BAB III

ANALISIS DAN PERANCANGAN

III.1. Analisis Masalah

Analisa sistem pada yang berjalan bertujuan untuk mengidentifikasi serta melakukan evaluasi terhadap Aplikasi Penyandian Pesan chat Menggunakan Metode Merkle Hellman. Adapun masalah yang terdapat pada sistem sebelumnya adalah kurang berkembang sebuah pesan chat dengan memanfaatkan sistem keamanan data yang dapat menjaga kerahasiaan dan keamanan pengiriman data pada pesan chat dan kurang berkembag sebuah pesan chat dengan menggunakan Algoritma Merkle Hellman

Strategi dalam melakukan pemecahan masalah yang sedang dianalisa oleh penulis mengenai perancangan Aplikasi Penyandian Pesan chat Menggunakan Metode Merkle Hellman adalah sebagai berikut :

1. Merancang sebuah aplikasi chat dengan memanfaatkan sistem keamanan data yang dapat menjaga kerahasiaan dan keamanan pengiriman text pada aplikasi chat.
2. Merancang dan membangun sebuah aplikasi chat dengan menggunakan merkle hellman

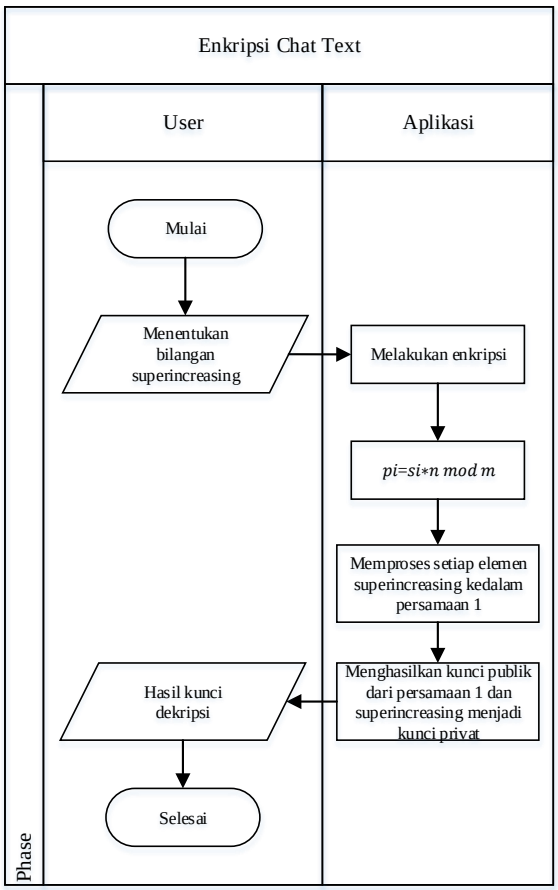
III.1.1. Analisis *Input*

Analisa input digunakan untuk melakukan analisis data pada aplikasi. Data yang digunakan pada analisa input adalah text chat sebelum dilakukan enkripsi. Misalnya text yang dimasukkan oleh user pada aplikasi chatting adalah Hafizah, data text ini kemudian akan diamankan menggunakan algoritma.

III.1.2. Analisis Proses

1. Analisis Proses Enkripsi Text

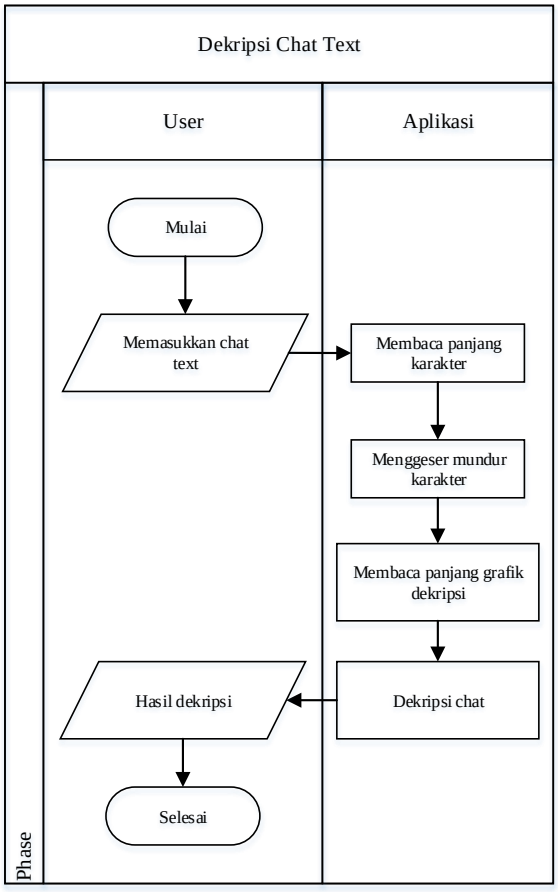
Analisa proses digunakan untuk mengentahui prosedur yang digunakan dalam melakukan pengolahan enkripsi text.



Gambar III.1. FOD Proses Enkripsi Chat Text

2. Analisis Proses Dekripsi Text

Analisa proses digunakan untuk mengentahui prosedur yang digunakan dalam melakukan pengolahan dekripsi text.



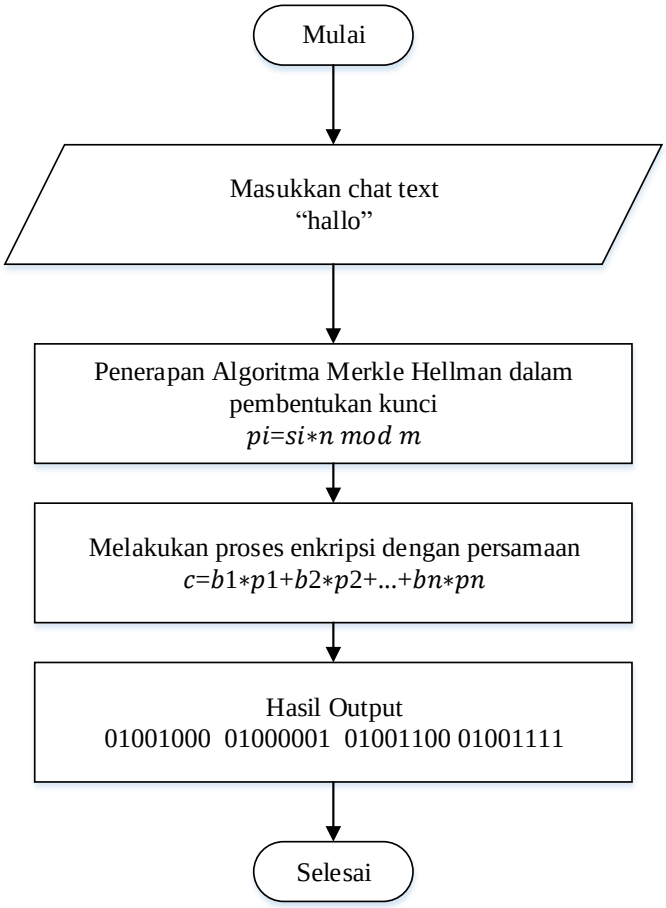
Gambar III.2. FOD Proses Dekripsi Chat Text

III.1.3. Analisis *Output*

Analisa output digunakan untuk melihat hasil akhir dari pengolahan data input dan analisa output. Analisa output yang dihasilkan oleh pengolahan data input adalah data dekripsi dari text yang akan dikirmkan, contohnya adalah kata "Hallo" setelah di enkripsikan maka yang keluar adalah "01001000 01000001 01001100 01001111" yang dibentuk sebagai kata kunci dekripsi.

III.1.4. Flowchart Algoritma *Merkle Hellman*

Flowchart atau Bagan alir adalah bagan (chart) yang menunjukkan alir (flow) di dalam program atau prosedur sistem secara logika. Bagan alir (flowchart) digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.



Gambar III.3. Flowchart Proses Algoritma *Merkle Hellman*

II.2. Penerapan Metode Algoritma *Merkle Hellman*

Berikut adalah beberapa mekanisme dari algoritma ini antara lain:

a. Menentukan deretan superincreasing di mana setiap elemen di dalam deretan harus lebih besar daripada jumlah elemen sebelumnya. Deretan superincreasing ini digunakan sebagai kunci privat.
b. Memilih bilangan prima *m* dengan ketentuan bilangan tersebut harus lebih besar daripada elemen terakhir deretan superincreasing

c. Memilih bilangan *n*, di mana bilangan tersebut relatif prima terhadap bilangan *m*.

d. Membangkitkan kunci publik dengan persamaan 1:

$$pi=si*n \ mod \ m \ (1)$$

Variabel i pada pada persamaan (1) merupakan indeks dari deretan superincreasing. Jadi, setiap elemen di dalam bilangan superincreasing diproses pada persamaan (1) dan menghasilkan kunci publik. Kunci publik tersebut memiliki panjang elemen yang sama dengan deretan superincreasing. Gambaran umum tentang pembangkitan kunci algoritma knapsack seperti pada Gambar III.1.

e. Melakukan proses enkripsi dengan persamaan (2):

$$c=b1*p1+b2*p2+...+bn*pn \ (2)$$

Selain persamaan (2), adapun *pseudocode* untuk proses enkripsi adalah sebagai berikut:

```
INPUT : publickey[]
OUTPUT : chipertext
FOR(i=0;í<text.lenght; i++) DO
ASCII=plaintext.charAt(i)
biner= Int.toBinary(ASCII)
c = 0
FOR(j=0;j<biner.lenght;j++) DO c=c+(biner.charAt[j].publickey[j])
ENDFOR
chipertext=chipertext+c+" "
ENDFOR
```

Variabel b pada persamaan (2) merupakan bentuk biner dari plainteks yang akan dienkripsi. Setiap indeks biner tersebut dikalikan dengan elemen kunci publik dengan indeks yang sama. Hasil tersebut ditambahkan dengan indeks selanjutnya. Proses tersebut dilakukan berulang sampai pada indeks terakhir.

f. Melakukan proses dekripsi dengan persamaan (3):

$$c* n{-}1 \ mod \ m \ (3)$$

Selain dari persamaan (3) ada pula pseudocode untuk proses dekripsi algoritma knapsack seperti berikut :

```
INPUT: m, n, chiprtxt
OUTPUT: plaintext
m_inv = n.modInvers(m)
String temp = ""
FOR(i=0;ciphrtxt.lenght;i++) DO
if((chiprtxt.charAt(i)+"")=" ") do
p = m_inv.mod(m)
ASCII = Integer.parseInt
plaintext = plaintext+ASCII
temp = ""
ELSE
temp=temp+chipertext.charAt(i)
ENDIF
ENDFOR
```

Nilai n-1 pada persamaan (3) merupakan nilai invers dari bilangan prima *m* yang sudah dibangkitkan. Penghitungan nilai invers ini menggunakan *extended euclidean*. Setelah dilakukannya proses dekripsi, proses selanjutnya adalah mengkonversi hasil tersebut ke dalam bentuk biner. Untuk mencari nilai binernya, hasil dekripsi dikurangi dengan elemen superincreasing yang besarnya mendekati hasil dekripsi tersebut. Proses tersebut berlanjut sampai hasil pengurangan menjadi 0. Diberikan nilai 1 pada indeks superincreasing apabila terpilih menjadi operasi pengurangan dan diberikan nilai 0 jika tidak terpilih. Hasil dari serangkaian proses tersebut menghasilkan nilai biner dan dikonversi kembali dalam bentuk karakter

Proses dekripsi pada kombinasi algoritma ini dilakukan dua kali. Proses yang pertama menggunakan persamaan (5). Hasil dari persamaan ini didekripsi lagi menggunakan persamaan (3).

Tabel 1. Hasil Enkripsi Himpunan a-z

K	P	C	K	P	C	K	P	C
a	97	143	j	106	336	s	115	276
b	98	259	k	107	61	t	116	346
c	99	329	l	108	269	u	117	25
d	100	8	m	109	63	v	118	118
e	101	101	n	110	202	w	119	119
f	102	102	o	111	19	x	120	126
g	103	103	p	112	153	y	121	213
h	104	196	q	113	343	z	122	283
i	105	266	r	114	160			

User B mengenkripsi dengan kunci publik yang dibagikan oleh User A yang menghasilkan nilai chiperteks 266 346. Disisi lain user C berhasil mendapatkan nilai chiperteks yang dienkripsi oleh user B yaitu 266 346. Dari hasil tersebut user C mencocokkan nilai chiperteks yang berkoresponden terhadap karakter. Dalam pencocokkannya, user C mendapatkan nilai yang berkoresponden yaitu C=266=105, C=346=116. Dari hasil tersebut user C mendapatkan plainteks P = 105 116 yang kemudian diubah ke karakter menjadi "it". Dari skenario diatas dapat disimpulkan bahwa user C hanya mendapatkan kunci publik dan chiperteks untuk mengetahui karakter asli dari chiperteks dan tidak perlu menggunakan kunci privat.

III.3. Perancangan

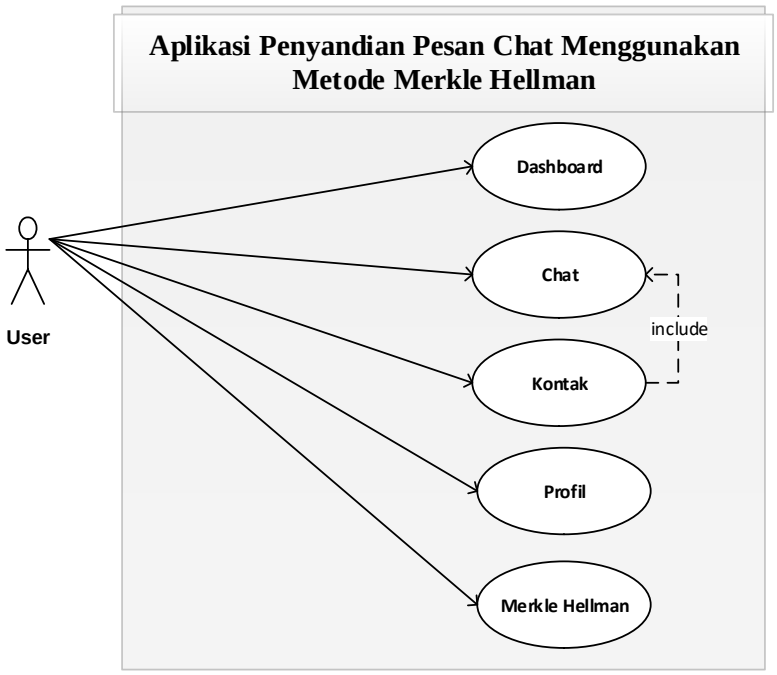
Desain sistem pada penelitian ini dibagi menjadi dua desain, yaitu desain sistem secara global untuk penggambaran model sistem secara garis besar dan desain sistem secara detail untuk membantu dalam pembuatan sistem.

III.3.1. Desain Sistem Secara Global

Desain sistem secara global menggunakan bahasa pemodelan UML yang terdiri dari *Usecase Diagram, Acitvity Diagram* dan *Sequence Diagram*.

III.3.1.1. *Usecase* Diagram

Dalam penyusunan suatu program diperlukan suatu model data yang berbentuk diagram yang dapat menjelaskan suatu alur proses sistem yang akan di bangun. Dalam penulisan skripsi ini penulis menggunakan metode UML yang dalam metode itu penulis menerapkan diagram *Use Case*. Maka digambarlah suatu bentuk diagram *Use Case* yang dapat dilihat pada gambar dibawah ini



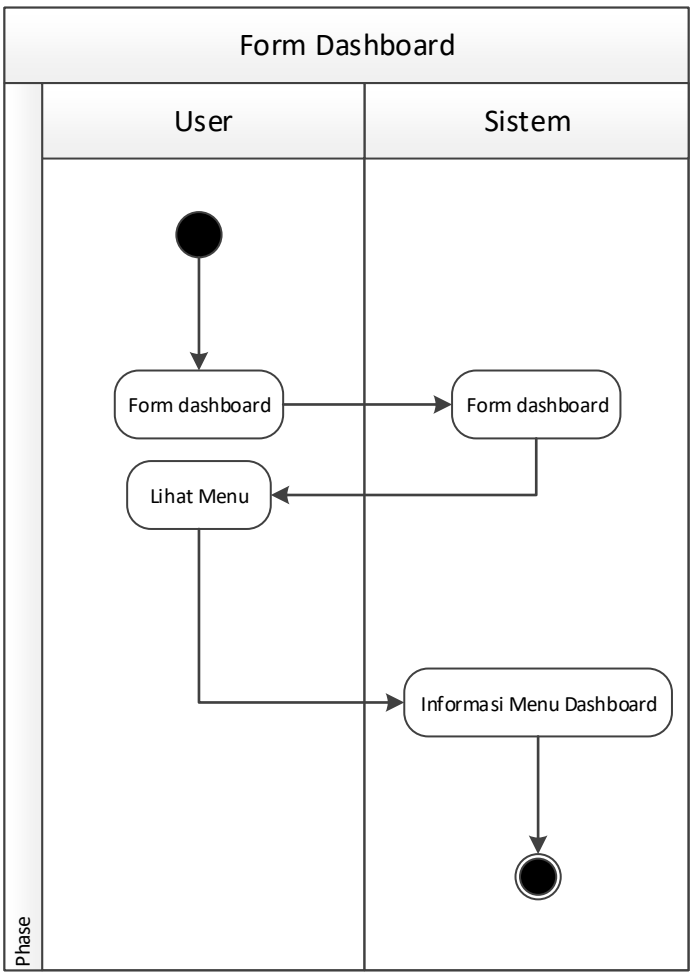
Gambar III.4. *Use Case* Diagram Aplikasi Penyandian Pesan chat Menggunakan Metode Merkle Hellman

III.3.1.2. *Activity* Diagram

Bisnis proses yang telah digambarkan pada *use case diagram* dijabarkan dengan *activity diagram* :

1. *Activity Diagram* Dashboard

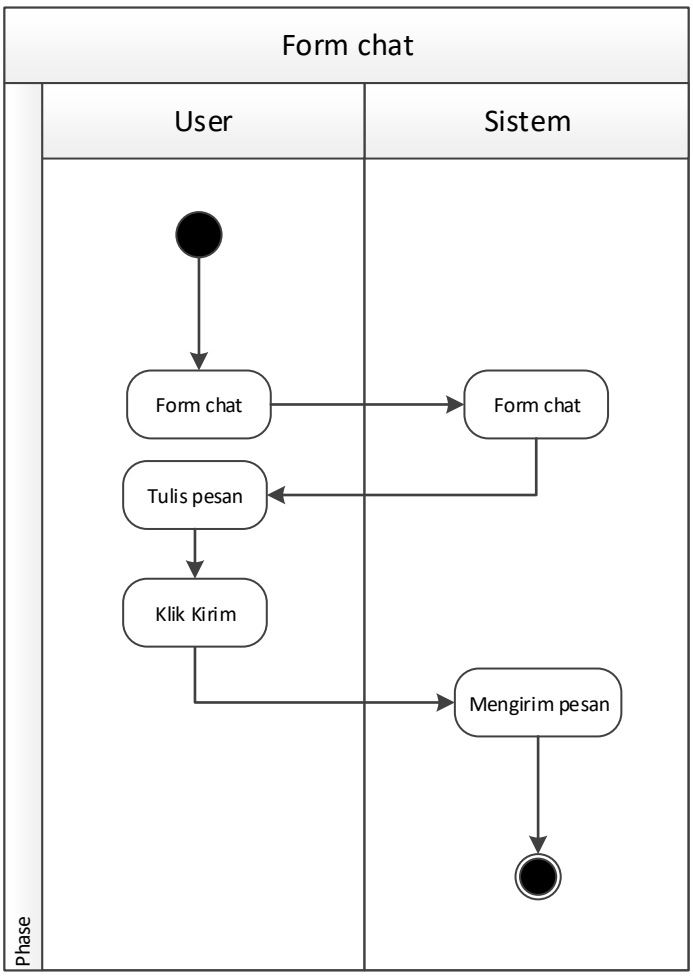
   Aktivitas yang dilakukan oleh user pada form dashboard dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.5 berikut :



Gambar III.5. *Activity Diagram* Form Dashboard

2. *Activity Diagram* Chat

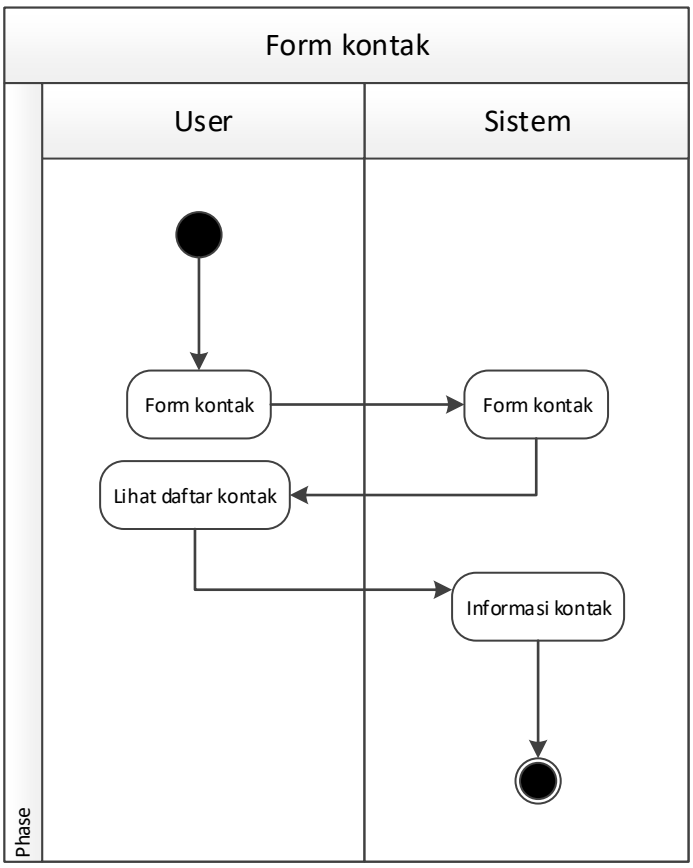
Aktivitas yang dilakukan oleh user pada form chat dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.6 berikut :



Gambar III.6. *Activity Diagram* Form Chat

3. *Activity Diagram* Kontak

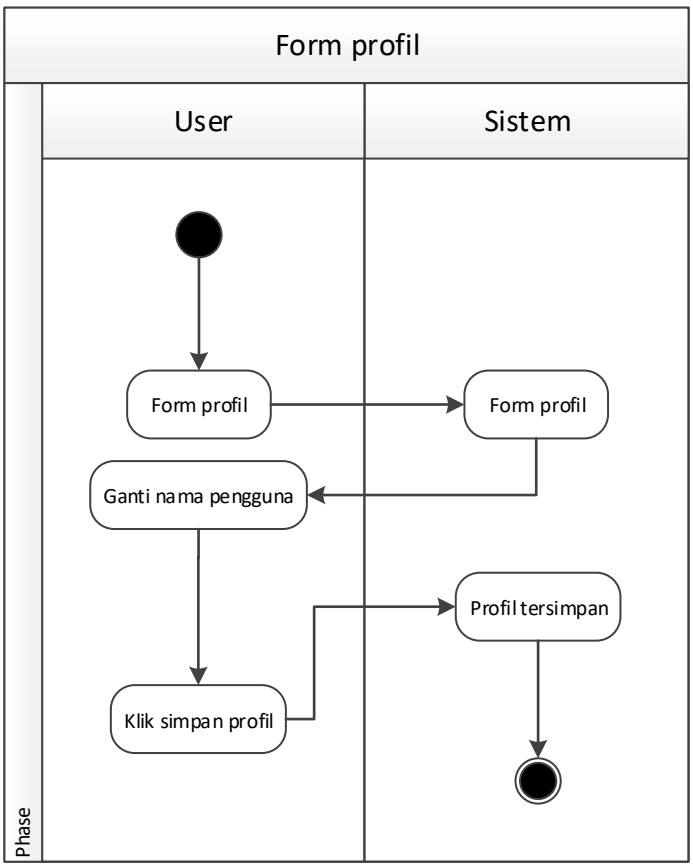
Aktivitas yang dilakukan oleh user pada form kontak dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.7 berikut :



Gambar III.7. *Activity Diagram* Form Kontak

4. *Activity Diagram* Profil

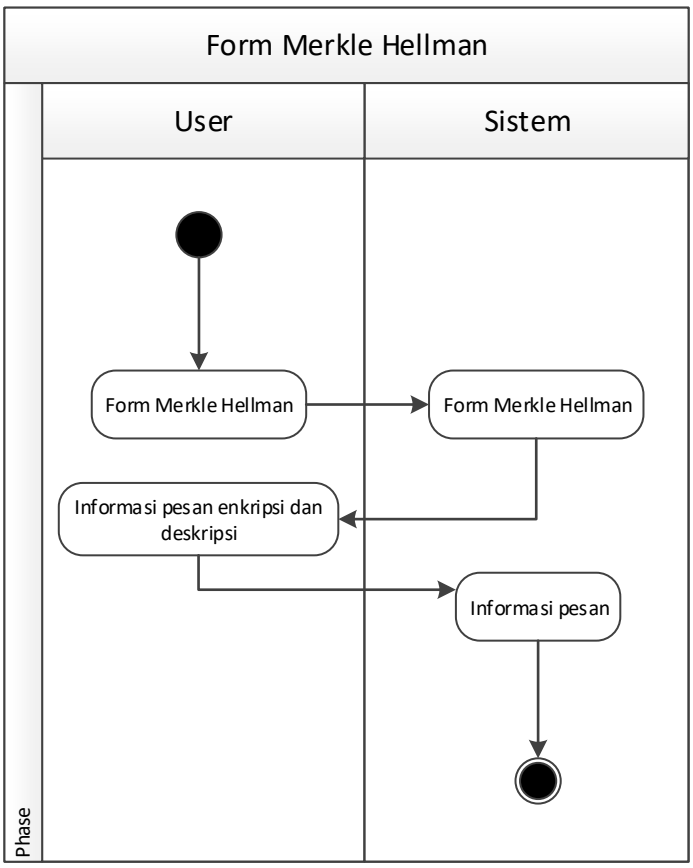
Aktivitas yang dilakukan oleh user pada form profil dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.8 berikut :



Gambar III.8. *Activity Diagram* Form Profil

5. *Activity Diagram Merkle Hellman*

Aktivitas yang dilakukan oleh user pada form *Merkle Hellman* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.9 berikut :



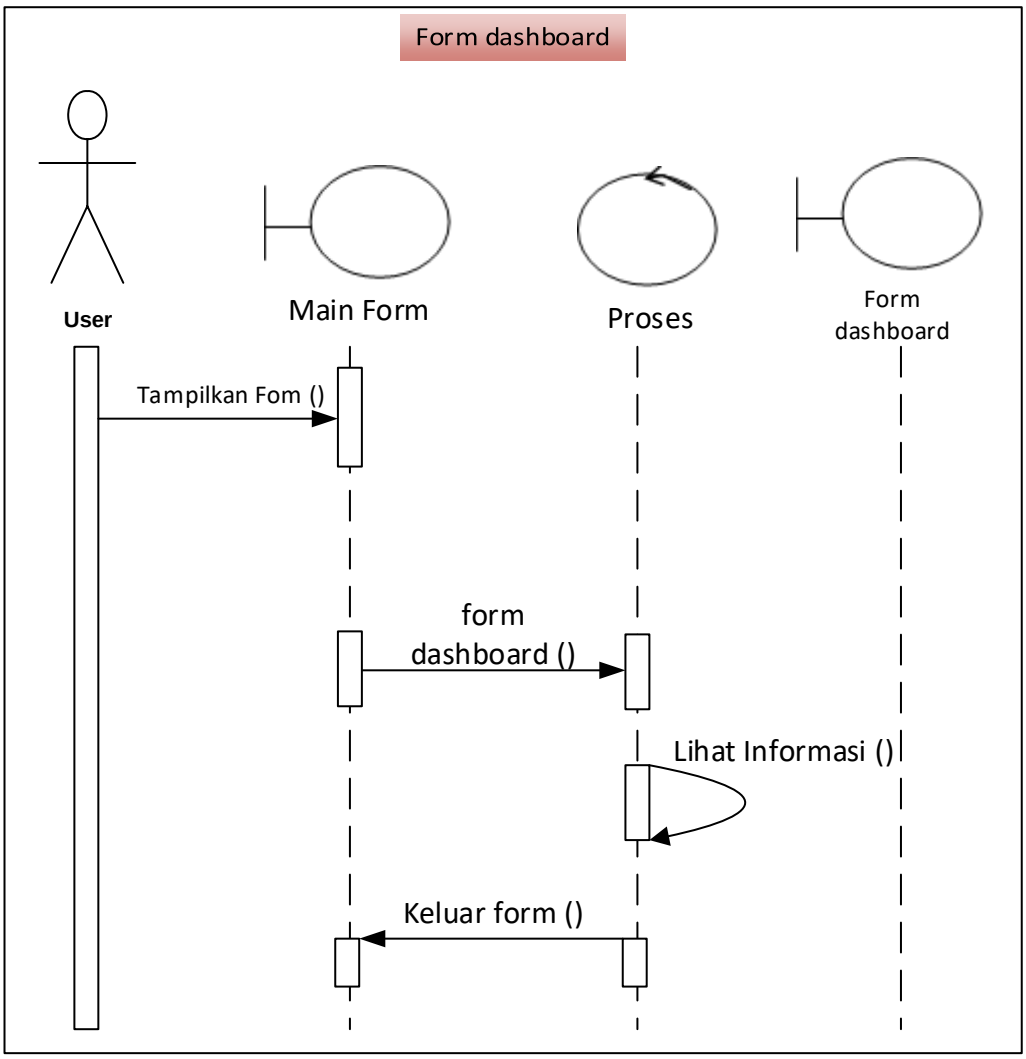
Gambar III.9. *Activity Diagram* Form *Merkle Hellman*

III.3.1.3. *Sequence* Diagram

Sequence Diagram (diagram urutan) adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, *display*, dan sebagainya berupa pesan/*message*.

1. *Sequence Diagram* Dashboard

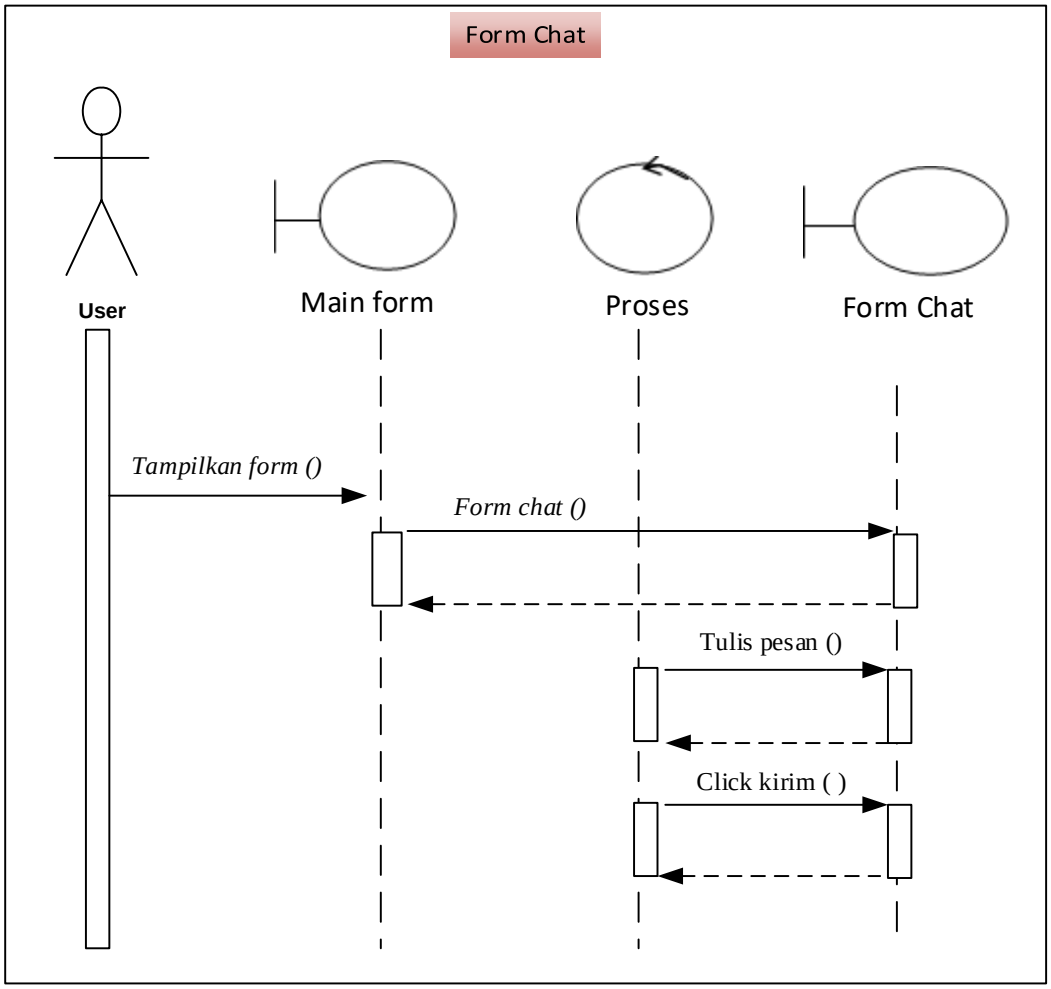
   Serangkaian kegiatan sistem yang dilakukan oleh user pada form dashboard dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.10 berikut :



Gambar III.10. *Sequence Diagram* Form Dashboard

2. *Sequence Diagram* Chat

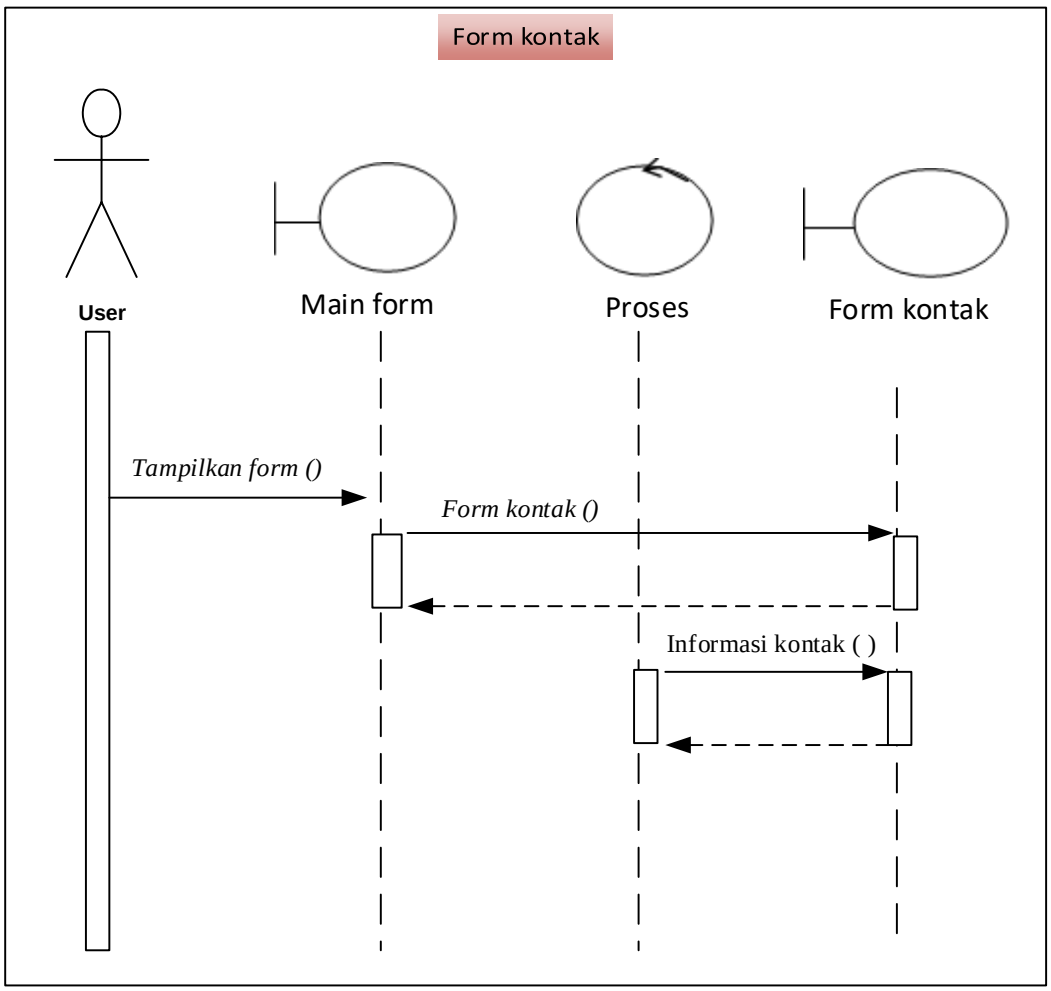
Serangkaian kegiatan sistem yang dilakukan oleh user pada form berkas pelajaran dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.11 berikut :



Gambar III.11. *Sequence Diagram* Form Chat

3. *Sequence Diagram* Kontak

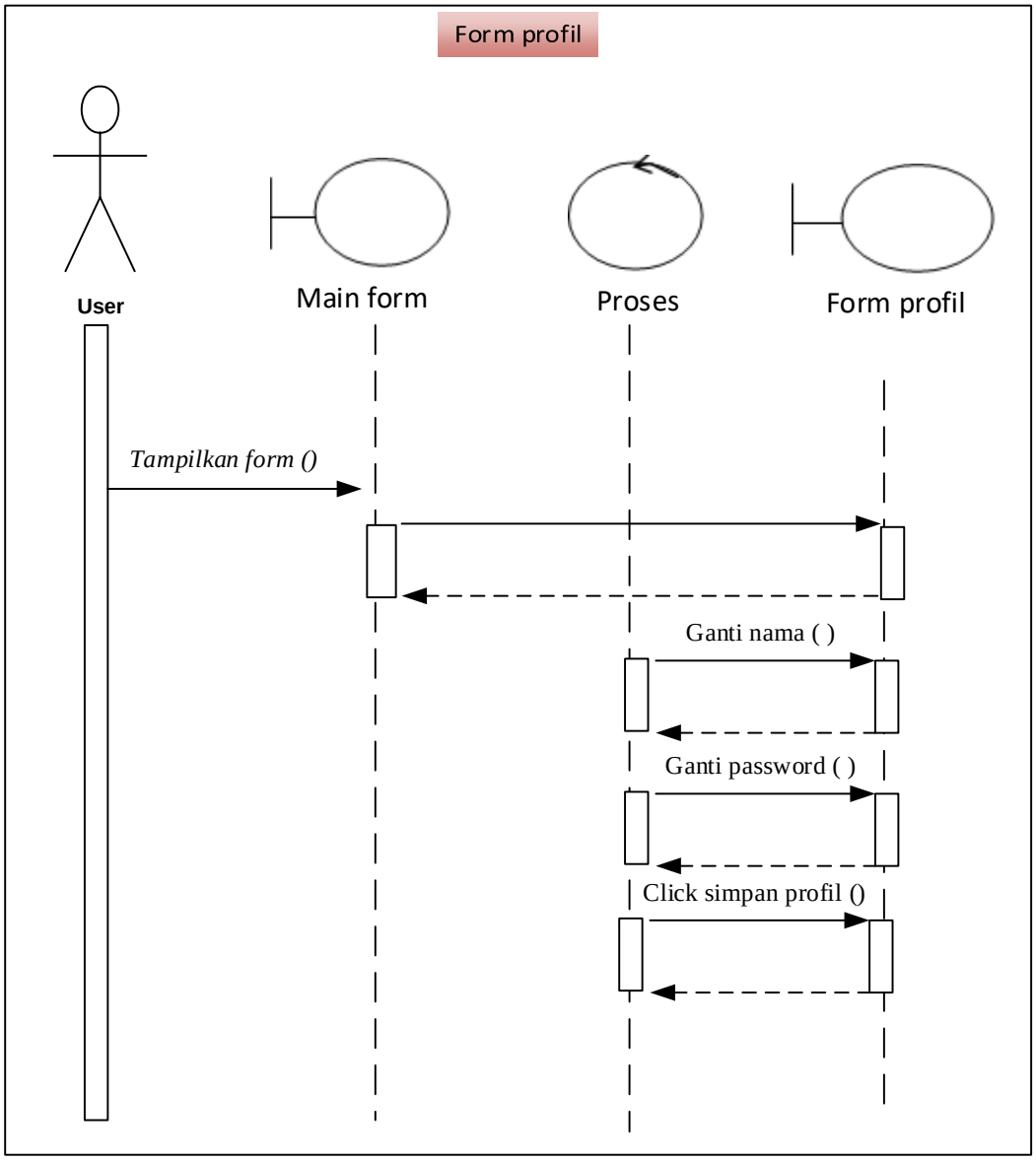
Serangkaian kegiatan sistem yang dilakukan oleh user pada form kontak dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.12 berikut :



Gambar III.12. *Sequence Diagram* Form Kontak

4. *Sequence Diagram* Profil

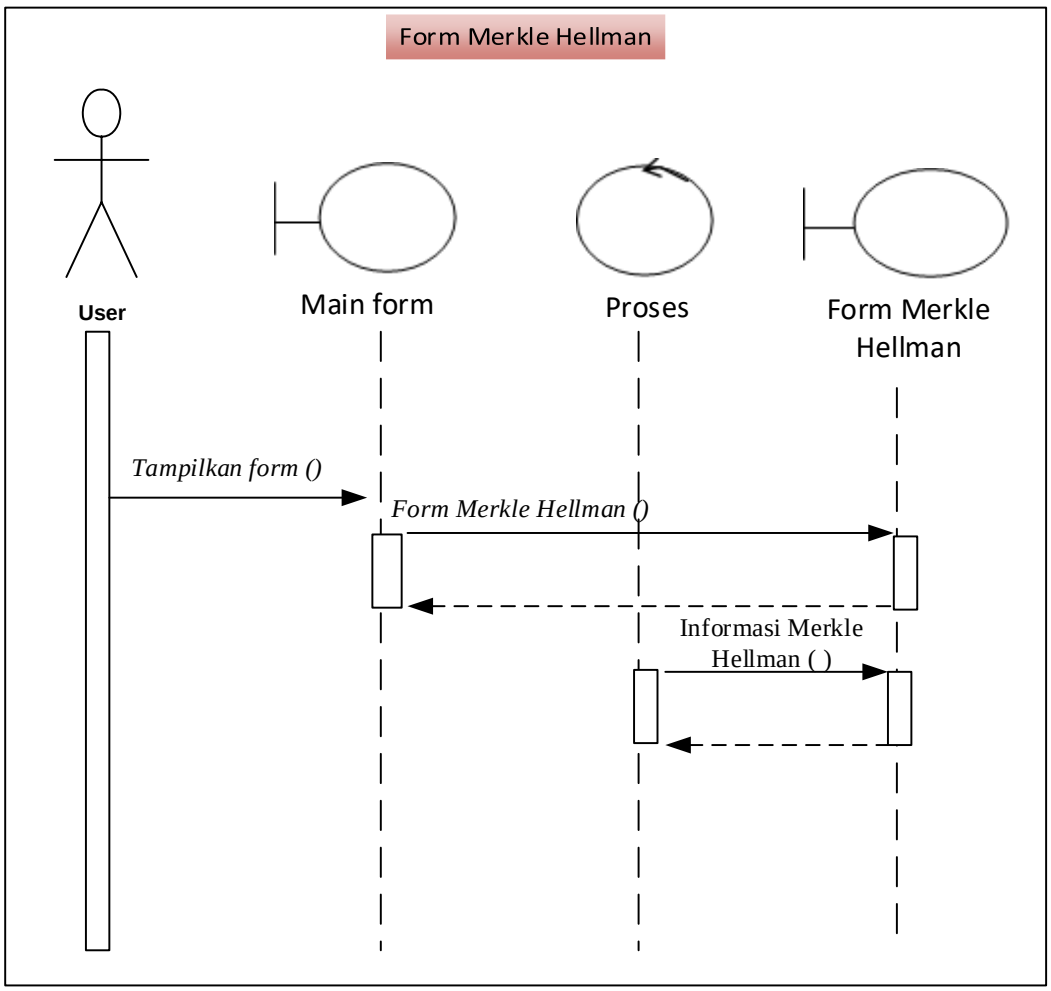
Serangkaian kegiatan sistem yang dilakukan oleh user pada form profil dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.13 berikut :



Gambar III.13. *Sequence Diagram* Form Profil

5. *Sequence Diagram Merkle Hellman*

Serangkaian kegiatan sistem yang dilakukan oleh user pada form *Merkle Hellman* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.14 berikut :



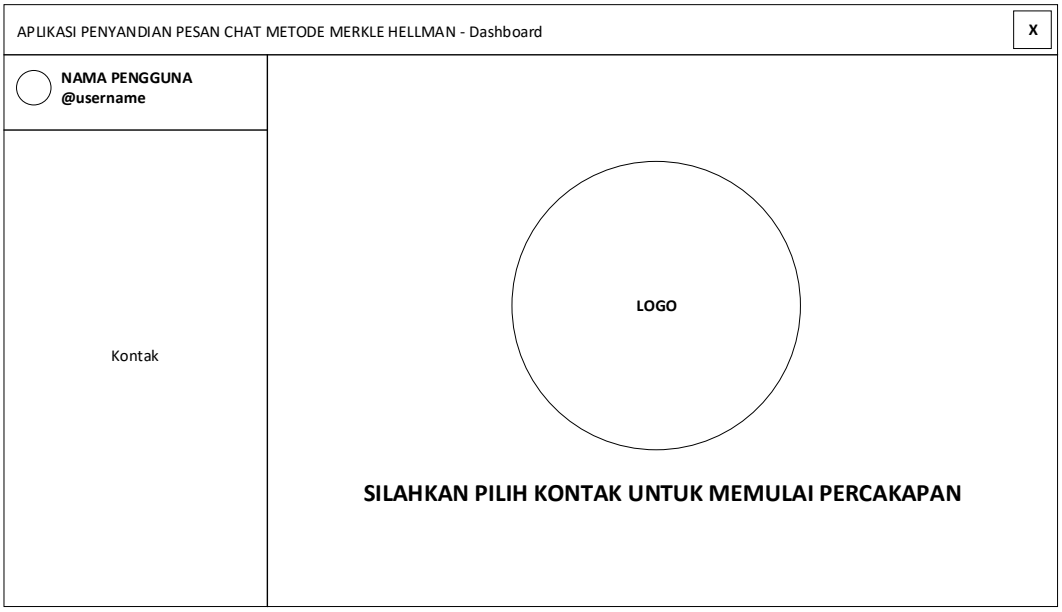
Gambar III.14. *Sequence Diagram* Form *Merkle Hellman*

III.3.2. Desain Sistem Secara Detail

Tahap perancangan berikutnya yaitu desain sistem secara detail yang meliputi desain sistem.

1. *Desain* Dashboard

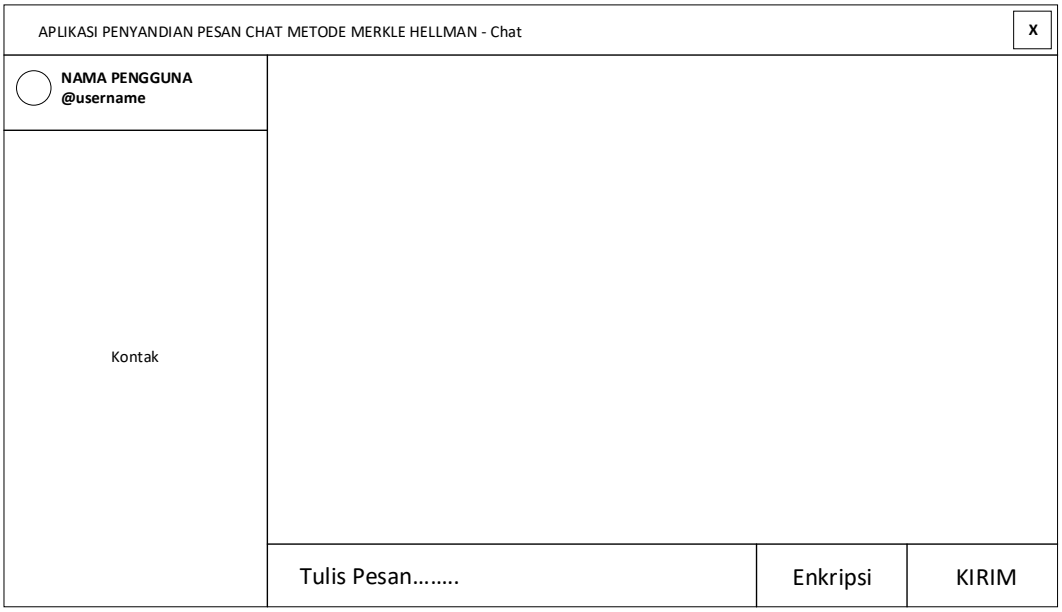
   Serangkaian kegiatan sistem yang dilakukan oleh user pada form dashboard dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.15 berikut :



Gambar III.15. *Desain* Form Dashboard

2. *Desain* Chat

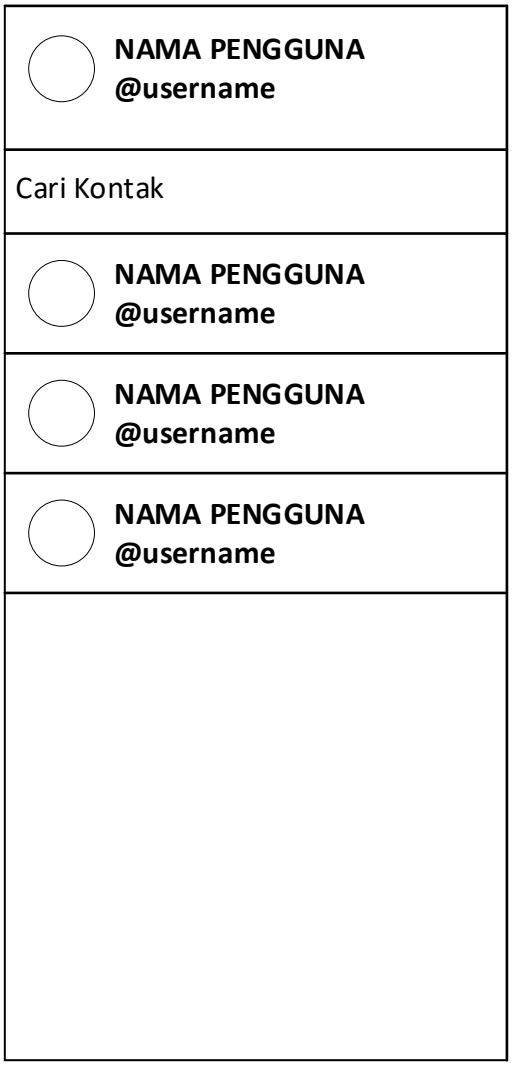
Serangkaian kegiatan sistem yang dilakukan oleh user pada form chat dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.16 berikut :



Gambar III.16. *Desain* Form Chat

3. *Desain* Kontak

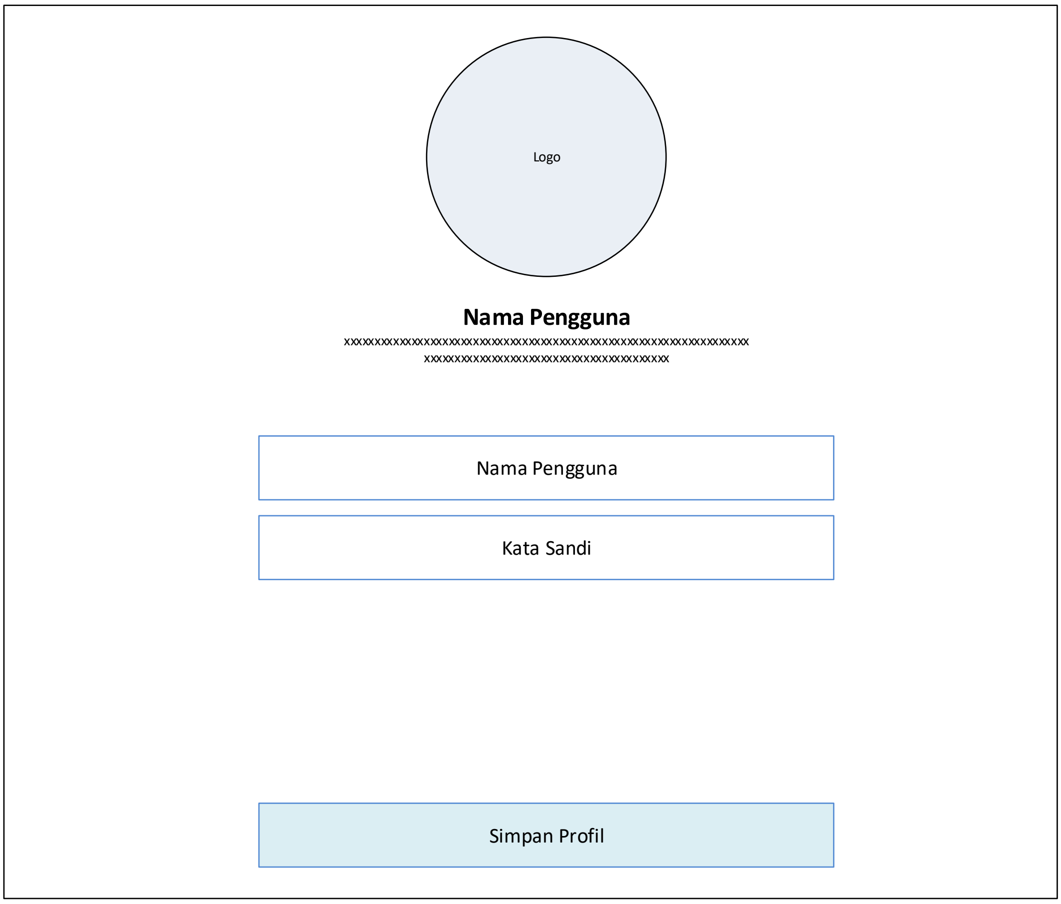
Serangkaian kegiatan sistem yang dilakukan oleh user pada form kontak dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.17 berikut :



Gambar III.17. *Desain* Form Kontak

4. *Desain* Profil

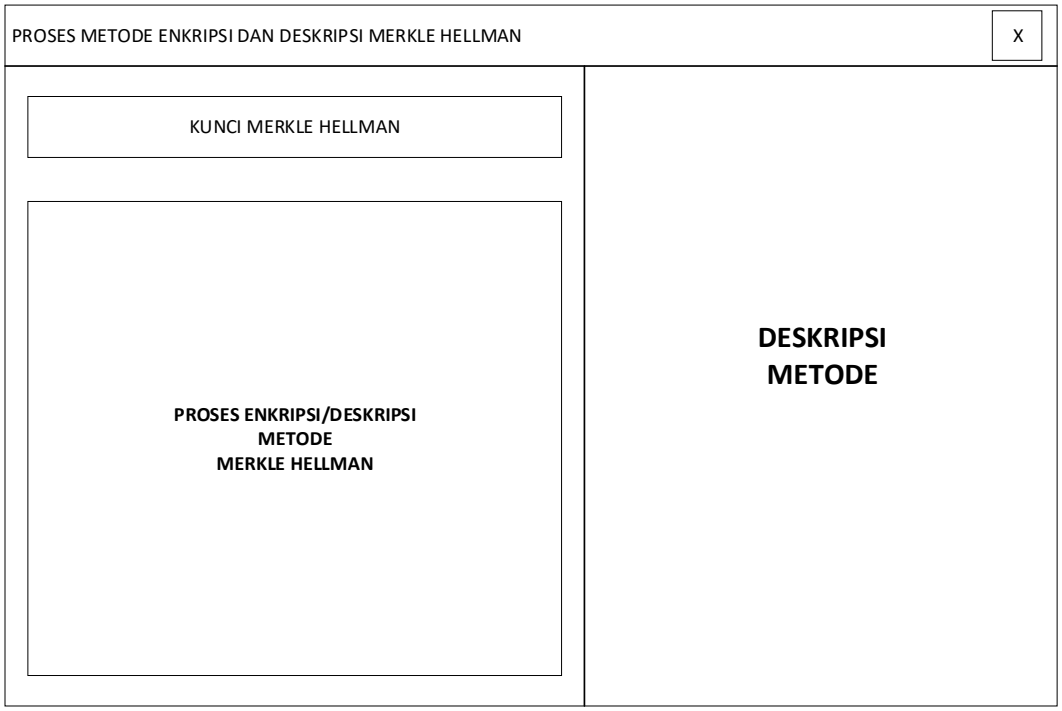
Serangkaian kegiatan sistem yang dilakukan oleh user pada form profil dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.18 berikut :



Gambar III.18. *Desain* Form Profil

5. *Desain Merkle Hellman*

Serangkaian kegiatan sistem yang dilakukan oleh user pada form *Merkle Hellman* dapat diterangkan dengan langkah-langkah *state* berikut, yang ditunjukkan pada gambar III.19 berikut :



Gambar III.19. *Desain* Form *Merkle Hellman*