BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisis Masalah

Surat adalah sarana komunikasi untuk menyampaikan informasi tertulis oleh suatu pihak kepada pihak lain. Fungsinya mencakup lima hal: sarana pemberitahuan, permintaan, buah pikiran dan gagasan, alat bukti tertulis, alat pengingat, bukti historis, dan pedoman kerja. Surat juga dapat dilengkapi oleh tanda tangan pengirim untuk menjamin keaslian surat yang dibuat oleh seseorang atau suatu lembaga. Perkembangan teknologi yang sangat pesat saat ini terutama *internet* membuat proses pembuatan pada surat mulai di alihkan dari proses yang manual ke proses yang lebih terkomputerisasi, yang mana memiliki beberapa kelebihan dan kekurangan.

Surat yang dibuat nantinya akan memberikan kemudahan dalam pemrosesan dan pertukaran informasi. Namun, konsekuensi logis dari hal itu menyebabkan rentannya surat terhadap penyalahgunaan fungsi atau tujuan dari dibuatnya surat tersebut. Adanya modifikasi secara illegal membuat surat tersebut tidak terjamin keasliannya serta lambatnya proses tanda tangan yang dilakukan secara manual membuat proses pembuatan surat menjadi terhambat. Oleh karena itu, adanya perlindungan pada surat sangat diperlukan dan adanya sistem keamanan menjadi hal yang penting untuk menjadi solusi dalam pemecahan masalah.

Adanya tanda tangan digital yang terdapat pada sebuah surat tentu saja sangat membantu dalam mencegah terjadinya penyalahgunaan pada surat. Proses tanda tangan pada sebuah surat biasanya hanya menggunakan cara manual. Hal tersebut menyebabkan rentannya surat untuk di modifikasi atau disalahgunakan. Untuk itu pada sistem yang akan di bangun, proses pembuatan tanda tangan pada surat dilakukan dengan menyisipkan sebuah *Qrcode* yang berisi data pengirim surat berupa id pengirim, nama dan data diri lainnya yang dimiliki oleh pengirim surat. Pada proses pembentukan tanda tangan digital dengan *Qrcode* adalah dengan mengenkripsi data berupa id pengirim dan beberapa data diri pengirim. Selanjutnya data hasil enkripsi tersebut di simpan kedalam sebuah *Qrcode.*dan selanjutnya untuk melihat isi dari *Qrcode* tersebut digunakanlah sebuah scanner *Qrcode* yang telah mendukung proses dekripsi didalamnya. Maka saat *Qrcode* di *scan* akan menampilkan data berupa hasil enkripsi yang akan otomatis di dekripsi oleh aplikasi *scanner* dan menampilkan data asli berupa id dan data diri pengirim.

Berdasarkan analisis masalah tersebut, maka pada skripsi ini akan diangkat sebuah judul "**Rancang Bangun Sistem Keabsahan Pengirim Algoritma *Affine Cipher* Dan *Qrcode* Sebagai Bukti Keaslian Surat**". Penulis berharap, semoga dengan adanya sistem ini dapat memberikan pemahaman tentang pentingnya keamanan dalam sebuah surat untuk mencegah adanya modifikasi serta penyalahgunaan secara illegal pada sebuah surat.

III.2. Strategi Pemecahan Masalah

Beberapa strategi pemecahan masalah dalam perancangan keamanan surat dengan *Qrcode* menggunakan algoritma *affine chiper* ini adalah sebagai berikut:

1. Sistem pengaman surat ini dapat digunakan pada semua perangkat yang telah *support* menggunakan *browser*.
2. Sistem yang dibangun ini digunakan untuk mengamankan surat dengan menyisipkan *Qrcode* yang telah disandikan isinya menggunakan algoritma *affine cipher*.
3. Sistem yang dibangun nantinya akan memiliki sebuah *scanner* khusus untuk mendeskripsikan data yang ada pada *Qrcode*.

III.3. Analisa Kebutuhan Sistem

Pembuatan aplikasi ini membutuhkan serangkaian peralatan yang dapat mendukung kelancaran proses pembuatan sistem. Berikut ini aspek-aspek yang di butuhkan.

III.3.1. Perangkat Keras (*Hardware*)

Hardware merupakan komponen yang terlihat secara fisik, yang saling bekerjasama dalam pengolahan data. Spesifikasi *minimum hardware* yang digunakan adalah sebagai berikut :

a. Laptop : *Core i3 Processor*
b. *Hard disk* : 500 GB
c. RAM 4 GB

d. Smartphone Android

III.3.2. Perangkat Lunak (*Software*)

Software adalah intruksi atau program-program komputer yang dapat digunakan oleh komputer dengan memberikan fungsi serta penampilan yang diinginkan. Dalam hal ini *software* yang digunakan dalam perancangan aplikasi adalah:

a. Sistem operasi Windows *10*
b. Sublime Text 3
c. Android Studio
d. XAMPP
e. *Browser*

III.4. Penerapan Algoritma *Affine Cipher*

Metode *Affine Cipher* adalah perluasan dari metode *Caesar Cipher*, keunggulan metode ini terletak pada kuncinya, yaitu nilai integer yang menunjukkan pergeseran karakter-karakter, kekuatan kedua terletak pada barisan bilangan-bilangan yang berfungsi sebagai pengali dengan kunci. Barisan tersebut dapat berbentuk barisan bilangan ganjil, barisan *fibonaci*, barisan bilangan prima, serta deret yang dapat kita modifikasi sendiri.

III.6.1. Langkah-langkah Penerapan *Affine Cipher*

Pada dasarnya *Affine Cipher* merupakan hasil pengembangan *Caesar Cipher* yang mengalikan *plaintext* dengan sebuah nilai *P* dan menambahkannya

dengan sebuah pergeseran *b* menghasilkan *Ciphertext C* dinyatakan dengan fungsi kongruen :

$$C \equiv m\,P + b \pmod{n} \quad \text{..... (7)}$$

Yang mana *n* adalah ukuran alphabet, *m* adalah bilangan bulat yang harus relatif prima dengan *n* (jika tidak relatif prima, maka dekripsi tidak bisa dilakukan) dan *b* adalah jumlah pergeseran (*Caesar Cipher* adalah bentuk khusus dari *Affine Cipher* dengan *m*=1). Untuk melakukan deskripsi, persamaan (2.3) harus dipecahkan untuk memperoleh *P*. Solusi kekongruenan tersebut hanya ada jika inver *m (mod n)*, dinyatakan dengan m^{-1}. Jika m^{-1} ada maka dekripsi dilakukan dengan persamaan sebagai berikut: (Munir, 2006)

$$P \equiv m^{-1}(C - b) \pmod{n} \quad \text{..... (8)}$$

III.6.2. Studi Kasus *Affine Cipher*

Berikut adalah contoh enkripsi dan dekripsi menggunakan metode *Affine Cipher* :

Misalkan *plaintext*

C A N D R A

Yang ekuivalen dengan:

2 0 13 3 17 0 (dengan memisalkan ‘A’ = 0, ‘B’ = 1 dst)

Dienkripsi dengan *Affine Cipher* dengan mengambil *m* = 7 (karena 7 relatif prima dengan 26) dan *b* = 10. Karena alphabet yang digunakan 26 huruf, maka *n* = 26.

Enkripsi *plaintext* dihitung dengan kekongruenan:

$$C \equiv 7P + 10 \pmod{26} \quad \ldots\ldots\ldots\ldots\ldots\ldots (9)$$

Perhitungannya adalah sebagai berikut:

$P_1 = 2 \rightarrow C_1 \equiv 7 .2 + 10 \equiv 24 \pmod{26} \equiv 24$ *(huruf 'Y')*

$P_2 = 0 \rightarrow C_2 \equiv 7 .0 + 10 \equiv 10 \pmod{26} \equiv 10$ *(huruf 'K')*

$P_3 = 13 \rightarrow C_3 \equiv 7 . 13 + 10 \equiv 101 \pmod{26} \equiv 23$ *(huruf 'X')*

$P_4 = 3 \rightarrow C_4 \equiv 7 .3 + 10 \equiv 31 \pmod{26} \equiv 5$ *(huruf 'F')*

$P_5 = 17 \rightarrow C_5 \equiv 7 .17 + 10 \equiv 129 \pmod{26} \equiv 25$ *(huruf 'Z')*

$P_6 = 0 \rightarrow C_2 \equiv 7 .0 + 10 \equiv 10 \pmod{26} \equiv 10$ *(huruf 'K')*

Ciphertext yang dihasilkan adalah:

Y K X F Z K

Untuk melakukan dekripsi, pertama-tama dihitung $7^{-1} \pmod{26}$, yang dapat dihitung dengan memecahkan kekongruenan lanjar:

$7x \equiv 1 \pmod{26}$..... (10)

Solusinya adalah $x \equiv 15 \pmod{26}$ sebab $7.15 = 105 \equiv 1 \pmod{26}$. Jadi, untuk dekripsi digunakan kekongruenan:

$P \equiv 15(C - 10) \pmod{26}$..... (11)

Perhitungannya adalah sebagai berikut:

$C_1 = 24$ $\rightarrow P_1 \equiv 15 \cdot (24 - 10) = 210 \pmod{26} \equiv 2$ (huruf 'C')

$C_2 = 10$ $\rightarrow P_2 \equiv 15 \cdot (10 - 10) = 0 \pmod{26} \equiv 0$ (huruf 'A')

$C_3 = 23$ $\rightarrow P_3 \equiv 15 \cdot (23 - 10) = 195 \pmod{26} \equiv 13$ (huruf 'N')

$C_4 = 5$ $\rightarrow P_4 \equiv 15 \cdot (5 - 10) = -75 \pmod{26} \equiv 3$ (huruf 'D')

$C_5 = 25$ $\rightarrow P_5 \equiv 15 \cdot (25 - 10) = 225 \pmod{26} \equiv 17$ (huruf 'R')

$C_6 = 10$ $\rightarrow P_2 \equiv 15 \cdot (10 - 10) = 0 \pmod{26} \equiv 0$ (huruf 'A')

Plaintext yang dihasilkan adalah

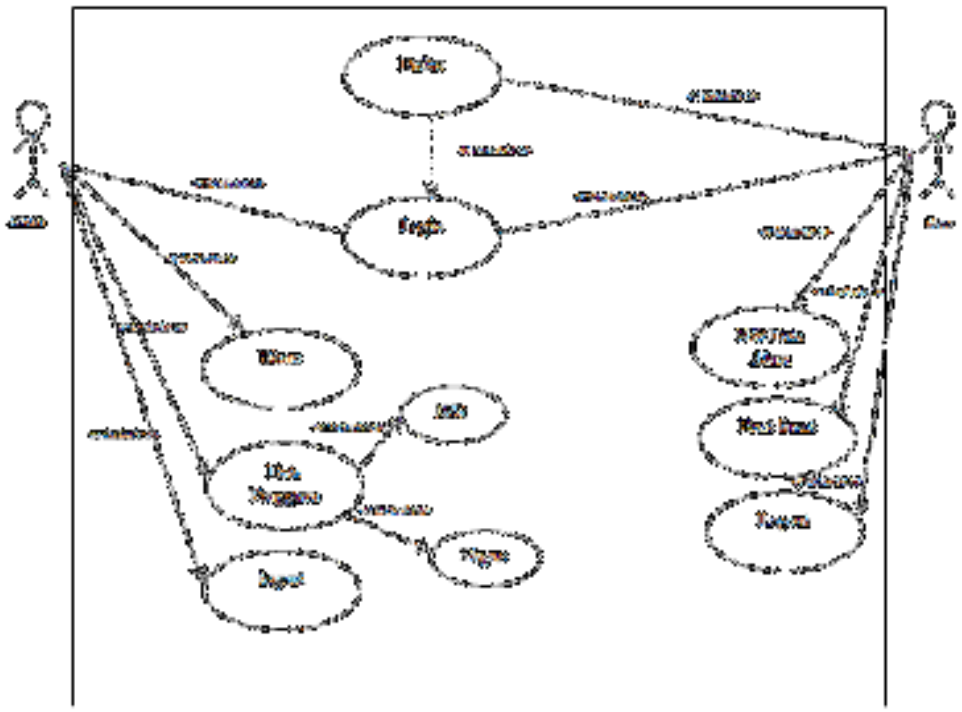
C A N D R A.

III.8. Desain Sistem

Pada tahap ini dirancang sebuah desain dari sistem yang akan dibangun. Bagaimana desain yang akan digunakan pada antarmuka perangkat berbasis *website*. Perancangan sistem yang dirancang terdiri dari *use case*, *activity diagram* serta desain dan penjelasan dari sistem yang dirancang. Berikut adalah perancangannya :

III.8.1. *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 :



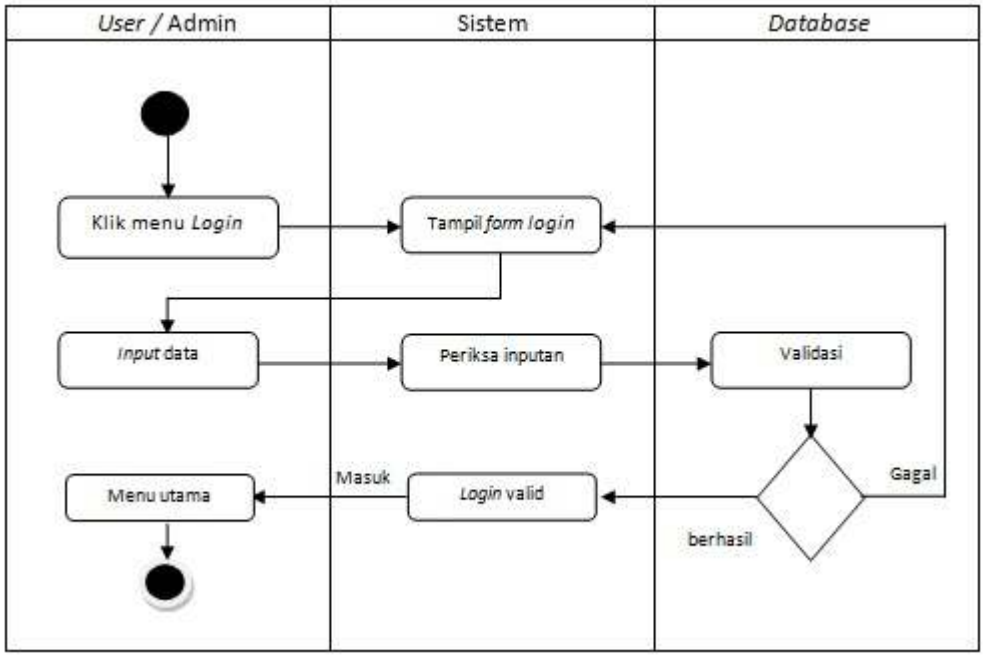
Gambar III.4. *Use Case Diagram User* Dan Admin Pada Sistem Pengamanan Surat Dengan *Qrcode*

III.8.2. *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 :

1. *Activity Diagram Login* pada *user* dan admin

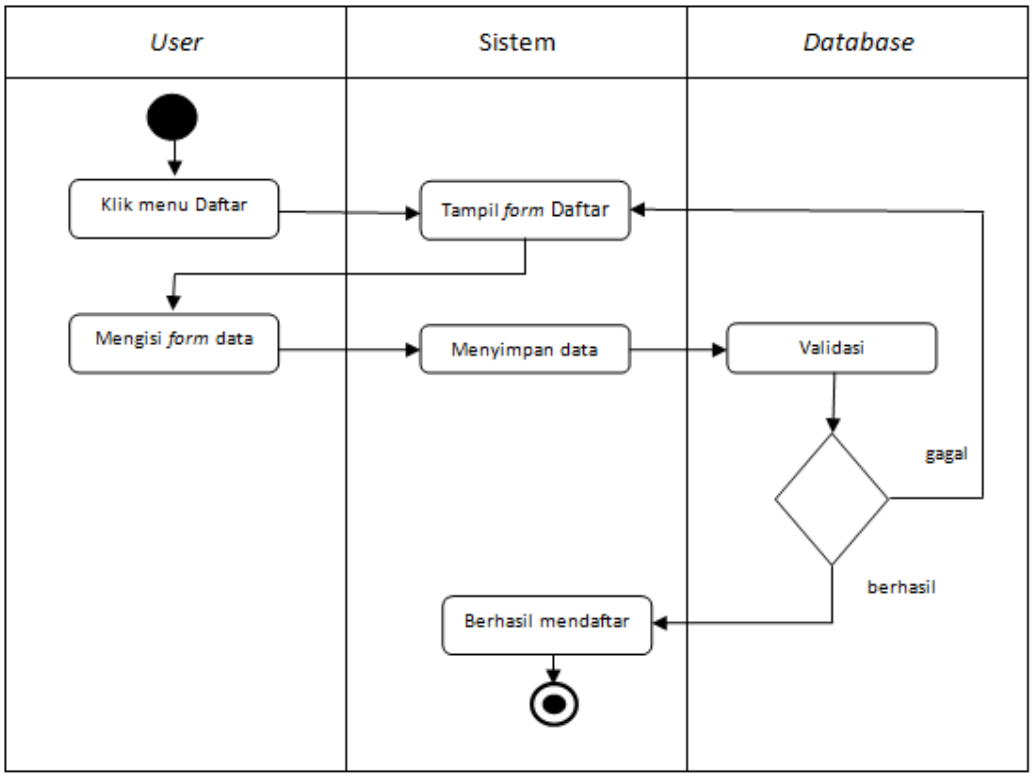
Activity diagram login menggambarkan alir aktifitas untuk melakukan proses *login* oleh *user* dan juga admin yang terdapat pada sistem yang dibuat. Proses *login* dapat dilihat pada gambar III.5.



Gambar III.5. *Activity Diagram Login User* dan Admin

1. *Activity Diagram* Daftar *User*

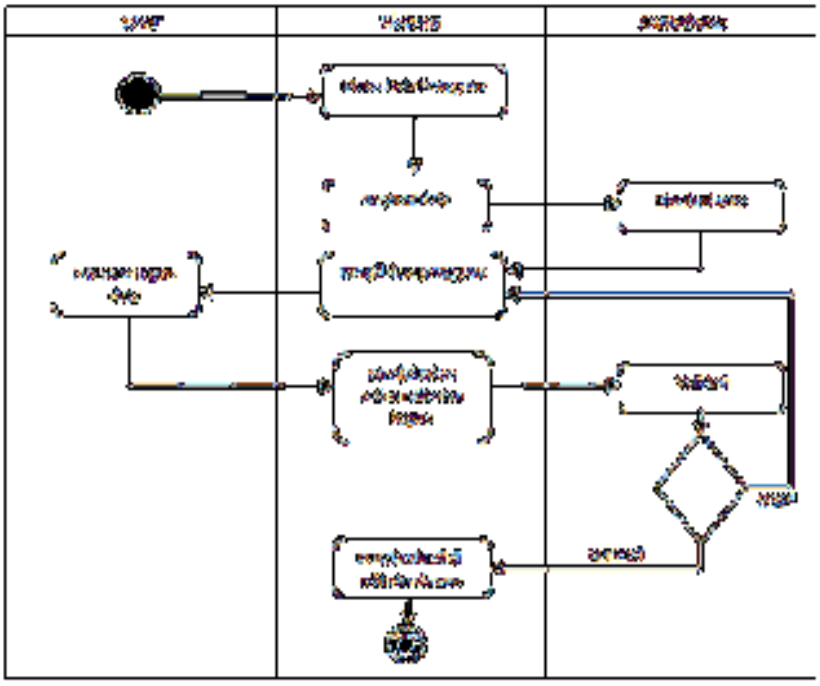
Activity diagram daftar *user* menggambarkan alir aktifitas untuk melakukan proses pendaftaran akun. Proses *activity diagram* dapat dilihat pada gambar III.6.



Gambar III.6. *Activity Diagram* Daftar Pada *User*

2. *Activity Diagram* Data Pengguna Pada Admin

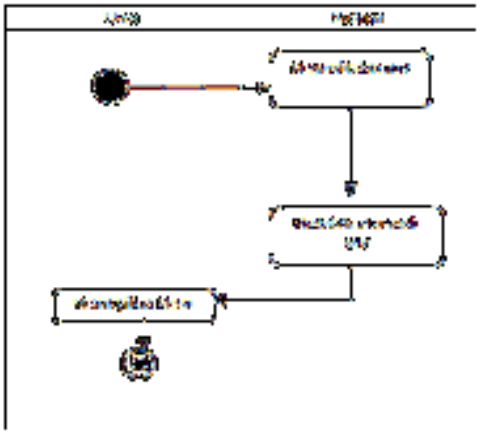
Activity diagram data pengguna pada admin menggambarkan alir aktifitas untuk melakukan proses pemeriksaan dan pengolahan akun *user* pada admin. Proses *activity diagram* dapat dilihat pada gambar III.7.



Gambar III.7. *Activity Diagram* Data Pengguna Pada Admin

3. *Activity Diagram* Edit Data Akun Pada *User*

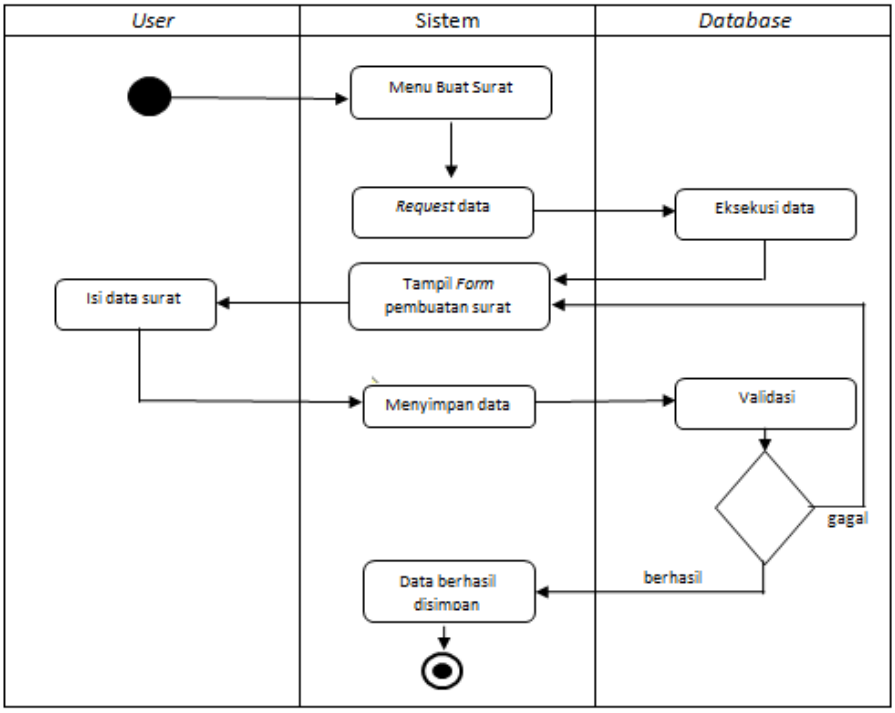
Activity diagram edit data akun pada *user* menggambarkan alir aktifitas untuk melakukan proses perubahan dan pengolahan data dari *user*. Proses *activity diagram* dapat dilihat pada gambar III.8.



Gambar III.8. *Activity Diagram* Edit Data Akun Pada *User*

4. *Activity Diagram* Buat Surat Pada *User*

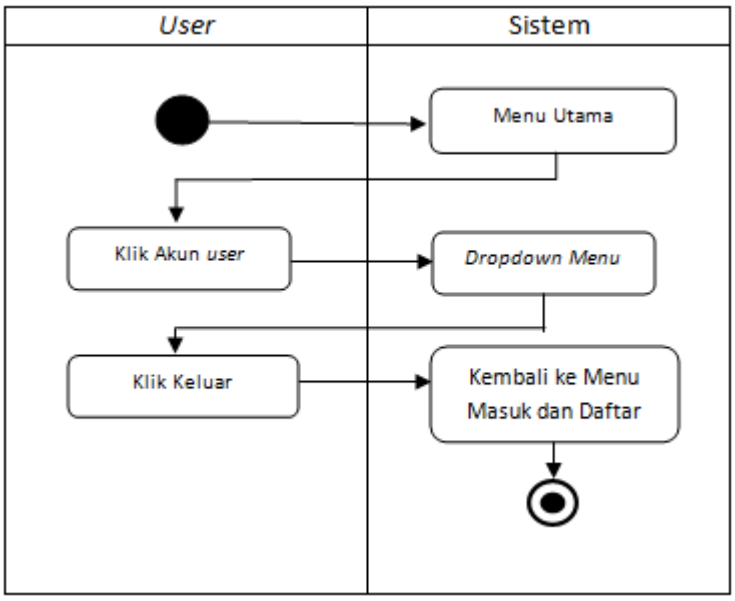
Activity diagram buat surat pada *user* menggambarkan alir aktifitas dalam melakukan pembuatan surat. *Activity diagram* dapat dilihat pada gambar III.9.



Gambar III.9. *Activity Diagram* Buat Surat Pada *User*

5. *Activity Diagram Logout* Pada Admin Dan *User*

Activity diagram logout pada admin dan *user* merupakan *activity diagram* pada saat *user* atau admin ingin keluar dari sistem. *Activity diagram logout* ditunjukkan pada gambar III.10.



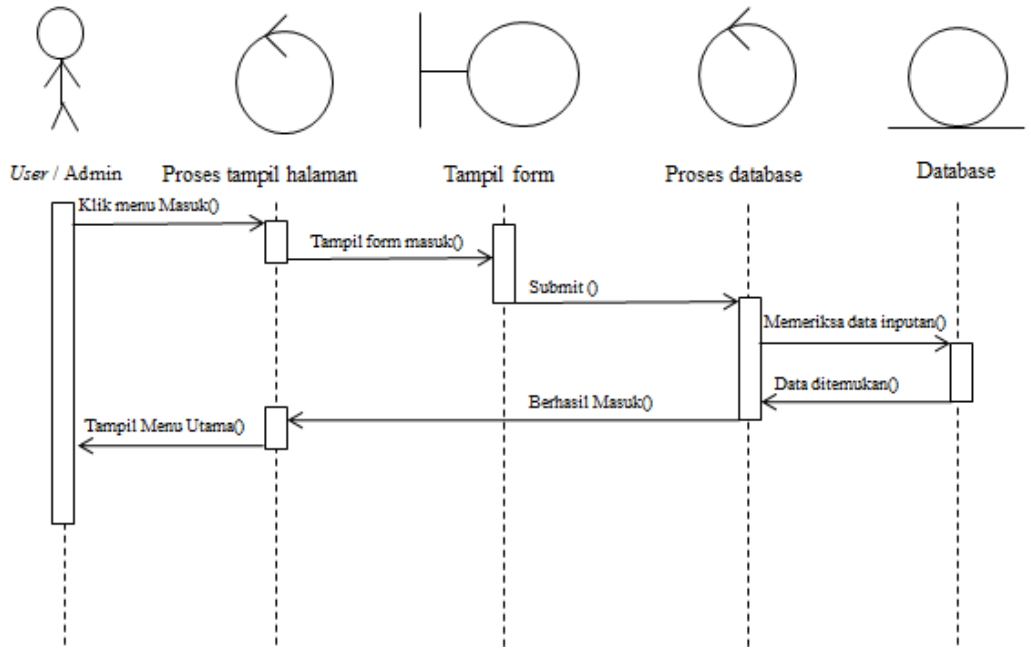
Gambar III.10. *Activity Diagram Logout* Pada Admin Dan *User*

III.8.3. *Sequence Diagram*

Sequence diagram pada aplikasi yang akan dibuat yaitu sebagai berikut :

III.8.3.1. *Sequence Diagram Login*

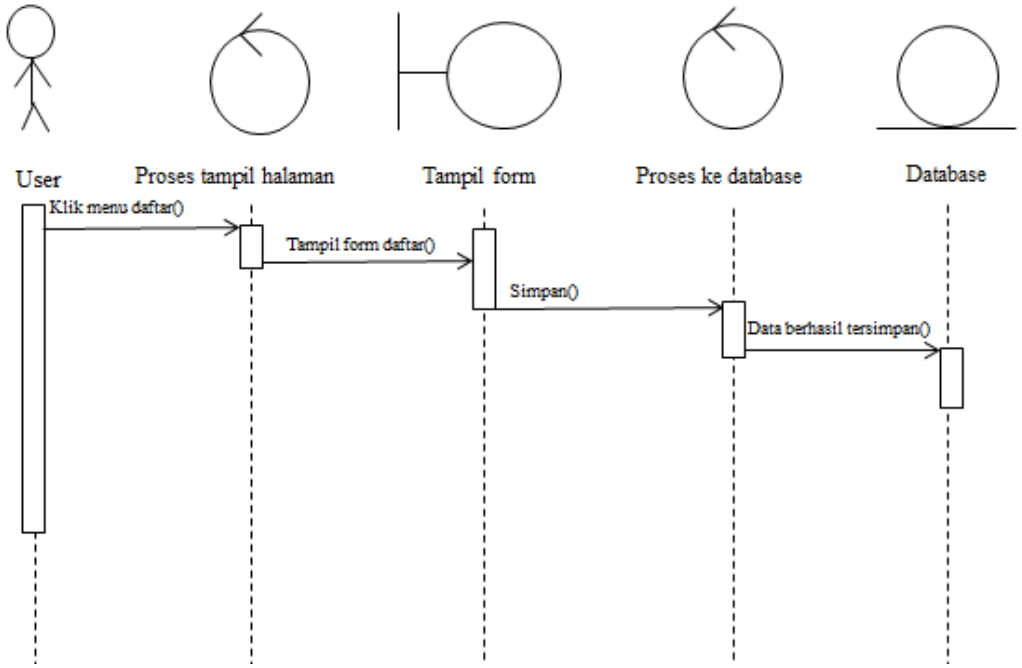
Sequence diagram masuk (*login)* menggambarkan interaksi yang terjadi pada saat melakukan proses masuk (*login)* kedalam sistem. *Sequence diagram* masuk (*login)* ditunjukkan pada gambar III.11.



Gambar III.11. *Sequence Diagram Login*

III.8.3.2. *Sequence Diagram* Daftar *User*

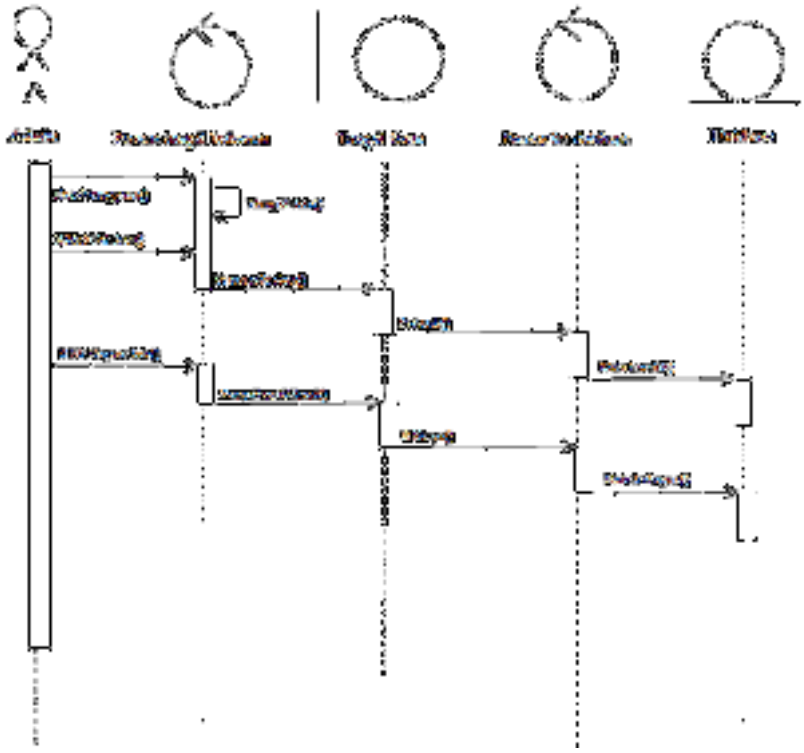
Sequence diagram daftar menggambarkan interaksi yang terjadi pada saat melakukan proses pendaftaran. *Sequence diagram* daftar ditunjukkan pada gambar III.12.



Gambar III.12. *Sequence Diagram* Daftar

III.8.3.3. *Sequence Diagram* Data Pengguna Pada Admin

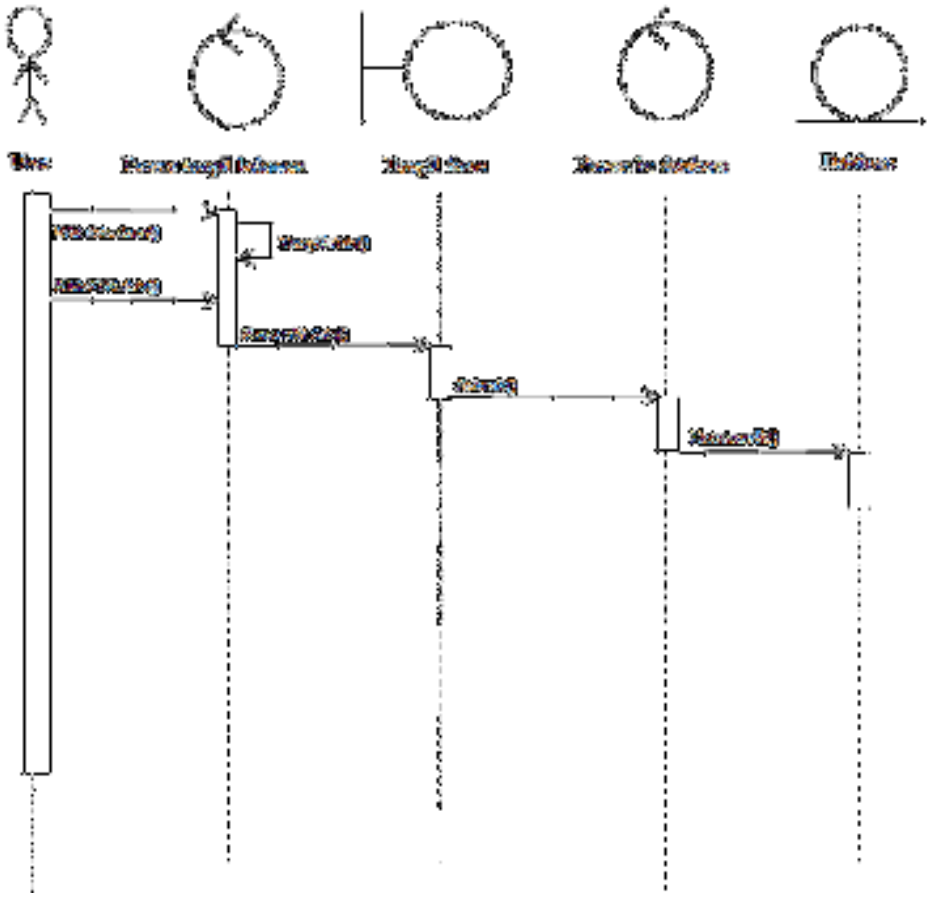
Sequence diagram data pengguna menggambarkan interaksi yang terjadi pada saat melakukan proses pengolahan data pengguna atau *user* yang terdaftar di sistem. *Sequence diagram* data pengguna ditunjukkan pada gambar III.13



Gambar III.13 *Sequence Diagram* Data Pengguna Pada Admin

III.8.3.5. *Sequence Diagram* Edit Data Akun Pada *User*

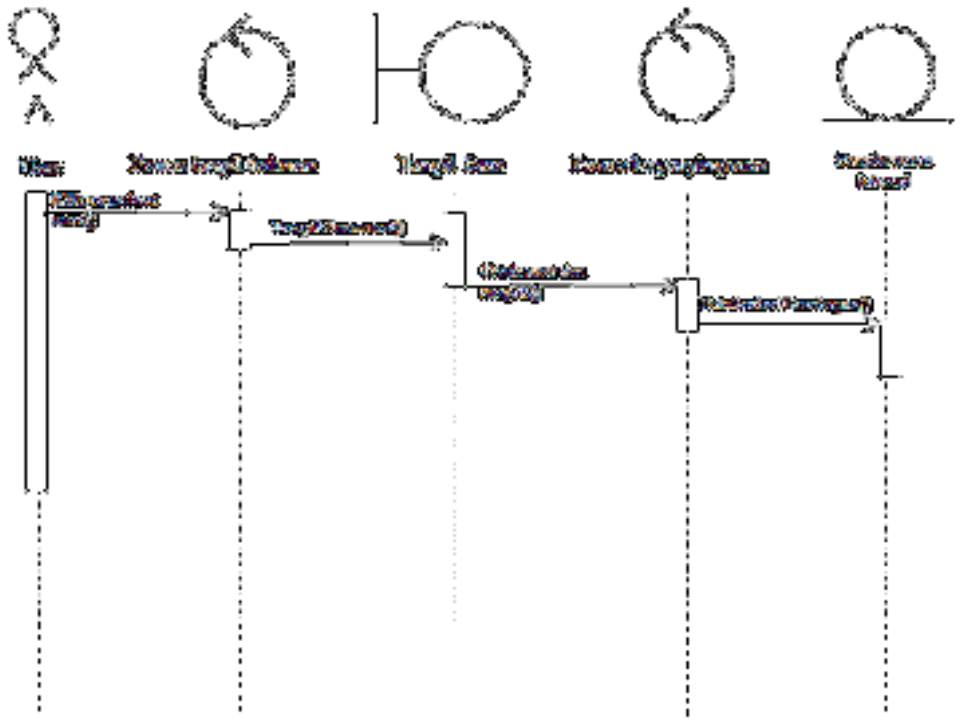
Sequence diagram edit data akun menampilkan proses pengolahan data yang dapat dilakukan *user*. *Sequence diagram* edit data akun ditunjukkan pada gambar III.14.



Gambar III.14. *Sequence Diagram* Edit Data Akun Pada *User*

III.8.3.6. *Sequence Diagram* Buat Surat

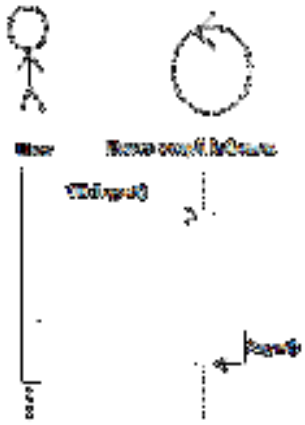
Sequence diagram buat surat menampilkan informasi proses pembuatan surat yang dilakukan oleh *user*. *Sequence diagram* buat surat ditunjukkan pada gambar III.15.



Gambar III.15. *Sequence Diagram* Buat Surat Pada *User*

III.8.3.7. *Sequence Diagram Logout*

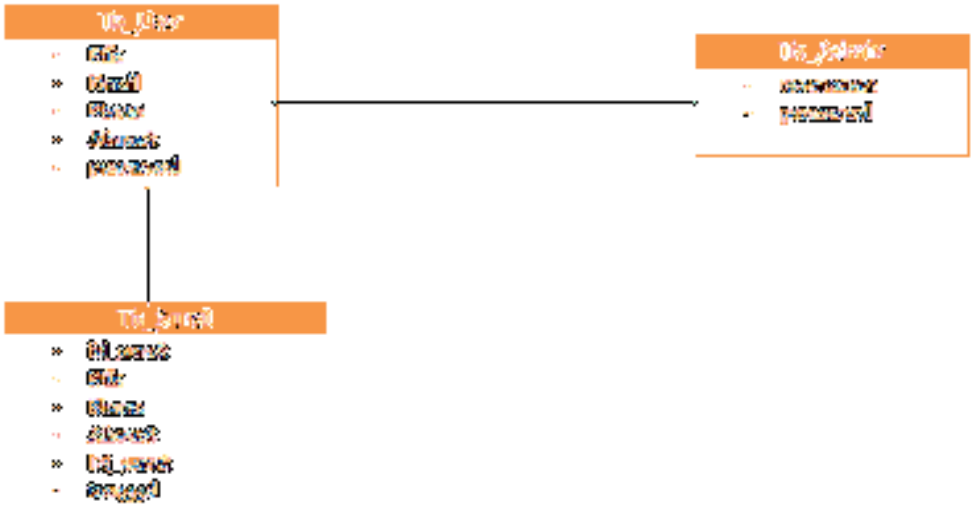
Sequence diagram logout menampilkan untuk admin dan *user* keluar dari akun sistem. *Sequence diagram logout* ditunjukkan pada gambar III.16.



Gambar III.16. *Sequence Diagram* Logout

III.8.4. *Class Diagram*

Class diagram dekripsi kelompok obyek-obyek dengan property, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class* diagram dapat memberikan pandangan global atas sebuah *system*. *Class diagram* yang terdapat pada aplikasi yaitu sebagai berikut :



Gambar III.17. *Class Diagram* Pada Sistem Pembuatan Surat

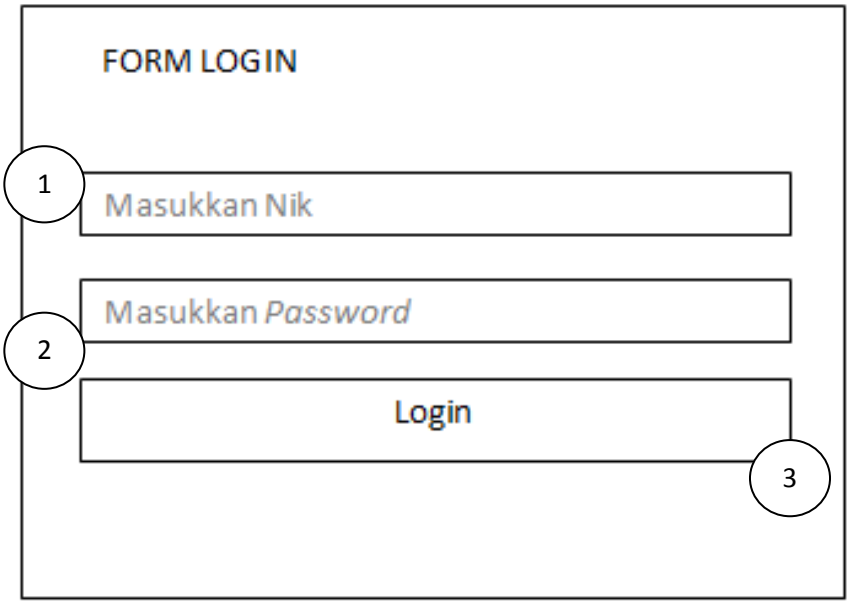
III.9. Desain *User Interface*

Antarmuka pemakai (*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 rancang bangun sistem pembuatan surat dengan menggunakan *qrcode* sebagai bukti keaslian surat yaitu :

III.9.1. Desain *User Interface*

Berikut tampilan sistem untuk *user* berbasis *website*

1. Desain Halaman *Login*

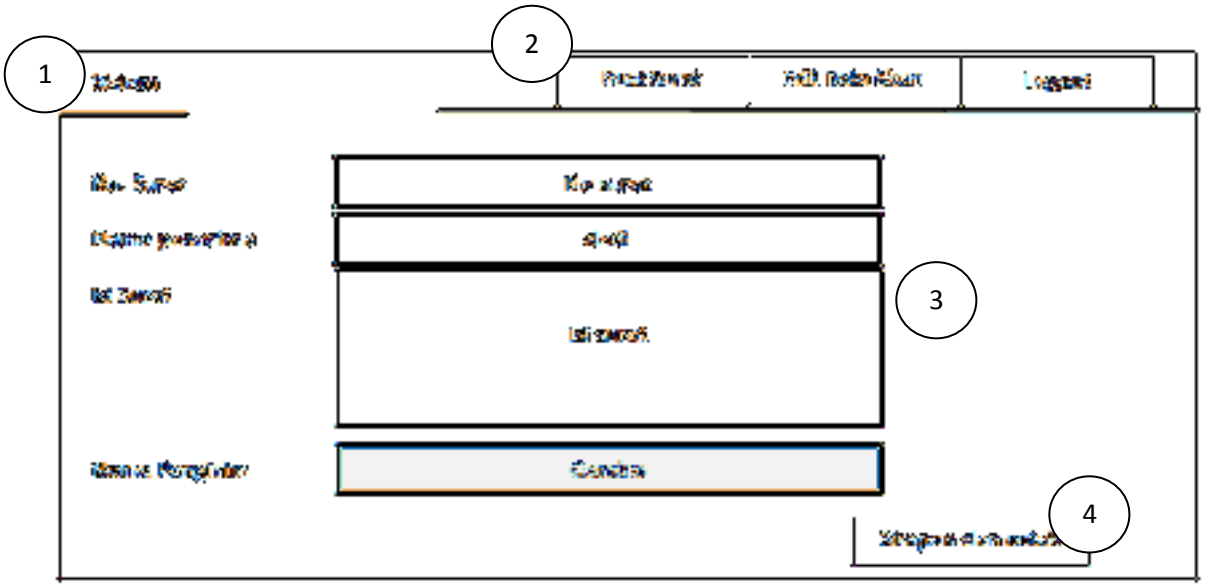


Gambar III.18. Desain Halaman *Login User*

Merupakan tampilan rancangan halaman utama saat dijalankan. Adapun keterangannya sebagai berikut :

1) *Field* untuk mengisi *username* pengguna
2) *Field* untuk mengisi *password* pengguna
3) Tombol untuk memproses data *login.*

2. Desain Halaman Buat Surat

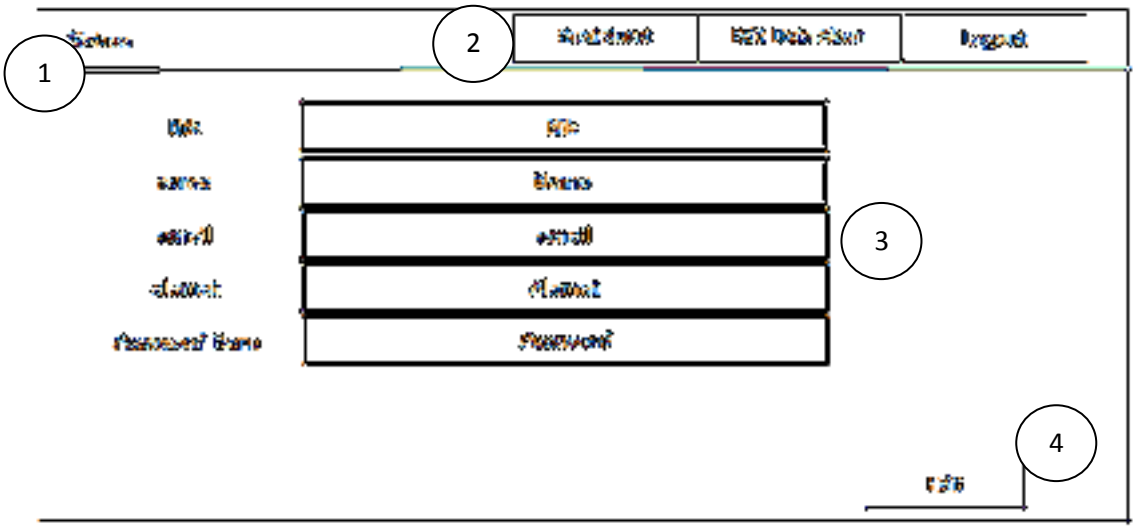


Gambar III.19. Desain Halaman Buat Surat

Keterangan tampilan halaman buat surat, yaitu :

1) Nama dari sistem yang dibangun.
2) Menu yang ada pada sistem untuk *user.*
3) Form untuk mengisi surat
4) Tombol untuk memproses data

3. Desain Halaman Edit Data Akun



Gambar III.20. Desain Halaman Edit Data Akun

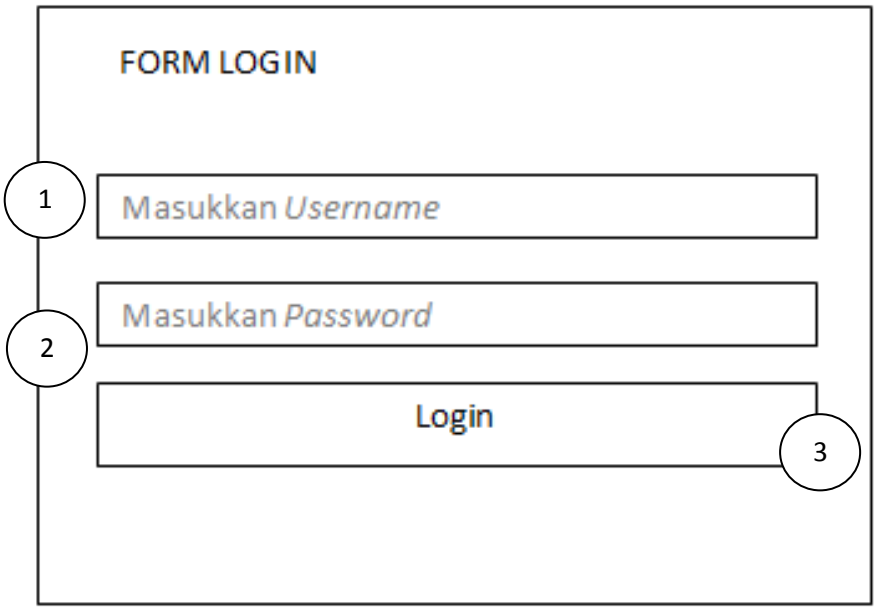
Keterangan tampilan halaman edit data akun, yaitu :

1) Nama dari sistem yang dibangun.
2) Menu yang ada pada sistem untuk *user*.
3) Form untuk mengisi data akun yang baru
4) Tombol untuk memproses data

III.9.2.. Desain *Interface* Pada Admin

Berikut tampilan sistem untuk admin :

1. Desain Halaman *Login*

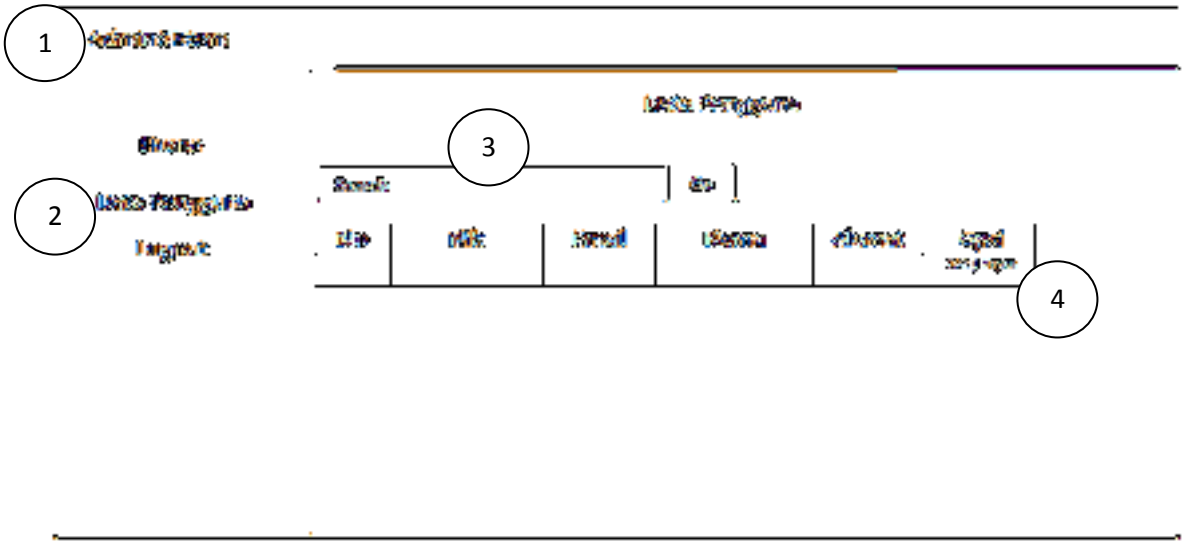


Gambar III.21. Desain Halaman *Login* Admin

Merupakan tampilan rancangan halaman *login* saat dijalankan. Adapun keterangannya sebagai berikut :

1) *Field* untuk mengisi *username* admin
2) *Field* untuk mengisi *password* admin
3) Tombol untuk memproses data *login.*

2. Desain Halaman Data Pengguna

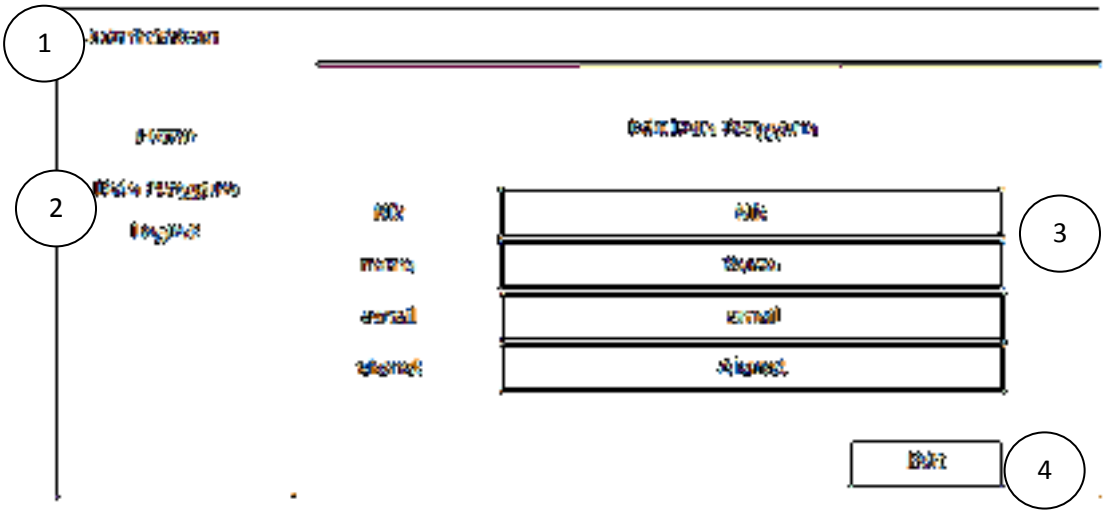


Gambar III.22. Desain Halaman Data Pengguna Pada Admin

Keterangan tampilan data pengguna, yaitu :

1) Nama dari sistem yang dibangun.
2) Menu yang ada pada sistem pembuatan surat untuk admin.
3) Form untuk mencari data
4) Tampilan data pengguna

3. Desain Halaman Edit Data Pengguna Pada Admin

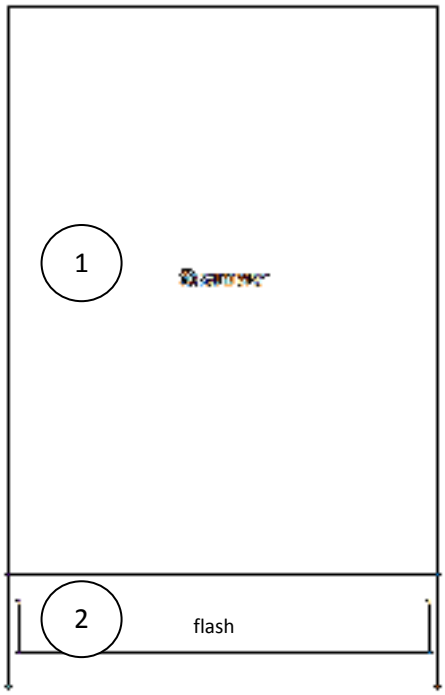


Gambar III.23. Desain Halaman Edit Data Pengguna Pada Admin

Keterangan tampilan halaman edit data pengguna, yaitu :

1) Nama dari sistem yang dibangun.
2) Menu yang ada pada sistem admin.
3) Form input data
4) Tombol menyimpan hasil edit data

4. Desain Halaman *Scanner*



Gambar III.24. Desain Halaman *Scanner*

Keterangan tampilan halaman *Scanner*, yaitu :

1) Kamera untuk meng*scan qrcode*.
2) Tombol untuk menyalakan flash