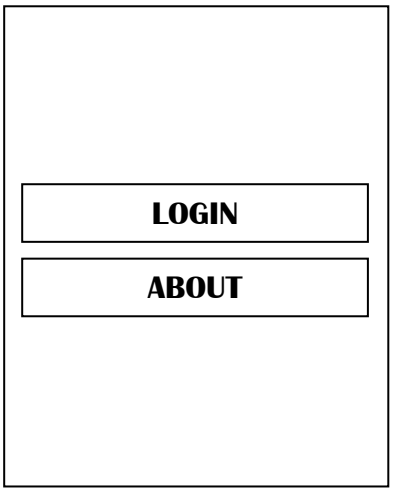


Gambar III.15. *Sequence Diagram About*

III.5. Desain *User Interface*

Antarmuka peamakai (*user interface*) adalah tampilan program yang dapat dilihat atau dipersepsikan oleh pengguna dan perintah-perintah atau mekanisme yang digunakan pemakai untuk mengendalikan operasi dan memasukkan data. Berikut ini merupakan perancangan antarmuka dari perancangan aplikasi implementasi algoritma ROT 13 dan *One Time Pad* (OTP) dalam aplikasi *real time mesengger* berbasis android, yaitu :

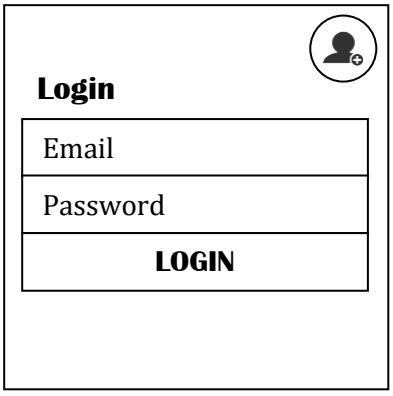
1. Desain Halaman Utama



Gambar III.16. Desain Halaman Utama

Merupakan tampilan rancangan halaman utama saat aplikasi pertama kali dijalankan. Pada halaman ini terdapat tombol login untuk menampilkan halaman login dan tombol about untuk menampilkan halaman tentang aplikasi.

2. Desain Halaman *Login*



Gambar III.17. Desain Halaman *Login*

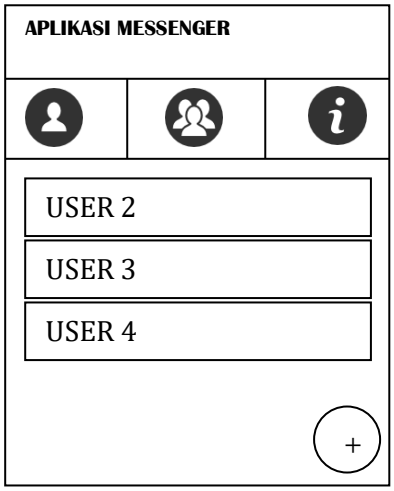
Merupakan tampilan rancangan halaman *login*. Pada halaman ini pengguna harus menginputkan *email* dan *password* untuk dapat masuk kedalam halaman *messenger*.

3. Desain Halaman *Register*

Gambar III.18. Desain Halaman *Register*

Pada halaman ini pengguna dapat mendaftarkan data baru untuk dapat *login*. Setelah melakukan *register* pengguna dapat melakukan *login* ke halaman *messenger*.

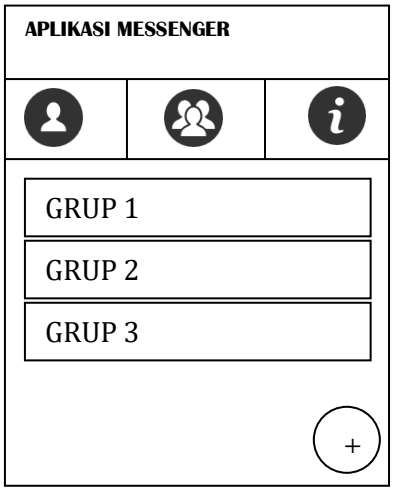
4. Desain Halaman *Privat*



Gambar III.19. Desain Halaman *Privat*

Halaman ini adalah halaman untuk melakukan pengiriman pesan secara *privat* antar *user*. Jika belum memiliki teman, pengguna dapat menambahkan teman dengan klik tombol tambah dan menginputkan *email* teman yang ingin ditambahkan.

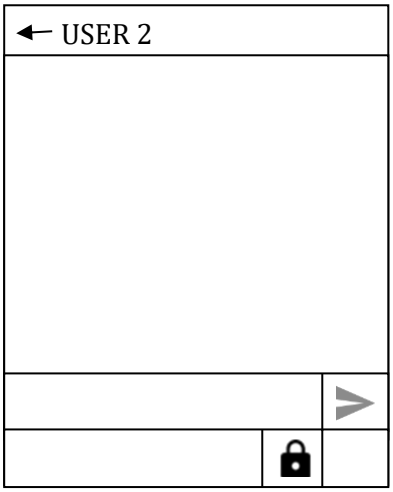
5. Desain Halaman Grup



Gambar III.20. Desain Halaman Grup

Halaman ini adalah halaman untuk melakukan pengiriman pesan secara bersamaan dalam satu ruang *messenger*. Untuk melakukan grup *messenger* pengguna harus mengundang minimal 3 teman untuk bergabung di dalam grup *messenger*.

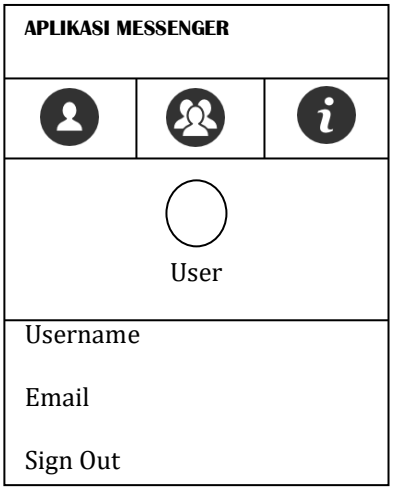
6. Desain Halaman *Form Messenger*



Gambar III.21. Desain Halaman *Form Messenger*

Halaman ini merupakan halaman *messenger* dimana pada halaman ini pengguna dapat saling bertukar pesan. Pada *form messenger* pengguna juga dapat melakukan enkripsi pesan sebelum pesan dikirim. Dan dapat juga melakukan dekripsi pesan yang di terima.

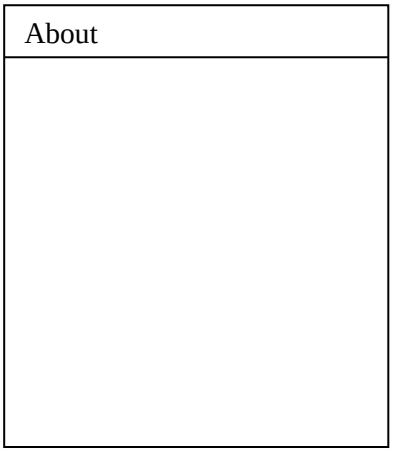
7. Desain Halaman Info *User*



Gambar III.22. Desain Halaman Info *User*

Halaman ini menampilkan data pengguna yang telah *login*. *Username* pengguna. Pada halaman ini pengguna dapat mengganti foto profil dan juga terdapat tombol untuk kembali ke halaman login.

8. Desain Halaman *About*



Gambar III.23. Desain Halaman *About*

Pada form ini akan ditampilkan tentang judul penelitian yang sedang dilaksanakan dan juga informasi dari peneliti.