BAB IV

ANALISIS DAN PERANCANGAN

4.1 Analisis Sistem Yang Sedang Berjalan

perancangan sistem lebih terarah dan sistem akan dapat berjalan sesuai kebutuhan pengguna apabila dilakukan analisis terlebih dahulu. Analisis adalah tahapan awal yang dilakukan dalam perancangan sistem sebagai landasan utama perancangan sistem.

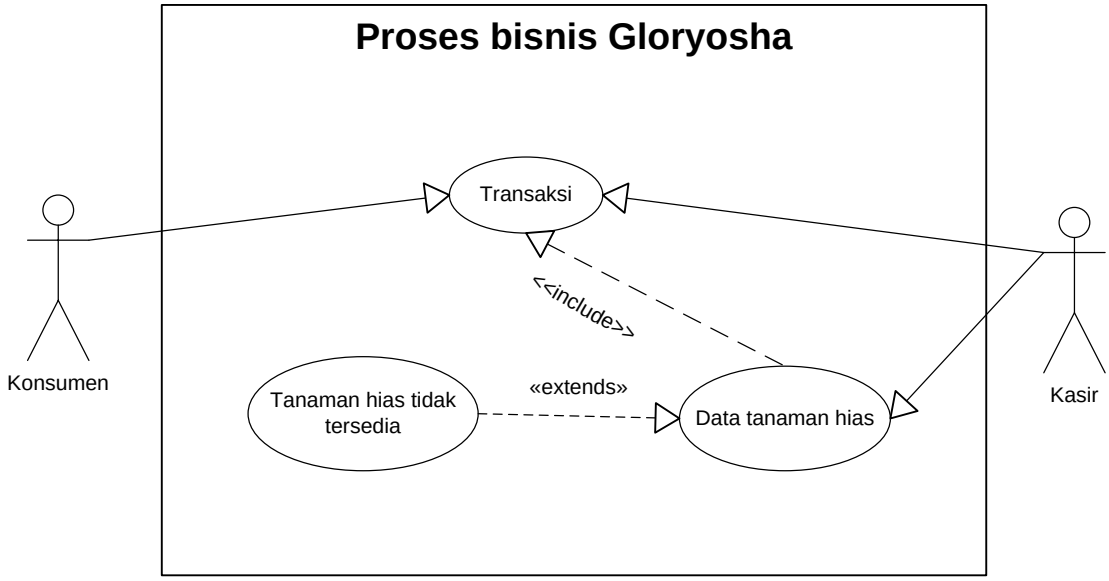
4.2 Analisis Prosedur Yang Sedang Berjalan

Gloryosha memiliki prosedur-prosedur dalam melaksanakan kegiatan bisnis sehari-hari. Oleh karena itu dalam mengembangkan sistem informasi yang dibuat prosedur-prosedur yang berjalan sehari-harinya di Gloryosha perlu di analisis sebagai gambaran untuk sistem yang akan dikembangkan. Adapun alat bantu analisis yang penulis gunakan adalah sebagai berikut :

4.2.1 Use Case Diagram

Dengan menggunakan *use case diagram* kita akan mendapatkan banyak informasi yang sangat penting yang berkaitan dengan aturan-aturan bisnis. Dalam hal ini, setiap obyek yang berinteraksi dengan sistem merupakan aktor untuk sistem kita, sementaraa *use case diagram* merupakan deskripsi lengkap tentang interaksi yang terjadi antara para aktor dengan sistem yang sedang kita kembangkan. *use case diagram* merupakan hal yang sangat penting pada tahap

analisis. Sedangkan *use case diagram* proses bisnis yang sedang berjalan di Gloryosha adalah :



Gambar 4.1. *Use case diagram* proses bisnis yang sedang berjalan Pada Gloryosha

4.2.1.1 Skenario Use Case

Untuk memperinci perilaku sistem dari masing-masing *use case* yang ada maka penulis mendeskripsikan skenario *use case* sebagi berikut :

Nama *use case* : Transaksi

Aktor : Konsumen dan kasir

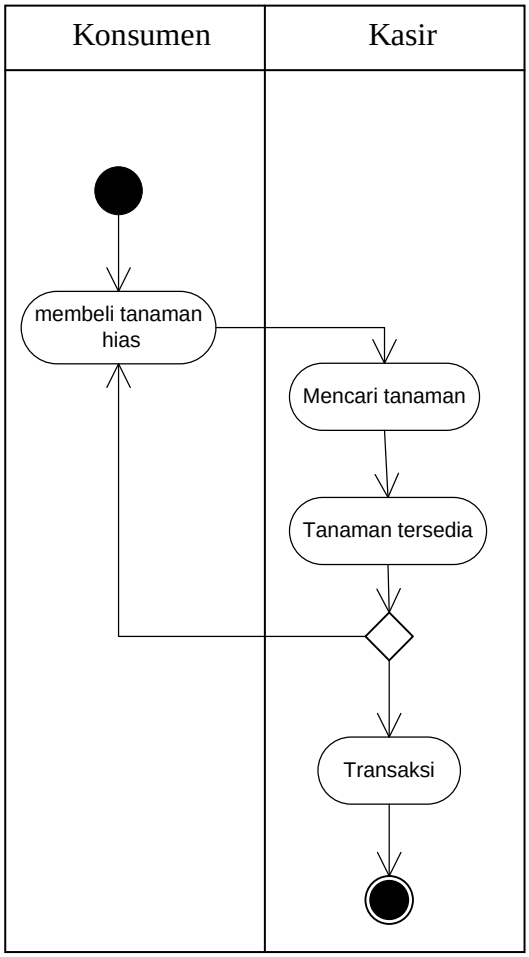
Tujuan : Membayar pembelian tanaman hias

a. Konsumen telah menerima tanaman hias
b. Kasir menghitung total yang harus dibayar konsumen
c. Konsumen melakukan pembayaran

4.2.2 Activity Diagram

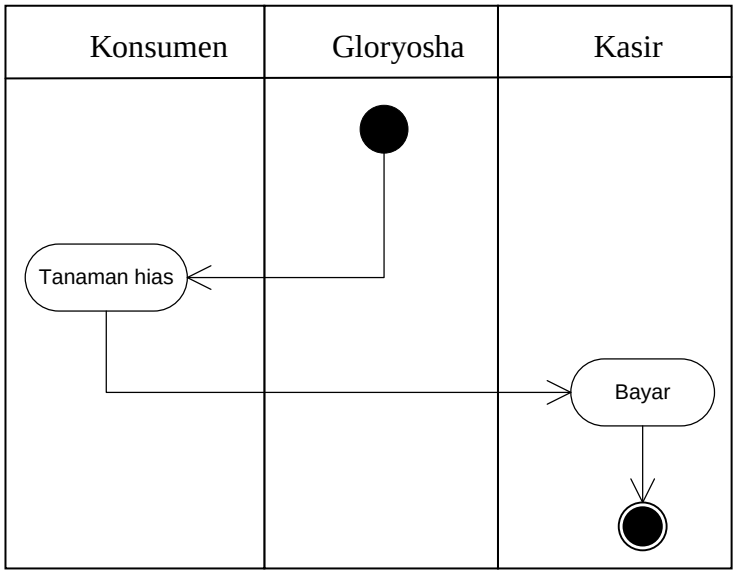
Untuk lebih merinci *use case diagram* proses bisnis yang berjalan, maka perlu dibuat *activitty diagram* sebagai berikut :

1. Activity diagram Penjualan tanaman hias



Gambar 4.2. *Activity diagram* Penjualan tanaman hias yang sedang berjalan Pada Gloryosha

2. Activity diagram transaksi penjualan tanaman hias



Gambar 4.3. *Activity diagram* Transaksi yang sedang berjalan Pada Gloryosha

4.2.3 Evaluasi Sistem Yang Sedang Berjalan

Setelah melakukan analisis terhadap proses bisnis yang berjalan, penulis menemukan beberapa masalah yang dapat mengakibatkan penurunan pelayanan terhadap kosumen. Adapun masalah-masalah yang terjadi adalah sebagai berikut :

1. Minimnya informasi dalam pengelolaan stok tanaman hias.
2. Kegiatan transaksi yang dinilai lambat karena penghitungan masih manual.
3. Belum adanya laporan transaksi penjualan tanaman hias.

4.3 perancangan Sistem

Sebelum memulai perancangan terlebih dahulu dilakukan perencanaan untuk melakukan pengembangan sistem. Perancangan sistem ini menyangkut estimasi dari kebutuhan-kebutuhan fisik dan tenaga kerja yang dibutuhkan untuk mendukung pengembanagn sistem. Perancangan sistem dapat diartikan sebagai :

1. Tahap setelah analisis dari siklus pengembangan sistem.
2. Pendefinisian atas kebutuhan-kebutuhan fungsional sistem.
3. Persiapan untuk rancang bangun implementasi.
4. Menggambarkan sistem yang akan dibentuk.
5. Konfigurasi komponen *software* dan *hardware* sistem.

4.3.1 Tujuan Perancangan Sistem

Perancangan sistem ini bertujuan untuk memberikan gambaran secara umum dan terperinci kepada pengguna tentang sistem yang akan dibangun dan mengidentifikasikan komponen-komponen sistem informasi yang akan dirancang secara detail. Tujuan perancangan sistem informasi penjualan tanaman hias ini akan dijabarkan sebagai berikut :

1. Merubah proses-proses pencatatan yang menyangkut pengolahan data menjadi terkomputerisasi.
2. Meningkatkan pelayanan terhadap konsumen dan memberikan kemudahan dalam proses pengolahan dan pengelolaan data.

4.3.2 Gambaran Umum Sistem Yang Diusulkan

Sistem informasi ini dirancang dengan tujuan membantu perusahaan untuk meningkatkan pelayanan terhadap konsumen dan mempercepat proses bisnis yang berjalan. Sistem informasi ini memiliki kegunaan untuk mengelola data tanaman hias, transaksi penjualan dan laporan.

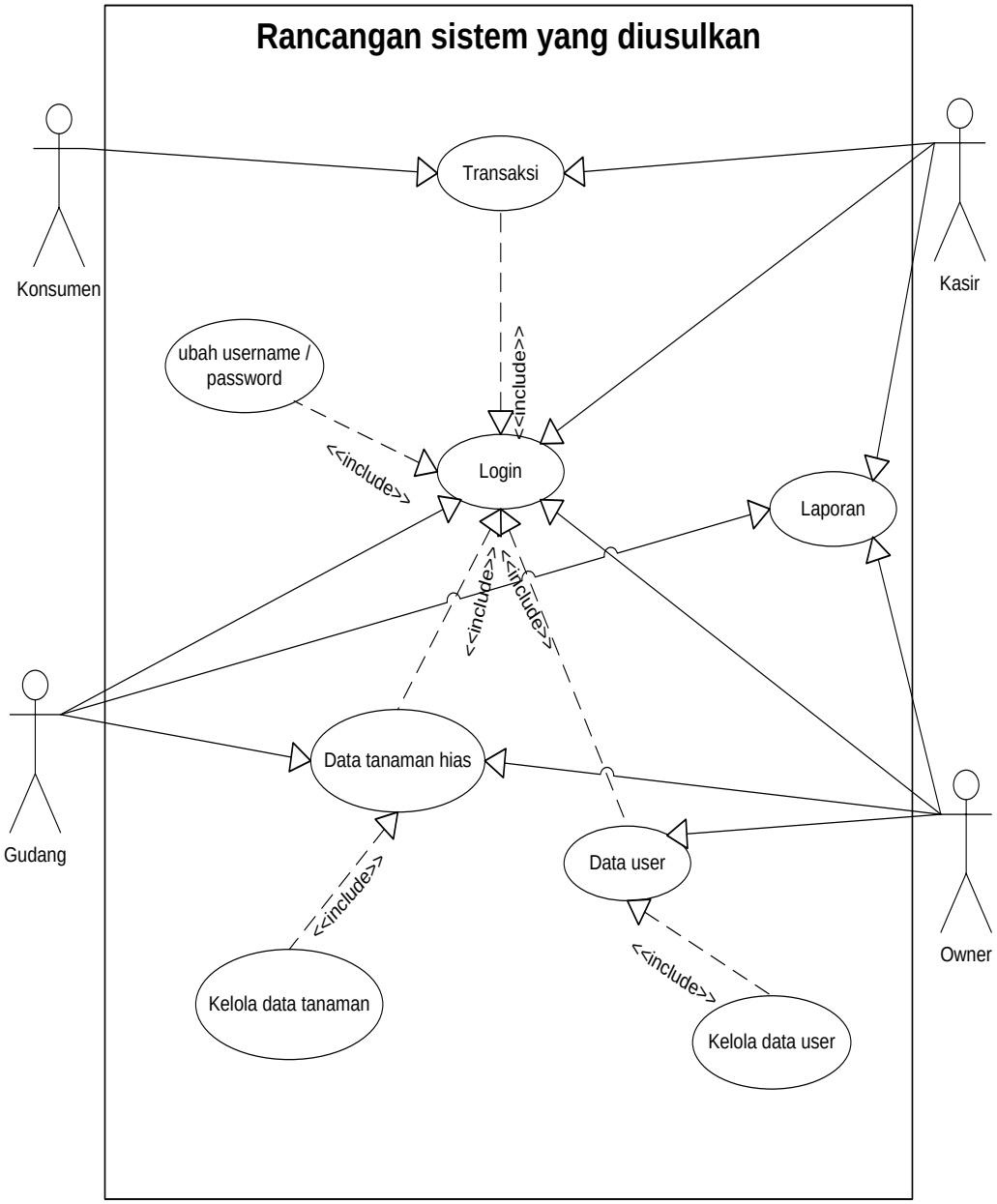
Fungsi yang terdapat dalam sistem informasi ini dibagi menjadi dua bagian. Pada bagian kasir sistem ini menpunyai fungsi pelayanan transaksi dan menyimpan data transaksi secara otomatis kedalam database. Sehingga dapat meminimalkan waktu yang digunakan dalam kegiatan menghitung jumlah transaksi dan menulis data transaksi kedalam buku besar. Pada bagian gudang sistem informasi ini dapat secara otomatis menghitung keluar-masuknya tanaman hias tanpa harus menghitung satu per satu tanaman hias secara manual yang tentu saja akan menghabiskan banyak waktu. Sistem informasi ini juga dapat menampilkan data tanaman hias yang memiliki stok kurang sehingga dapat dijadikan peringatan kepada parusahaan untuk segera menambah stok sebelum stok benar-benar habis. Dengan demikian diharapkan sistem informasi ini dapat membantu menngatasi masalah yang ada.

4.3.3 Perancangan Prosedur Yang Diusulkan

Dalam perancangan ini penulis menggunakan alat bantu berupa *use case diagram, activity diagram, collaboration diagram, sequence diagram* dan *class diagram* untuk menghasilkan sistem yang baik dan sesuai dengan kebutuhan.

4.3.3.1 Use Case Diagram

Berikut ini adalah *use case diagram* perancangan sistem yang diusulkan oleh penulis pada Gloryosha :



Gambar 4.4. *Use Case diagram* sistem yang diusulkan Pada Gloryosha.

4.3.3.1.1 Skenario Use Case

Skenario *use case* yang penulis usulkan adalah sebagai berikut:

1. Nama use case : Login.

   Aktor : *Owner*, Kasir, dan gudang.

   Tujuan : Otentifikasi *user.*

Tabel 4.1. Skenario *Use Case Login User*

User	Sistem
1. Masuk ke halaman *login user.*	
2. *M*engisi *username* dan *password* pada form *login* akun member.	
	3. Verifikasi *Username* dan *password* pada sistem database.
	4. Jika *account* terdaftar, maka secara otomatis akan dapat mengakses menu utama, jika *account* tidak terdaftar maka sistem akan memberikan peringatan 'Username atau Password tidak valid' dan kembali ke halaman *login user.*
5. *User* mengakses tampilan menu utama sesuai hak akses.	

2. Nama use case : Transaksi tanaman hias.

   Aktor : Konsumen, Kasir, dan Sistem.

   Tujuan : Melayani penjulan tanaman hias yang konsumen inginkan.

Tabel 4.2. Skenario *Use Case* Transaksi Tanaman

Konsumen	**Kasir**	**Sistem**
1. Permintaan tanaman.		
	2. Bagian kasir menerima permintaan tanaman.	
	3. Bagian Kasir melakukan proses transaksi masuk ke menu transaksi tanaman.	
		4. Menampilkan halaman transaksi untuk menginput data tanaman dan menampilkan nomor transaksi secara otomatis.
	5. Bagian Kasir menginputkan nama tanaman untuk ditampilkan.	
		6. Proses cari dari nama tanaman akan menampilkan kode, harga, ukuran dan jumlah tanaman yang tersedia

Konsumen	Kasir	Sistem
		7. Apabila stok tanaman masih tersedia di gudang, maka transaksi dilanjutkan, tetapi apabila stok habis maka permintaan dikembalikan kepada konsumen.
	8. Bagian Kasir menginputkan jumlah tanaman yang dibeli konsumen.	
	9. Bagian Kasir menambahkan ke daftar beli	
		10. Verifikasi data pembelian tanaman.
	11. Bagian Kasir memproses dengan 'BAYAR'.	
		12. Menghitung secara otomatis semua total pembelian tanaman.
		13. Konfirmasi berhasil 'Input sukses'
		14. Menyimpan data yang sudah *valid* di *database*
		15. Menampilkan struk penjualan tanaman untuk dicetak kasir

Konsumen	Kasir	Sistem
	16. Mencetak struk pembayaran	
17. Menerima Struk		

3. Nama use case : Data tanaman hias.

   Aktor : Bagian gudang.

   Tujuan : Mengelola data tanaman hias.

Tabel 4.3. Skenario *Use Case* pengelolaan tanaman

Gudang	Sistem
1. Bagian gudang akan menambah Tanaman baru untuk dijual maka pilih fungsi tambah tanaman pada menu tanaman	
	2. Menampilkan tambah tanaman untuk input jenis tanaman baru
3. Menginputkan seluruh data yang diperlukan tentang tanaman baru.	
Gudang	**Sistem**
4. Konfrimasi dengan 'SIMPAN'	
	5. Proses penyimpanan data tanaman
	6. Apabila tanaman berhasil di input akan menampilkan input suskses, apabila tidak kembali ke input data tanaman
	7. Menyimpan data yang sudah *valid* di *database*

Gudang	Sistem
8. Bagian Gudang akan mengganti detail produk lama maka bagian gudang akan *update* Tanaman lama tersebut dengan cara mengubahnya	
	9. Menampilkan tampilan *update* Tanaman
10. Bagian gudang mencari tanaman berdasarkan kode tanaman yang akan di update datanya atau secara langsung memilih data tanaman yang sudah ada pada tabel tanaman	
11. Memulai pengubahan dengan 'Update'	
	12. Menampilkan Data-data tanaman yang akan di update secara otomatis
13. Merubah data tanaman yang akan di update	
14. konfirmasi dengan'*UPDATE'*	
	15. Proses pengupdatean tanaman
	16. Apabila tanaman berhasil di update akan menampilkan update berhasil, dan bila gagal kembali ke update data tanaman
	17. Menyimpan data *valid* di *database*
18. Bagian Gudang yang mempunyai akses tanaman dalam menu utama untuk menambah Stok	
	19. Menampilkan form tambah stok
20. Menambah stok tanaman yang akan segera habis	

Gudang	Sistem
21. Konfirmasi dengan 'TAMBAH STOK TANAMAN'	
	31. Proses penambahan stok tanaman
	32. Apabila stok tanaman berhasil di tambah akan menampilkan tambah stok berhasil, dan bila gagal kembali ke tambah stok tanaman
	33. Valid data base

4. Nama use case : Laporan.

   Aktor : *Owner*, kasir, dan gudang.

   Tujuan : Membuat laporan atau mengecek data tanaman hias, data transaksi tanaman hias.

Table 4.4 Skenario *Use Case* Laporan

User	Sistem
1. Bagian gudan,kasir,admin masuk ke menu laporan.	
	2. Menampilkan menu laporan.
3. Memasukan parameter tanggal .	
	4. Mensortir data sesuai tanggal yang sudah di tentukan.

User	**Sistem**
	5. Apabila data yang sesuai tanggal ada akan di tampilkan,dan bila tidak akan kembali ke menu laporan.
	6. Cetak laporan

5. Nama use case : Data user.

   Aktor : *Owner*.

   Tujuan : Mengelola data user.

Table 4.5 Skenario *Use Case* kelola data user.

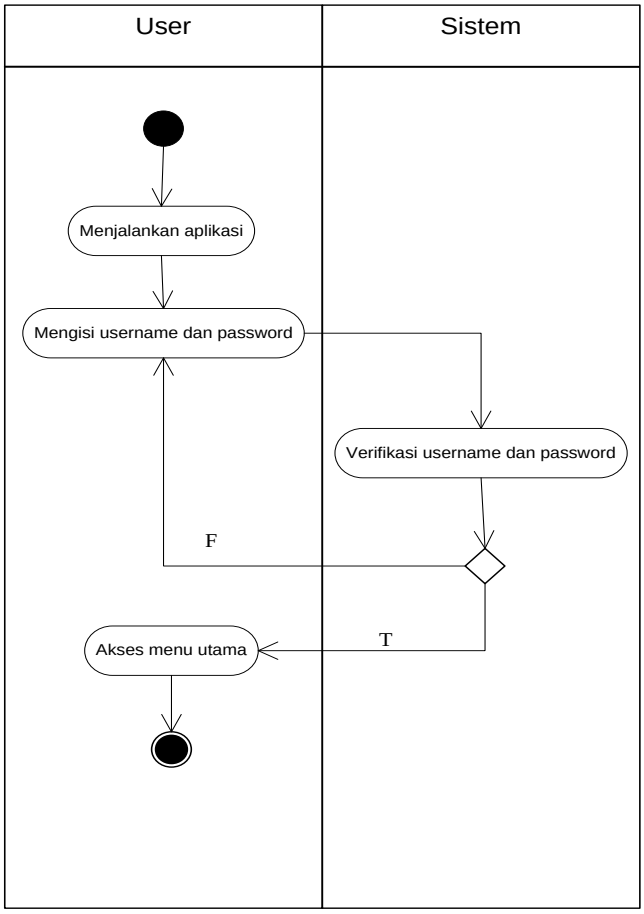
Owner	**Sistem**
1. *Owner* akan menambahkan *user* baru, tekan tombol tambah user pada form data user.	
	2. Memunculkan tampilan tambah user untuk input data user baru.
3. Menginputkan data user baru.	
4. Konfirmasi dengan “simpan”.	
	5. Proses penyimpaan data user.
	6. Jika tanaman berhasil di input tampil input suskses, apabila tidak kembali ke input data *user*.

Owner	Sistem
	7. Menyimpan data yang sudah *valid* di *database.*
8. *Owner* akan mengganti detail *user* lama maka *owner* akan *update user* lama tersebut dengan cara mengubahnya.	
	9. Menampilkan tampilan *update user*.
10. *Owner* memilih data *user* yang sudah ada pada tabel data *user*.	
11. Memulai pengubahan dengan ‘Update’.	
	12. Menampilkan Data-data *user* yang akan di *update* secara otomatis.
13. Merubah data *user* yang akan di update *owner* menginputkan *detail* data *user* yang baru.	
14. konfirmasi dengan‘*UPDATE’*.	
	15. Proses pengupdatean *user*.
	16. Apabila data *user* berhasil di *update* akan menampilkan *update* berhasil, dan bila gagal kembali ke *update* data *user*.

4.3.3.2 Activity Diagram

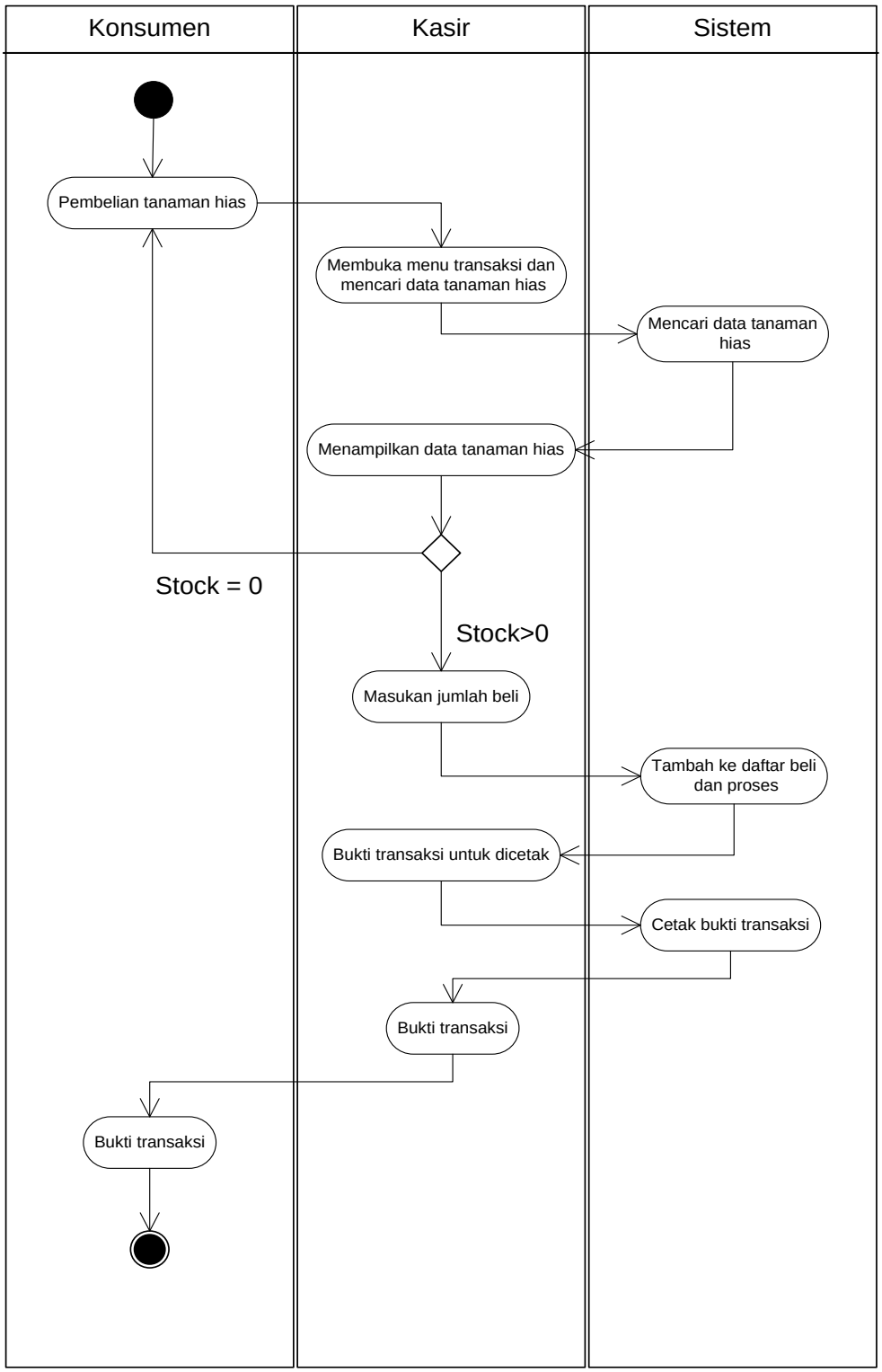
Activity diagram menggambarkan rangkaian aliran dari aktifitas, mendeskripsikan aktifitas yang terjadi dalam suatu proses atau operasi, sehingga dapat juga untuk aktifitas lainnya seperti *use case* atau interaksi. Berikut adalah *activity diagram* yang diusulkan :

1. Activity diagram login.



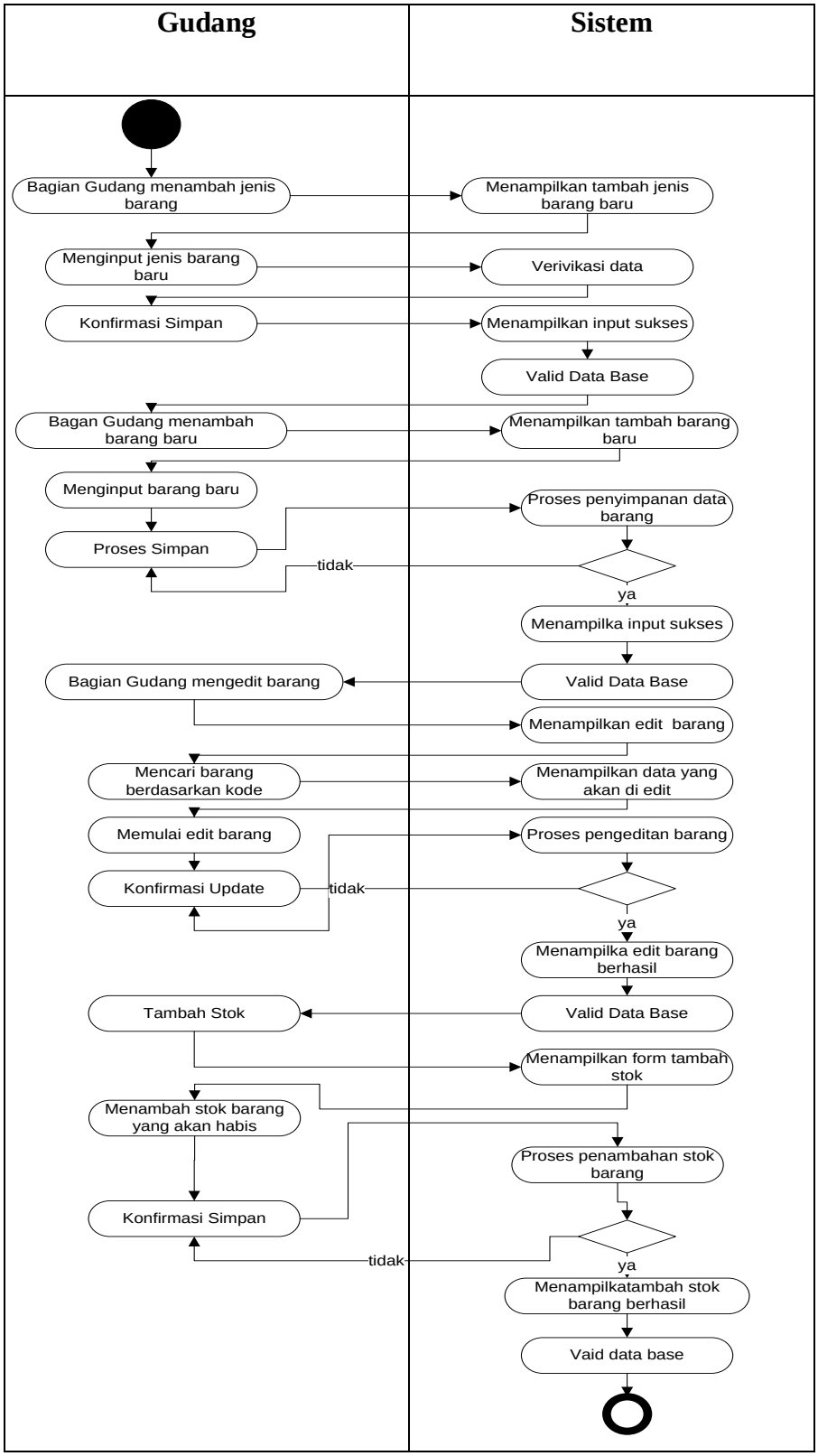
Gambar 4.5. *Activity diagram* login sistem yang diusulkan.

2. Activity diagram transaksi penjualan tanaman hias.



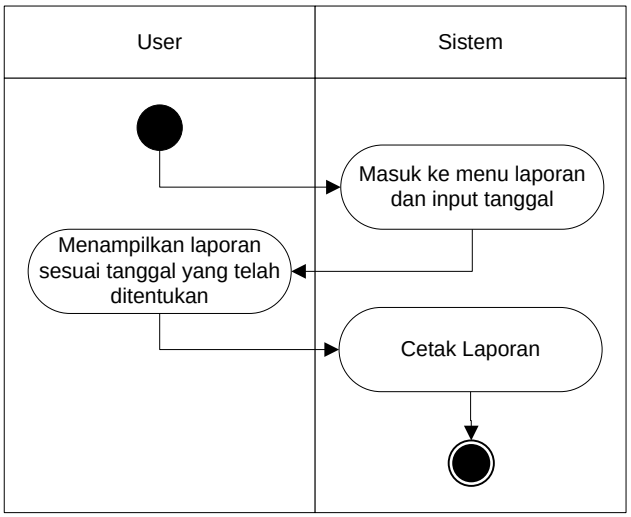
Gambar 4.6. *Activity diagram* transaksi penjualan yang diusulkan.

3. Activity diagram kelola data tanaman hias.



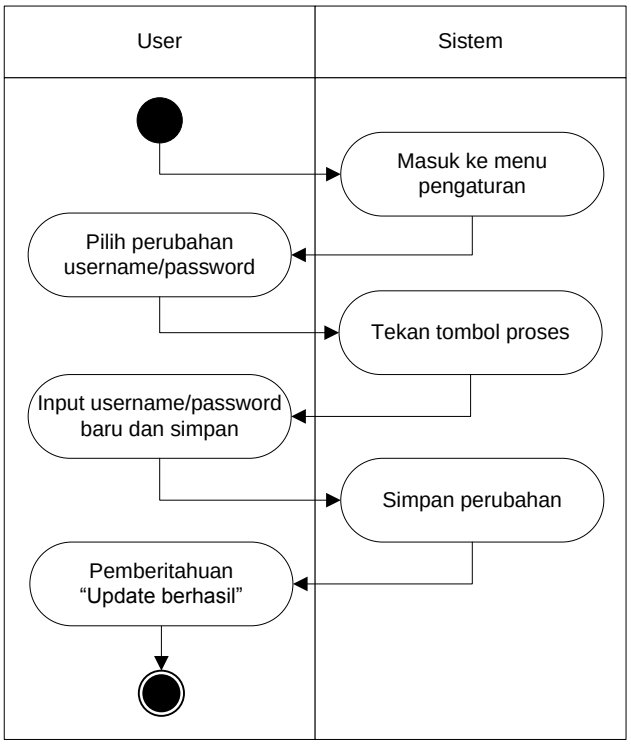
Gambar 4.7. *Activity diagram* kelola data tanaman.

4. Activity diagram laporan.



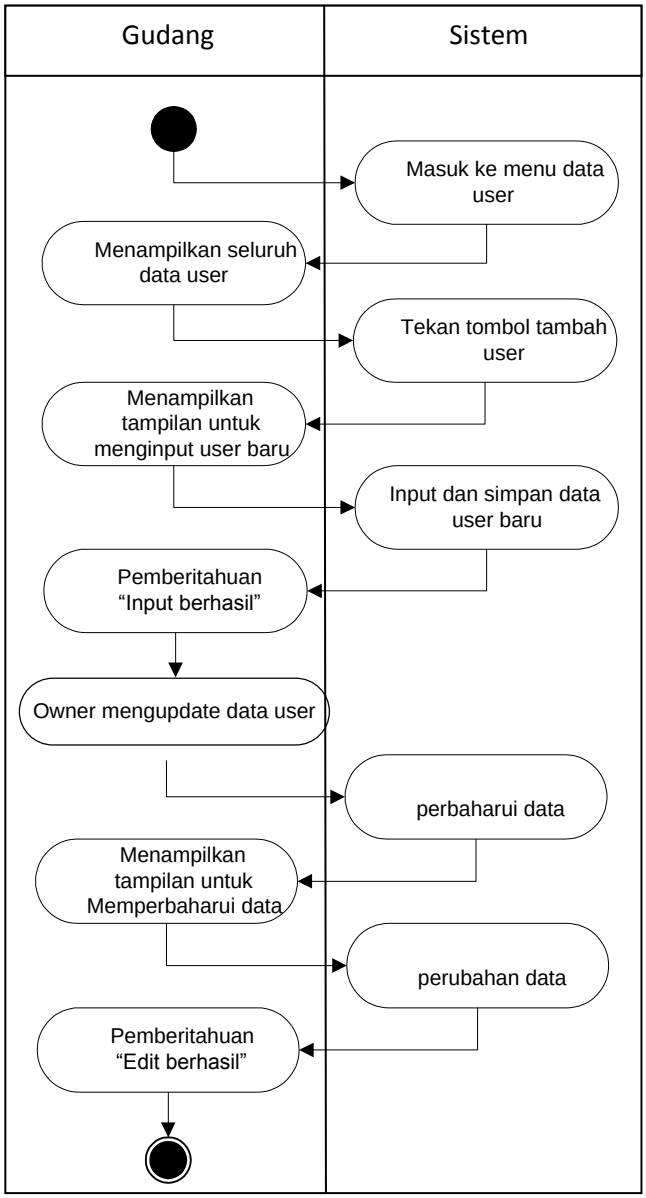
Gambar 4.8. *Activity diagram* laporan.

5. Activity diagram update username/password.



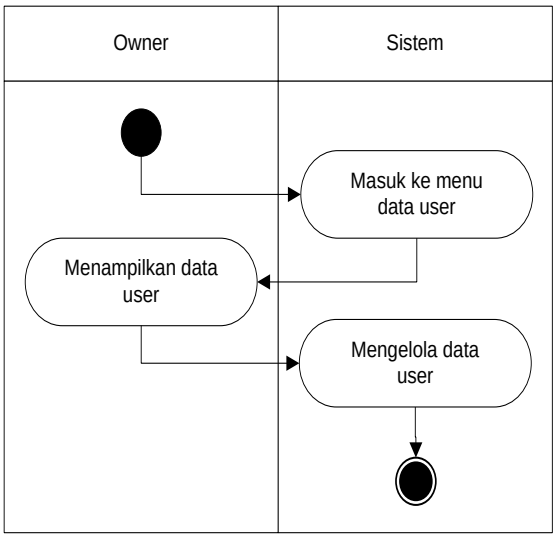
Gambar 4.9. *Activity diagram update username/password.*

6. Activity diagram kelola user.



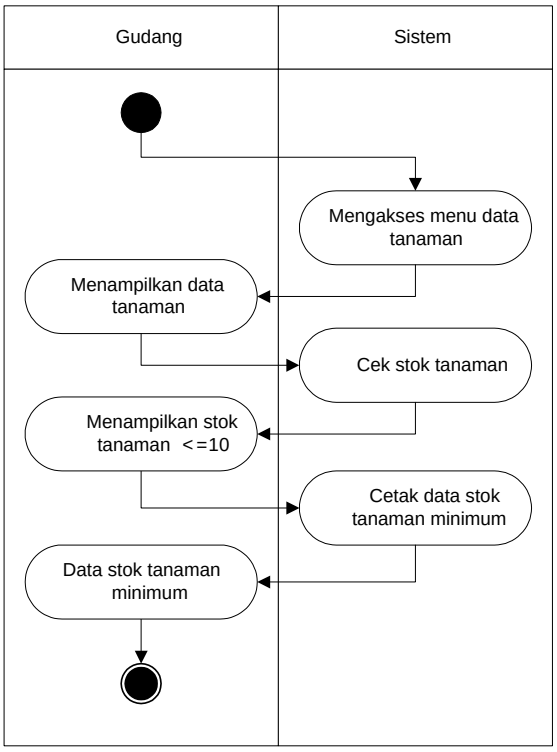
Gambar 4.10. *Activity diagram update* data *user.*

7. Activity diagram data user.



Gambar 4.11. *Activity diagram* data *user.*

8. Activity diagram data tanaman.

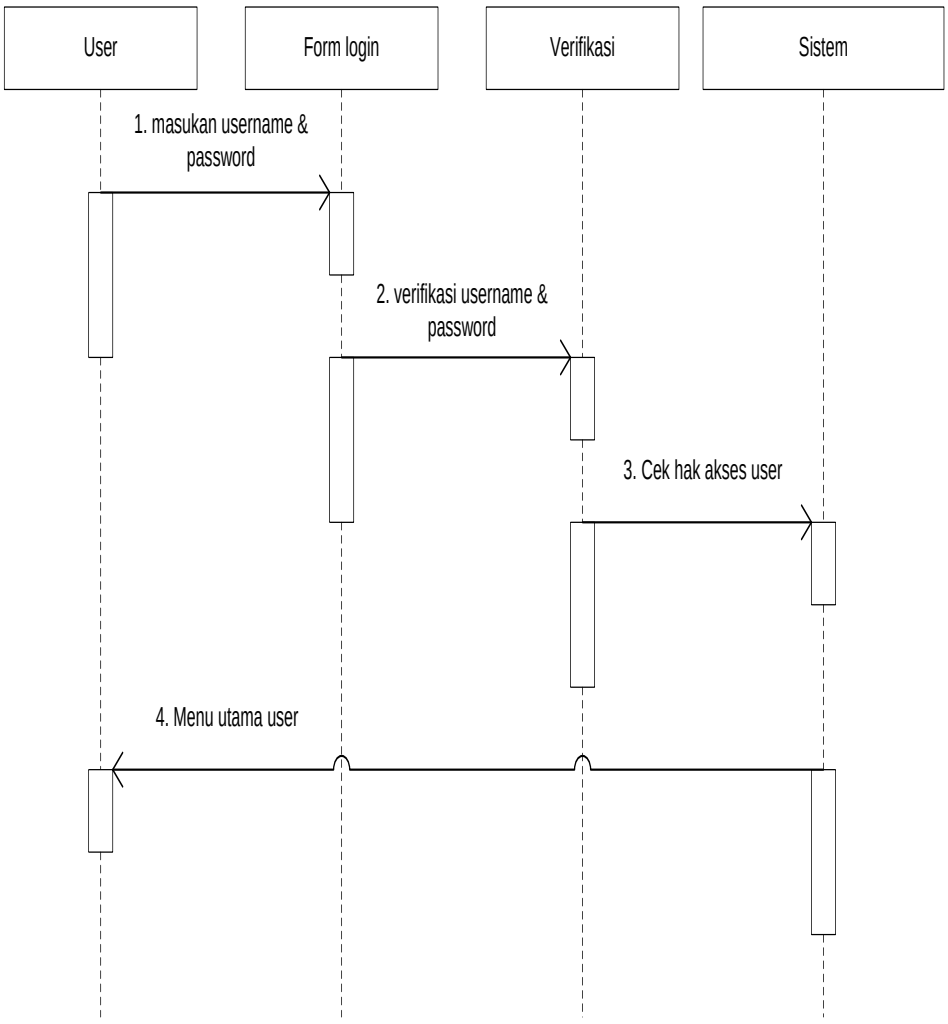


Gambar 4.12. *Activity diagram* data tanaman yang diusulkan.

4.3.3.3 Sequence Diagram

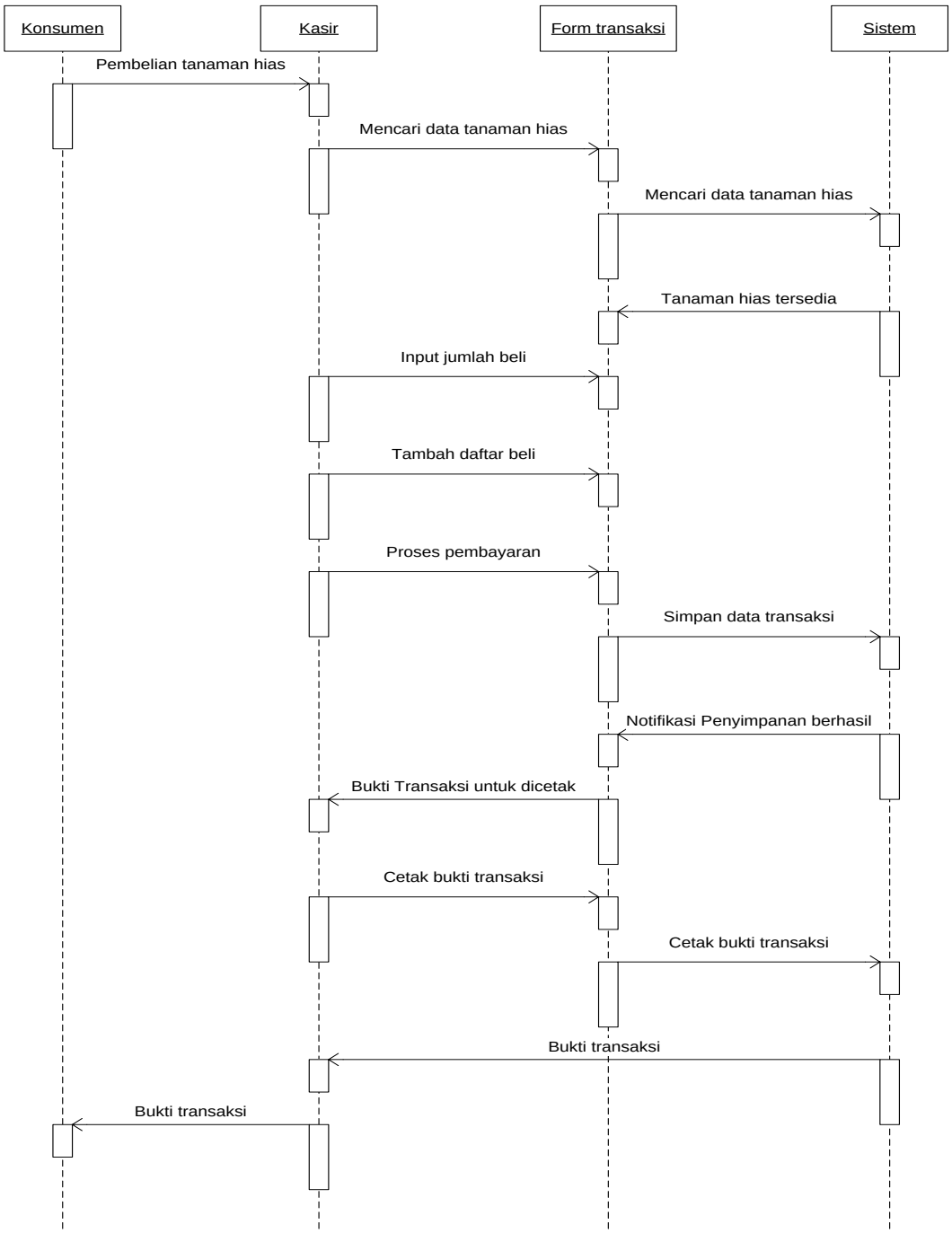
Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek-obyek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek-obyek yang melakukan suatu tugas atau aksi tertentu. Adapun *sequence diagram* sistem yang diusulkan adalah sebagai berikut :

1. Sequence diagram login.



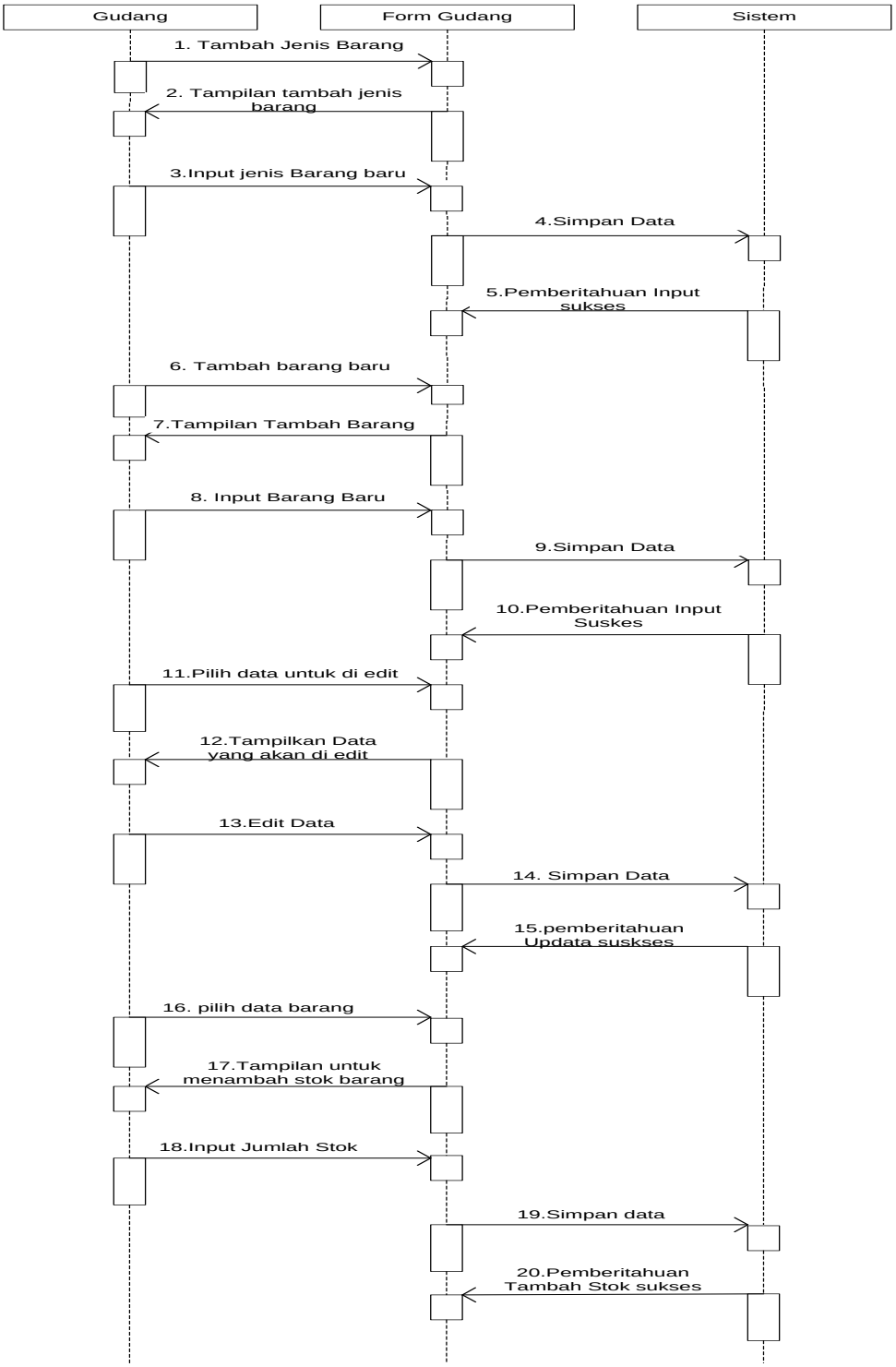
Gambar 4.13. *Sequence diagram* login.

2. Sequence diagram transaksi penjualan tanaman.



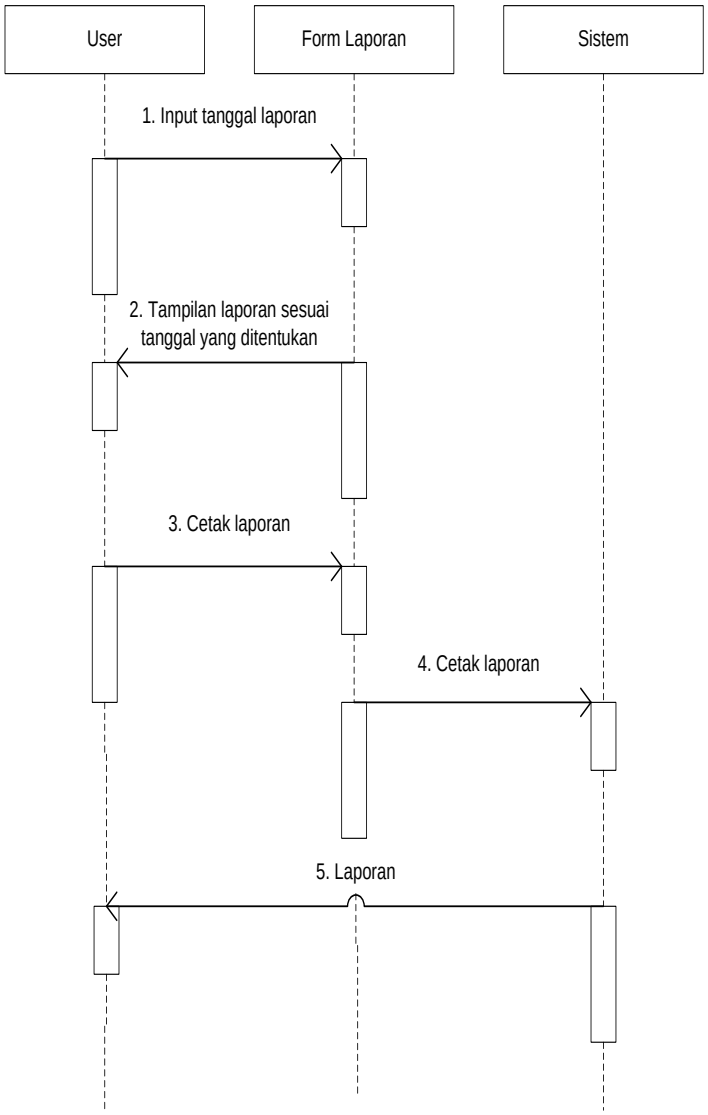
Gambar 4.14. *Sequence diagram* transaksasi penjualan tanaman.

3. Sequence diagram kelola data tanaman.



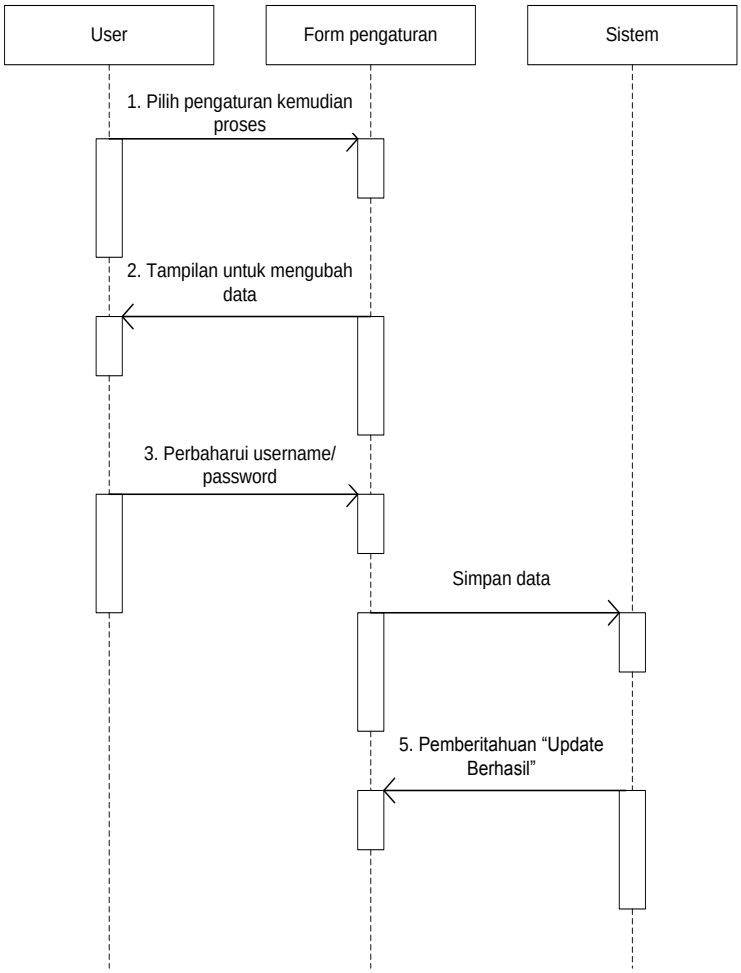
Gambar 4.15. *Sequence diagram* kelola data tanaman.

4. Sequence diagram laporan.



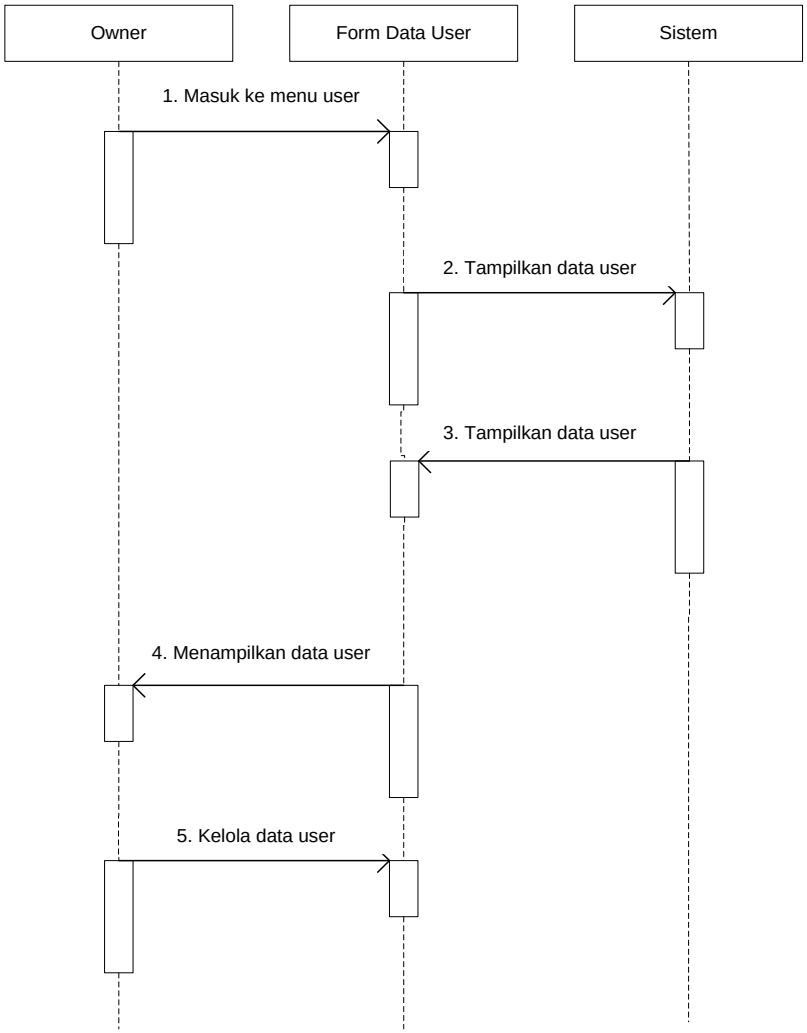
Gambar 4.16. *Sequence diagram* laporan.

5. Sequence diagram update username / password.



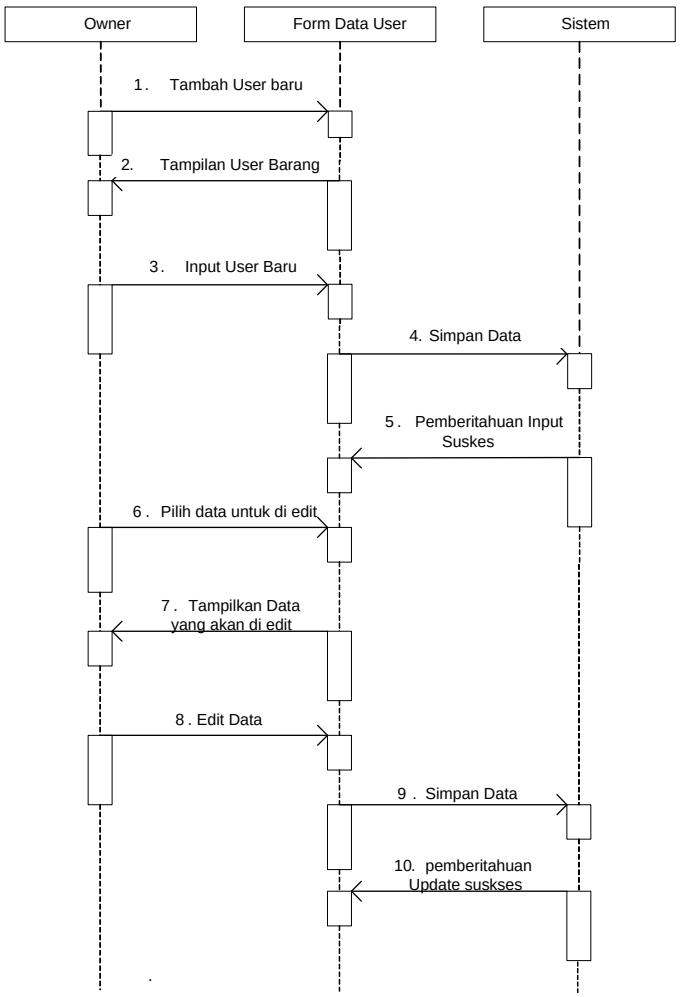
Gambar 4.17. *Sequence diagram update username / password.*

6. Sequence diagram data user.



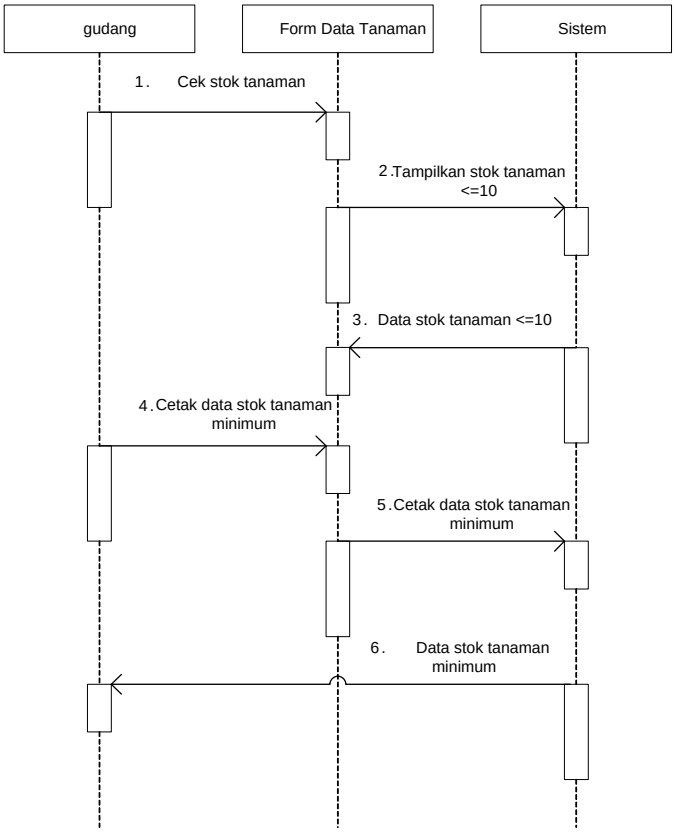
Gambar 4.18. *Sequence diagram* data user.

7. Sequence diagram kelola user.



Gambar 4.19. *Sequence diagram* kelola user.

8. Sequence diagram data tanaman.

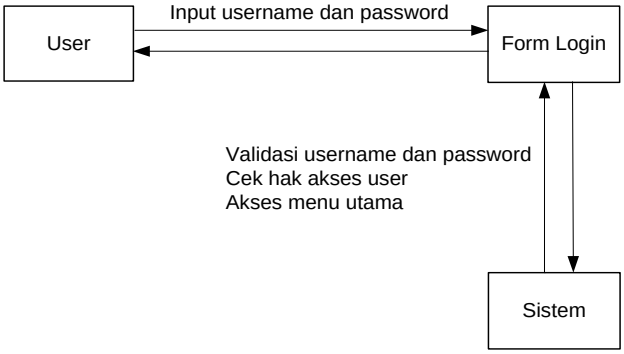


Gambar 4.20. *Sequence diagram* data tanaman.

4.3.3.4 Collaboration Diagram

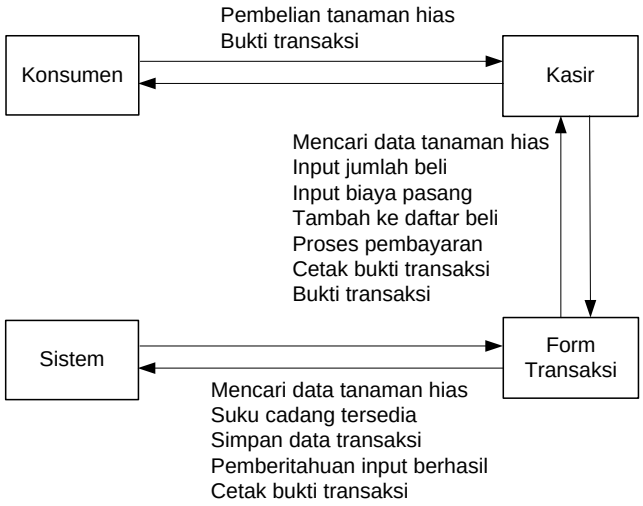
Collaboration diagram adalah suatu diagram yang memperlihatkan/ menampilkan pengorganisasian interaksi yang terdapat disekitar obyek dan hubungannya terhadap yang lain. *Collaboration diagram* lebih menekankan kepada peran setiap obyek dan bukan pada waktu penyampaian pesan.

1. Collaboration diagram login.



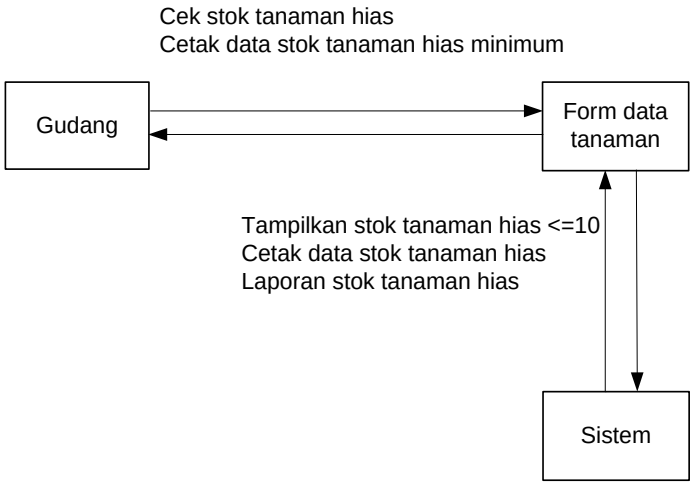
Gambar 4.21. *Collaboration diagram* login.

2. Collaboration diagram transaksi penjualan tanaman hias.



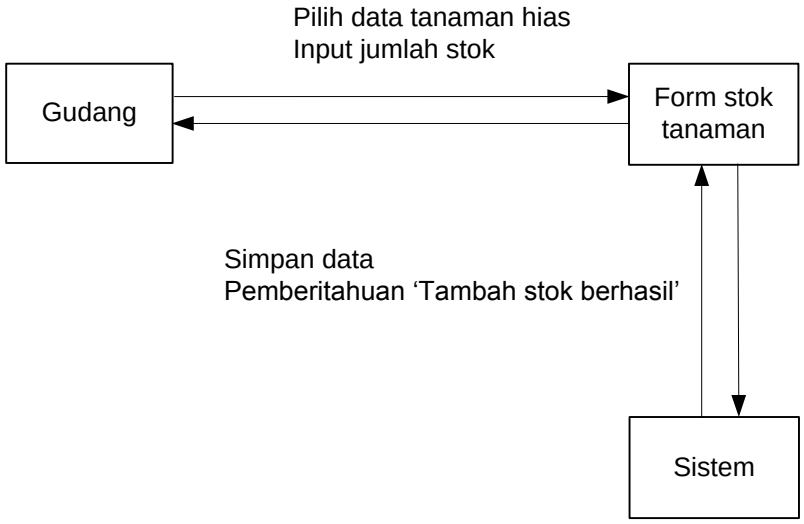
Gambar 4.22. *Collaboration diagram* transaksi penjualan tanaman hias.

3. Collaboration diagram data tanaman hias.



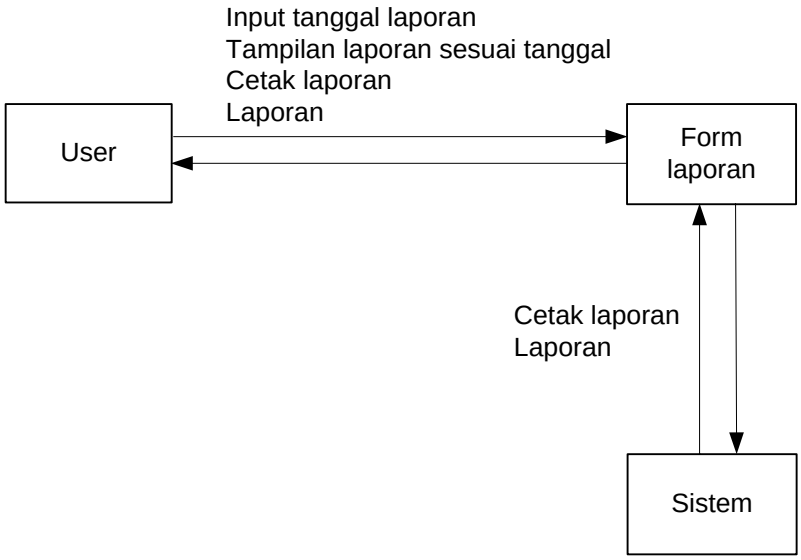
Gambar 4.23. *Collaboration diagram* data tanaman hias.

4. Collaboration diagram tambah stok.



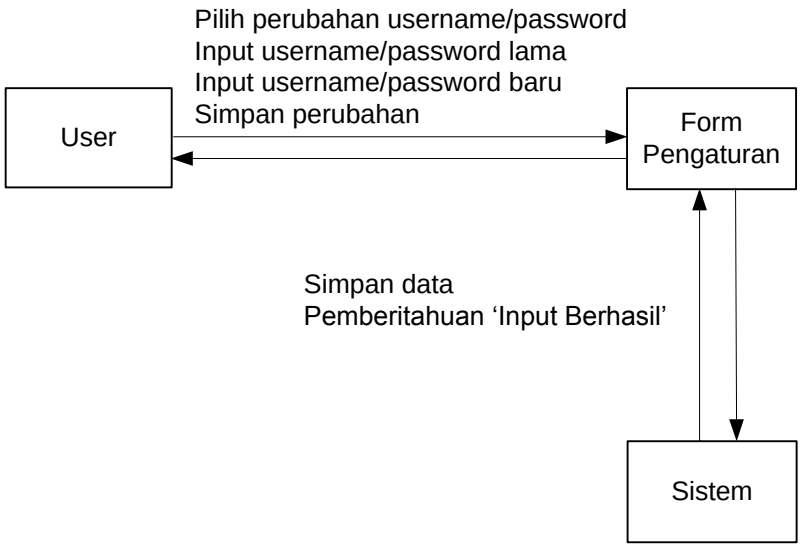
Gambar 4.24. *Collaboration diagram* tambah stok.

5. Collaboration diagram laporan.



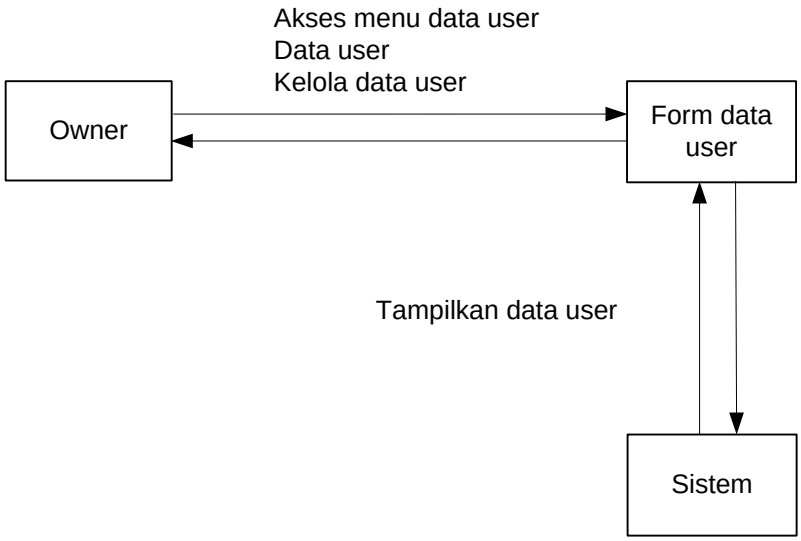
Gambar 4.25. *Collaboration diagram* laporan.

6. Collaboration diagram update username/password.



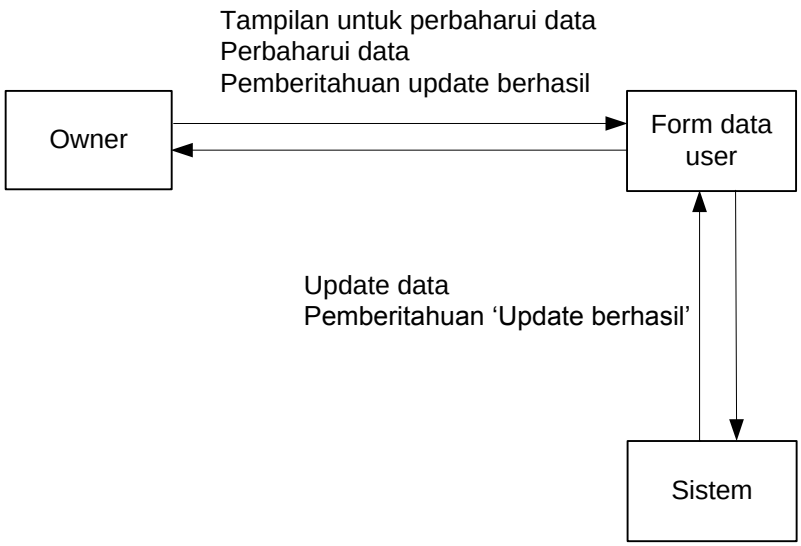
Gambar 4.26. *Collaboration diagram update username/password.*

7. Collaboration diagram data user.



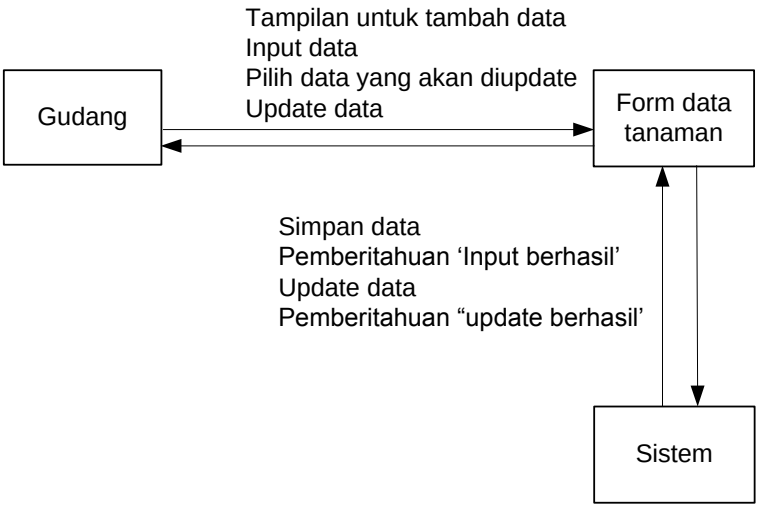
Gambar 4.27. *Collaboration diagram* data *user.*

8. Collaboration diagram kelola data user.



Gambar 4.28. *Collaboration diagram* kelola data *user.*

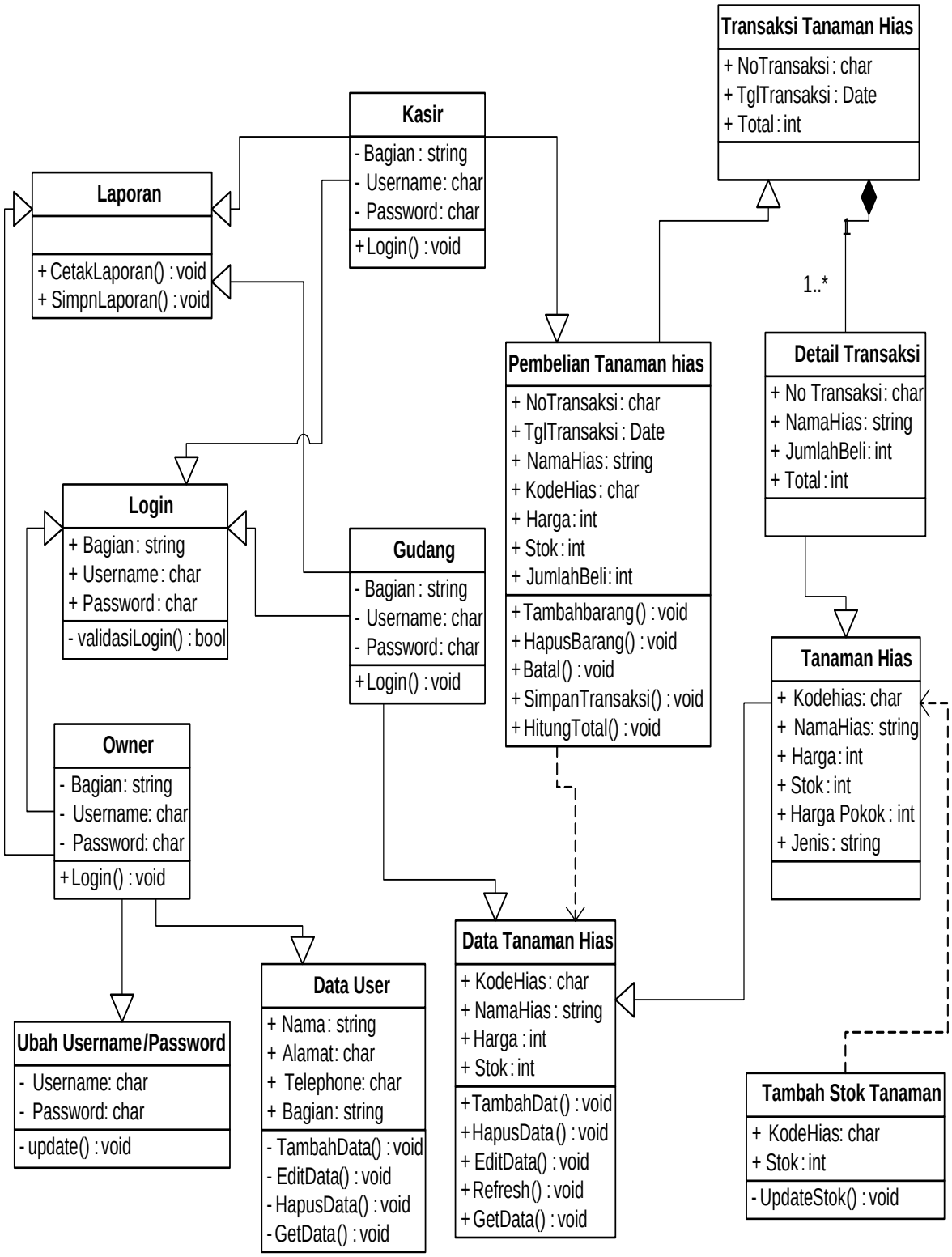
9. Collaboration diagram kelola data tanaman.



Gambar 4.29. *Collaboration diagram* kelola data tanaman.

4.3.3.5 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Adapun *class diagram* sistem yang diusulkan adalah sebagai berikut :



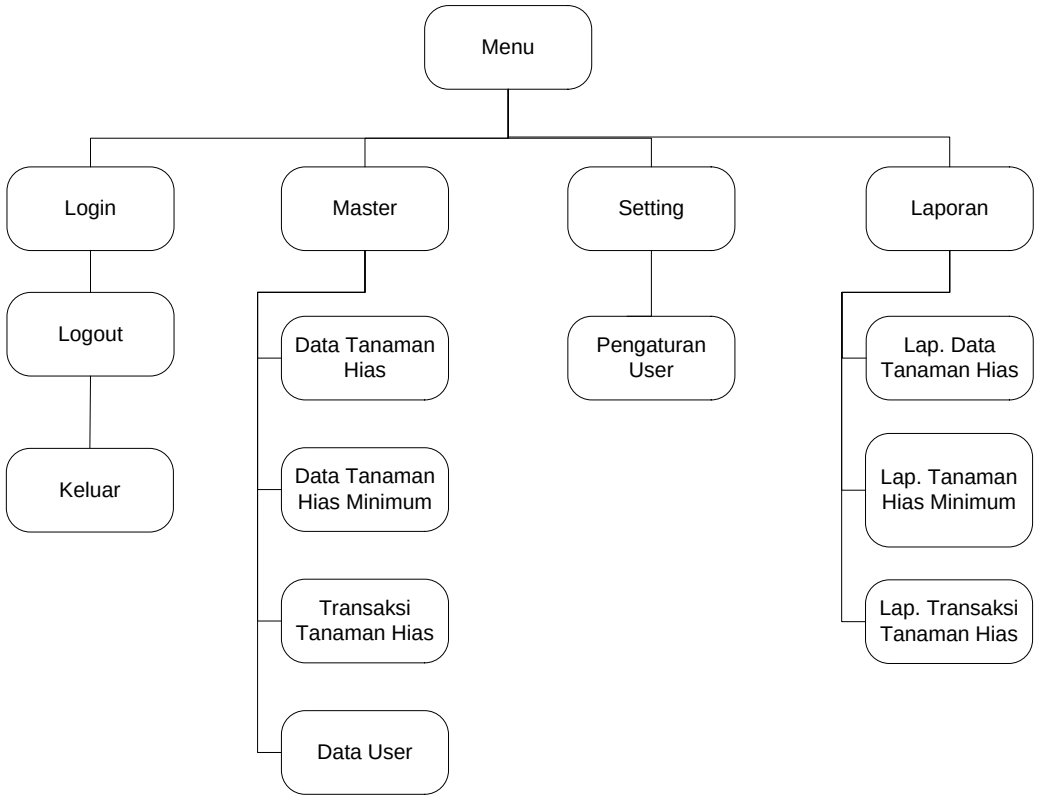
Gambar 4.30. *Class diagram* sistem informasi yang diusulkan.

4.3.4 Perancangan Antarmuka

Agar pengguna dan sistem bisa berinteraksi dengan baik, maka perlu dirancang *interface* yang dapat membantu pengguna dalam menggunakan sistem tersebut. Secara umum perancangan antar muka suatu program meliputi : struktur menu, perancangan *input* dan perancangan *output*.

4.3.4.1 Struktur Menu

Menu banyak digunakan dalam dialog karena merupakan jalur pemakai *(user interface)* yang mudah dipahami dan mudah digunakan. Menu berisi beberapa alternatif atau pilihan yang disajikan kepada *user*. Struktur menu dibawah ini menggambarkan hierarki dari sistem informasi penjualan tanaman hias yang dapat dilihat pada gambar dibawah ini :

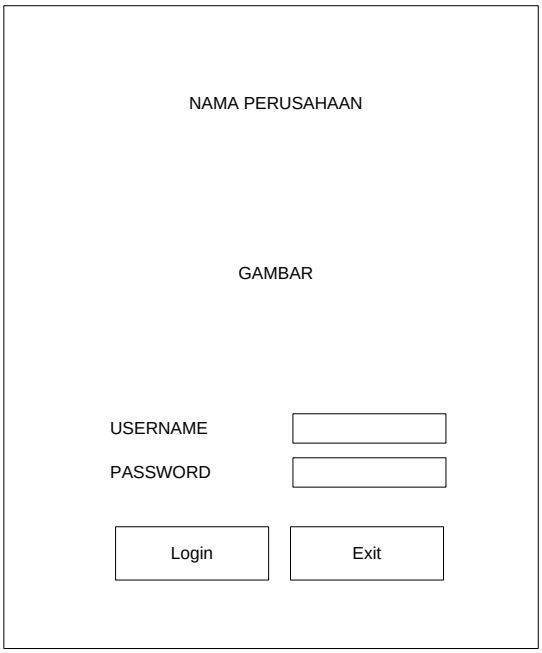


Gambar 4.31. Struktur menu.

4.3.4.2 Perancangan Input

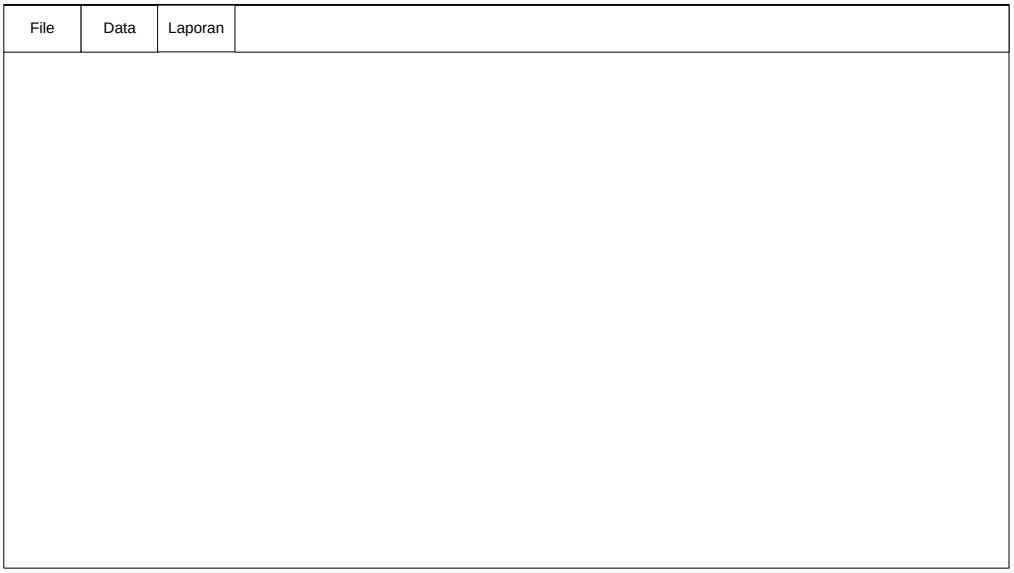
Perancangan *input* merupakan desain yang dirancang untuk menerima perintah dan masukan dari pengguna. Rancangan *input* ini harus dapat memberikan penjelasan/petunjuk bagi pemakainya, baik dari bentuk maupun dari masukan-masukan yang akan diisi.

1. Tampilan Login.



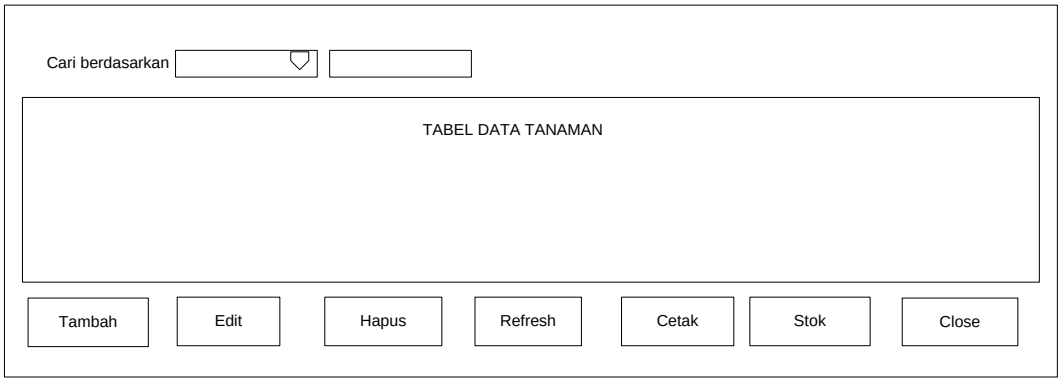
Gambar 4.32. *Login user*.

2. Tampilan Menu Utama.



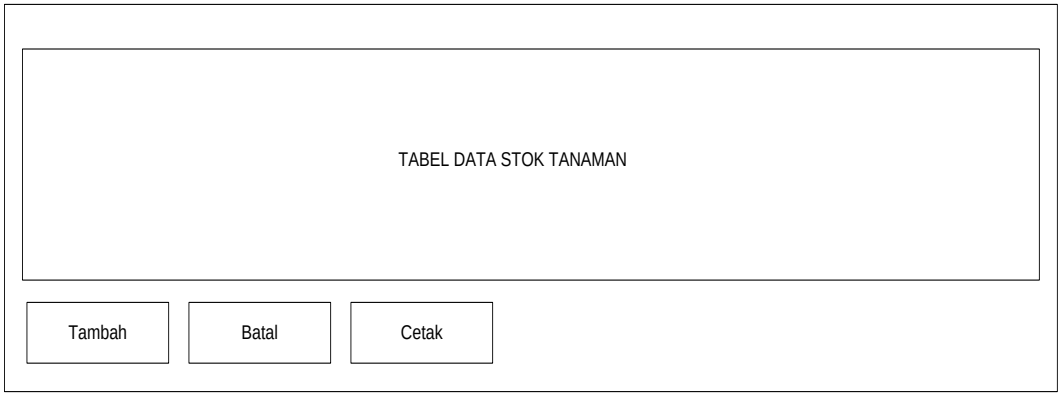
Gambar 4.33. Menu Utama.

3. Tampilan Form Data Tanaman.



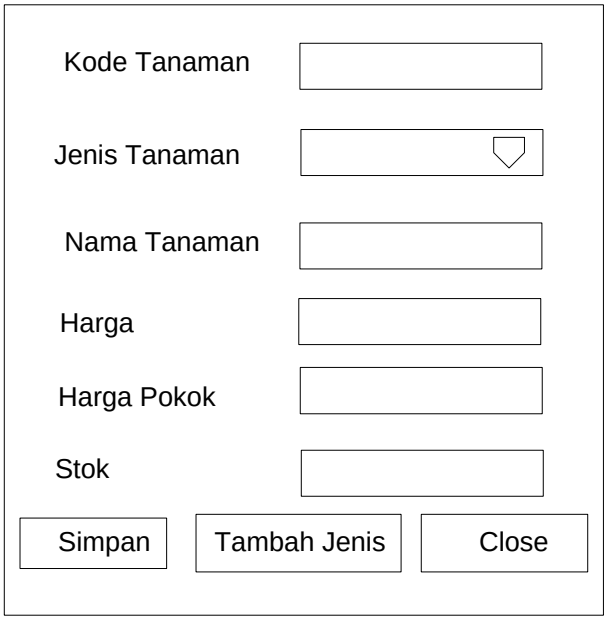
Gambar 4.34. *Form* data tanaman.

4. Tampilan Form Stok Minimum.



Gambar 4.35. *Form* stok minimum

5. Tampilan Form Tambah Tanaman.



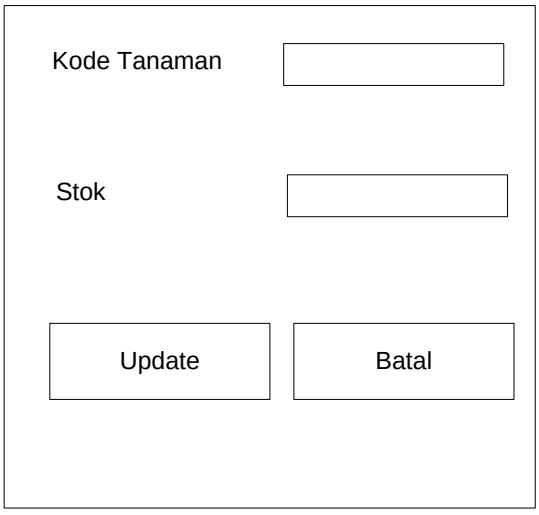
Gambar 4.36. *Form* tambah Tanaman

6. Tampilan Form Update Tanaman.



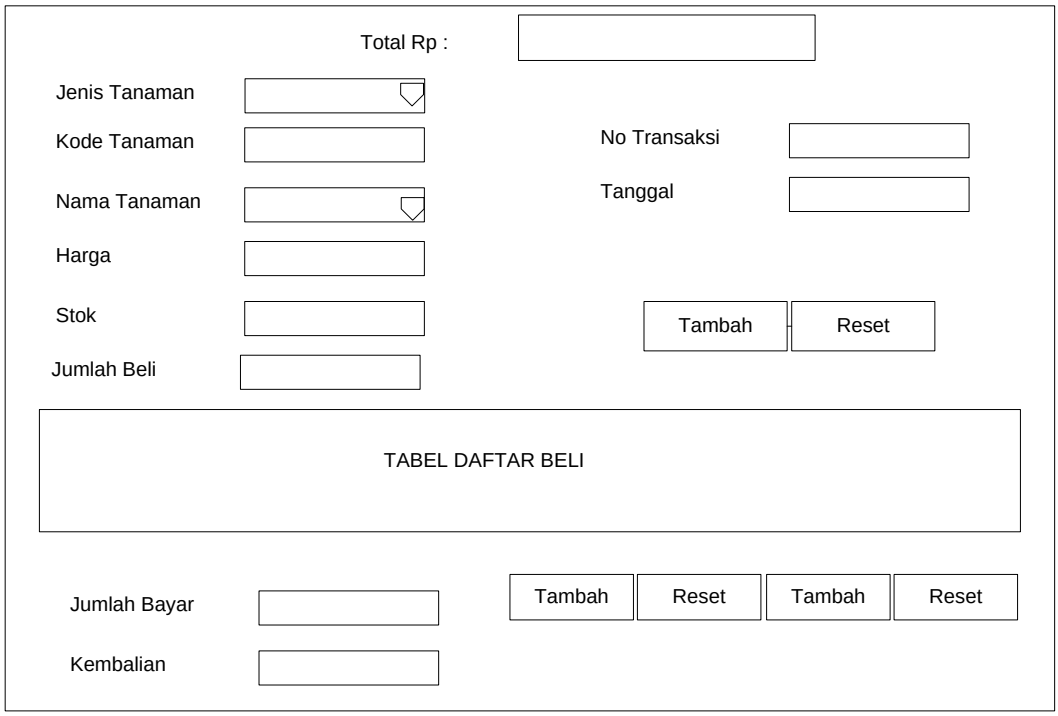
Gambar 4.37. *Form update* Tanaman.

7. Tampilan Form Tambah Stok Tanaman.



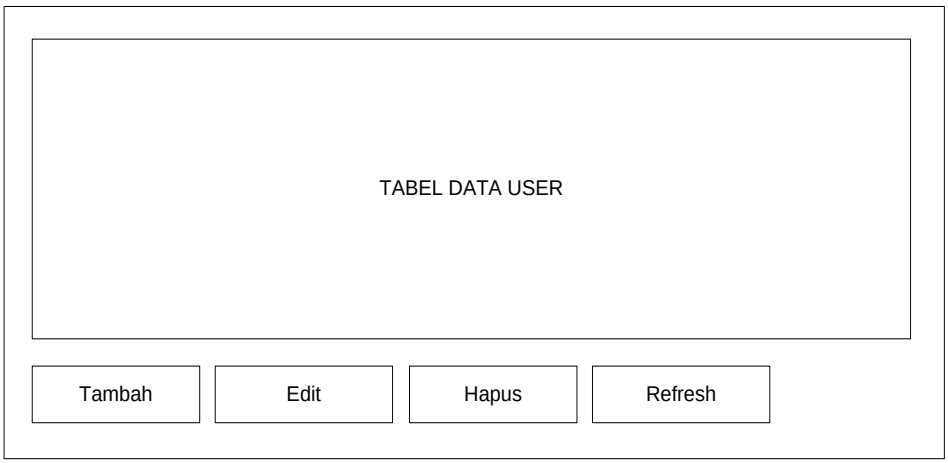
Gambar 4.38. *Form* tambah stok Tanaman.

8. Tampilan Form Transaksi Penjualan Tanaman.



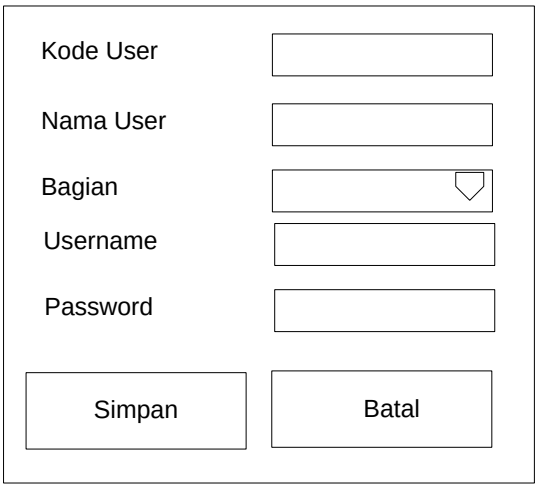
Gambar 4.39. *Form* transaksi penjualan tanaman.

9. Tampilan Form Data User.



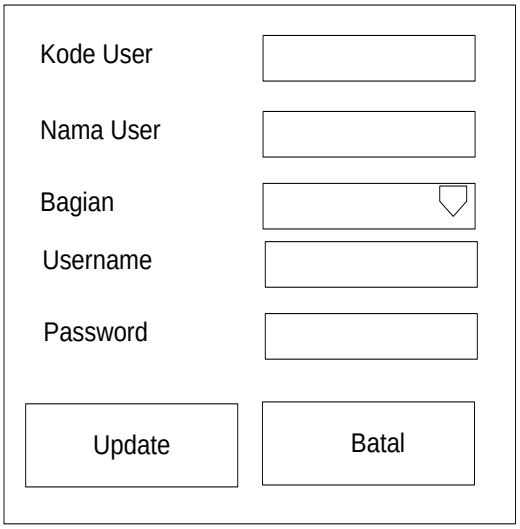
Gambar 4.40. *Form* data *User*.

10. Tampilan Form Tambah User.



Gambar 4.41. *Form* tambah data *user*.

11. Tampilan Form Update Data User.

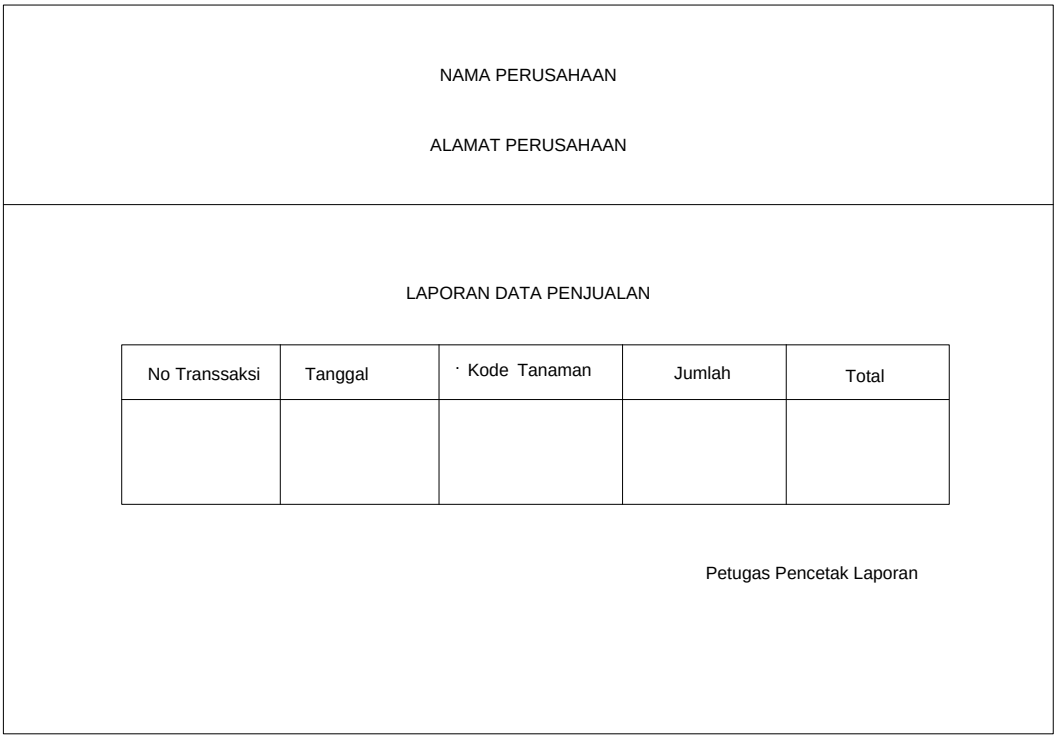


Gambar 4.42. *Form update* data *user*.

4.3.4.3 Perancangan Output

Perancangan output merupakan bentuk tampilan keluaran berupa laporan-laporan hasil transaksi.

1. Laporan Penjualan Tanaman.


NAMA PERUSAHAAN

ALAMAT PERUSAHAAN

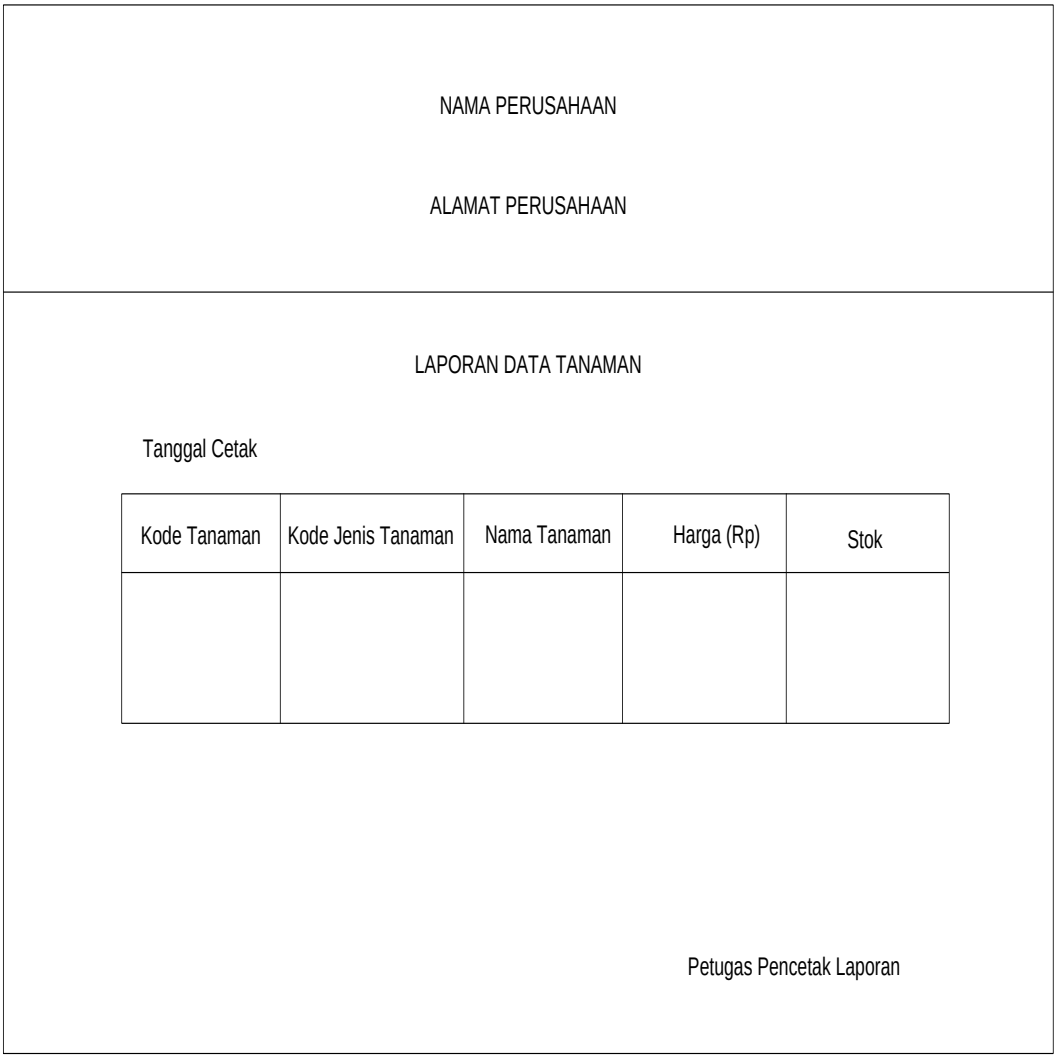
LAPORAN DATA PENJUALAN

No Transsaksi	Tanggal	· Kode Tanaman	Jumlah	Total

Petugas Pencetak Laporan

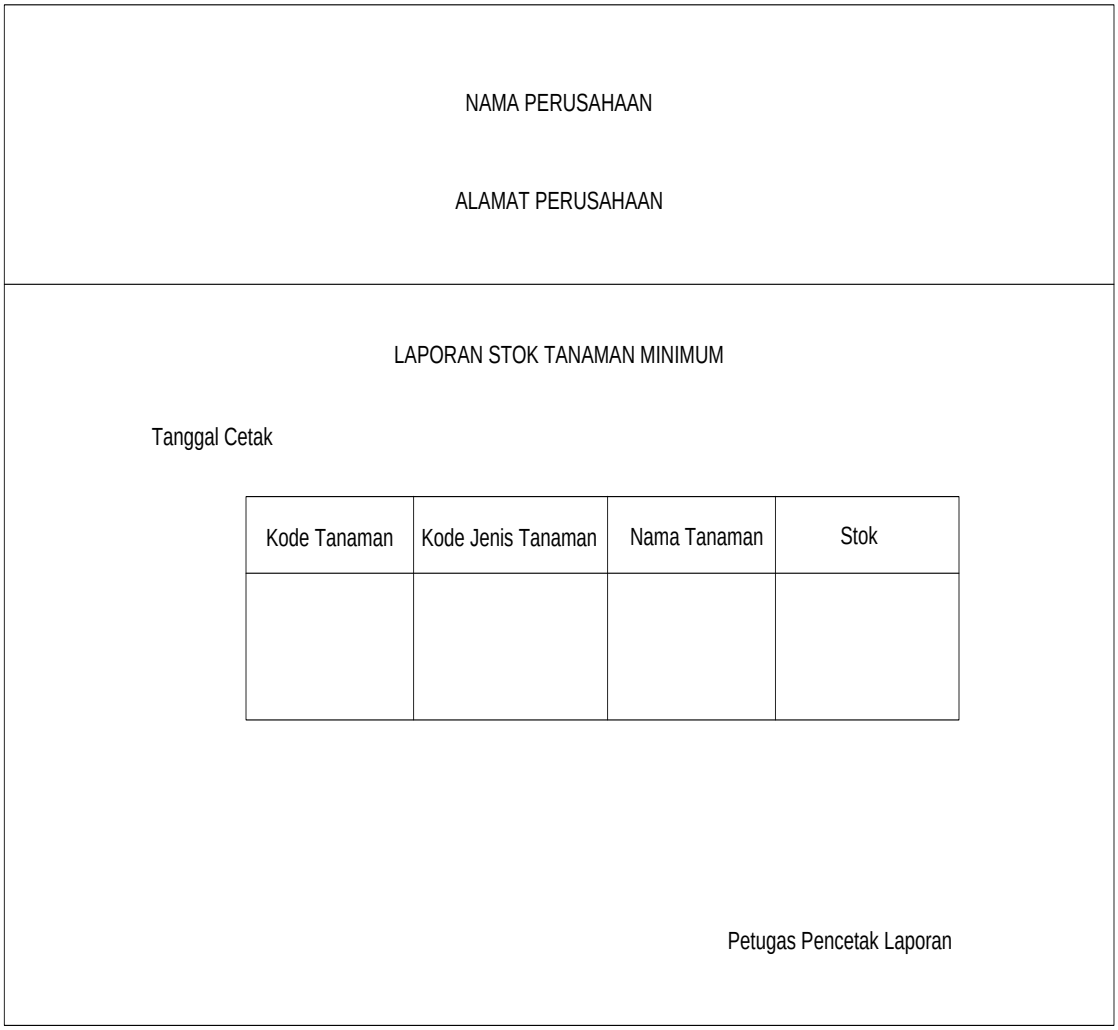
Gambar 4.43. Laporan penjualan tanaman.

2. Laporan Data Tanaman.



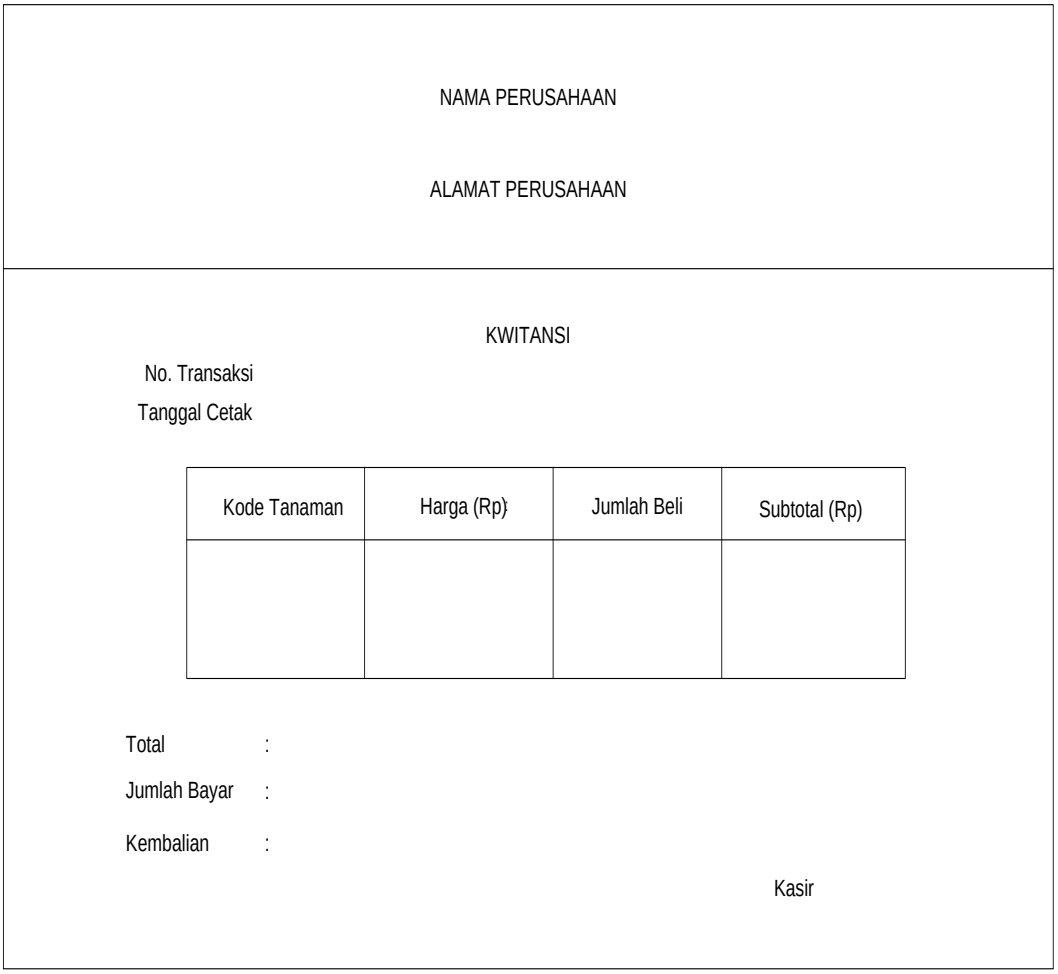
Gambar 4.44. Laporan data tanaman.

3. Laporan Stok Tanaman Minimum.



Gambar 4.45. Laporan stok tanaman minimum.

4. Faktur Transaksi Tanaman.

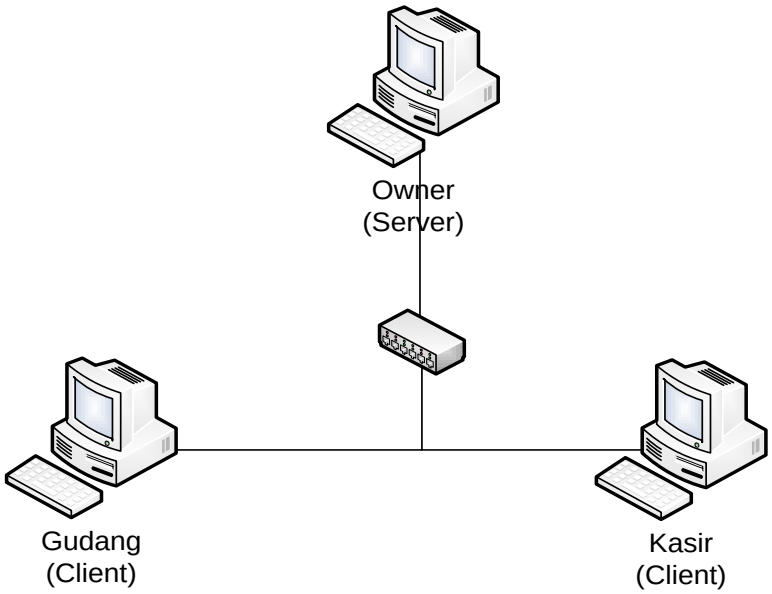


Gambar 4.46. Faktur transaksi tanaman.

4.3.5 Perancangan Jaringan

Topologi jaringan adalah bentuk hubungan dari suatu jaringan. Jenis topologi jaringan yang dipakai adalah *Point To Point* (titik ke titik).

Jaringan kerja titik ke titik merupakan jaringan kerja yang paling sederhana tetapi dapat digunakan secara luas. Begitu sederhananya jaringan ini, sehingga seringkali tidak dianggap sebagai suatu jaringan tetapi hanya merupakan komunikasi biasa. Dalam hal ini, kedua simpul mempunyai kedudukan yang setingkat, sehingga simpul manapun dapat memulai dan mengendalikan hubungan dalam jaringan tersebut. Data dikirim dari satu simpul langsung kesimpul lainnya sebagai penerima, misalnya antara terminal dengan CPU.



Gambar 4.45. Arsitektur Jaringan.