BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem Yang Sedang Berjalan

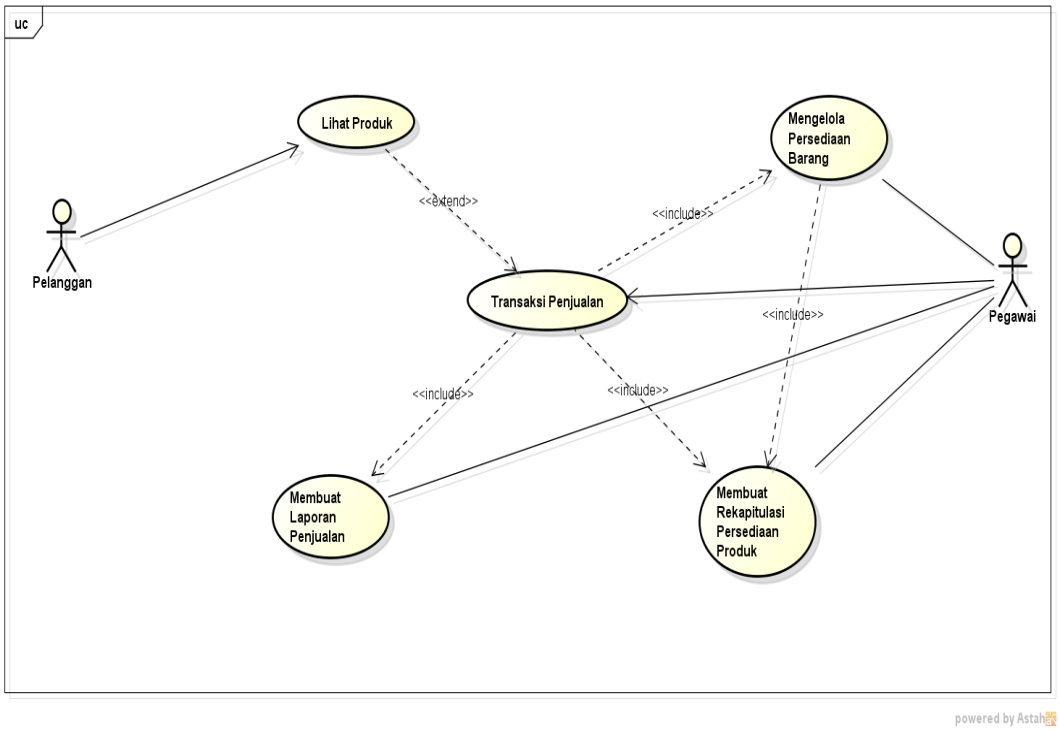
Analisis terhadap sistem yang berjalan dimaksudkan untuk mempelajari secara seksama terhadap suatu sistem yang sedang dijalankan oleh suatu organisasi, sehingga mampu memperkirakan dan merincikan seluruh dokumen ataupun prosedur yang terlibat dan membuat rekomendasi untuk organisasi dengan memperhitungkan segi penyelesaian yang akan dicapai.

4.1.1 Analisis Prosedur Yang Sedang Berjalan

Analisis terhadap prosedur-prosedur yang sedang berjalan hendaknya perlu dilakukan terlebih dahulu sebelum dilakukannya perancangan terhadap sistem yang akan diusulkan. Dan harus pula dilakukan analisis terhadap hal-hal yang menjadi tujuan pemakai sehingga masalah tersebut dapat didefinisikan secara jelas.

4.1.1.1 Use Case Diagram

Use case diagram menggambarkan siapa saja aktor yang melakukan prosedur dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi pada sistem tersebut. Adapun *use case diagram* yang berjalan saat ini pada Distro Celtic yaitu :



Gambar 4.1 *Use Case* Diagram Sistem Informasi Penjualan pada Dsitro Celtic yang Sedang Berjalan.

4.1.1.2 Skenario Use Case

Peran skenario *use case* yaitu guna memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase berikutnya dengan melakukan penilaian terhadap skenario tersebut. Adapun tahapan-tahapan sekenario *use case* yang sedang berjalan pada Distro Celtic Bandung adalah sebagai berikut :

1. Nama *Use Case* : Lihat produk

   Aktor : Pelanggan

   Tujuan : Melihat produk

Tabel 4.1 Skenario *Use Case* Melihat Produk.

Pelanggan	Sistem
1. Berkunjung, melihat dan memilih produk yang tersedia	
	2. Menampilkan atau Memajang produk yang tersedia

2. Nama *Use Case* : Transaksi penjualan Produk

   Aktor : Pelanggan, Pegawai

   Tujuan : Melakukan transaksi penjualan produk

Tabel 4.2 Skenario *Use Case* Transaksi Penjualan produk.

Pelanggan	Pegawai
1. Melakukan pemesanan produk	
	2. Memeriksa atau memberikan informasi pesedian dan harga produk
3. Menerima informasi tentang persediaan dan harga produk	
	4. Menyimpan informasi dari proses transaksi penjualan produk
	5. Membuat dan Mencetak struk penjualan
6. Menerima struk penjualan	

3. Nama *Use Case* : Mengelola persedian produk

   Aktor : pegawai

   Tujuan : Memanipulasi persedian produk

Tabel 4.3 Tabel Skenario *use case* mengelola persedian produk.

Pegawai	Sistem
1. Melakukan manipulasi persedian produk.	
	2. Menyimpan informasi perubahan persedian produk.

4. Nama *Use Case* : Membuat rekapitulasi persedian produk

   Aktor : Pegawai

   Tujuan : Membuat dan mencetak rekapitulasi produk

Tabel 4.4 Tabel Skenario *Use Case* membuat rekapitulasi produk.

Karyawan	Sistem
1. Menampilkan data-data persediaan barang.	
	2. Membuat dan mencetak rekapitulasi persedian produk.

5. Nama *Use Case* : Membuat rekapitulasi transaksi penjualan produk

   Aktor : Pegawai

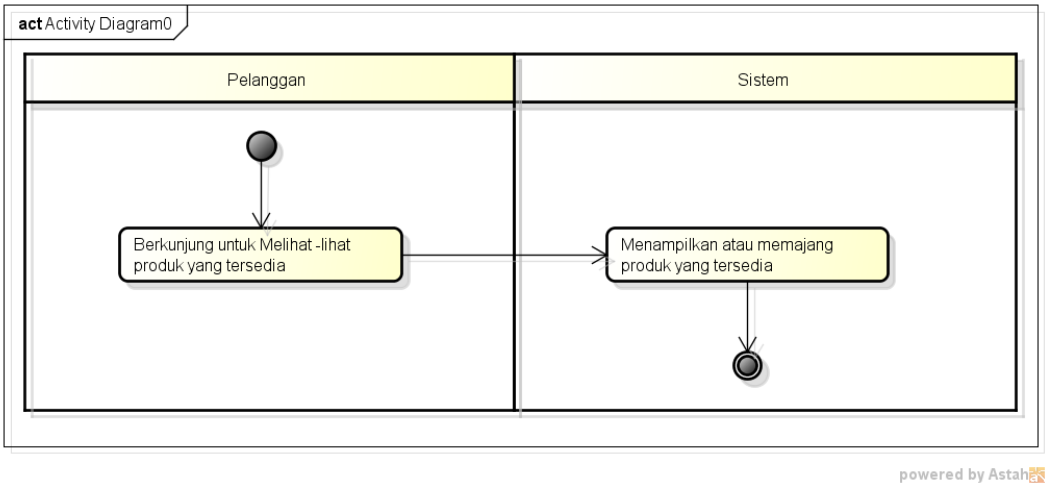
   Tujuan : Membuat dan mencetak rekapitulasi transaksi penjualan produk.

Tabel 4.5 Tabel Skenario *Use Case* membuat rekapitulasi transaksi penjualan produk.

Pegawai	Sistem
1. Menampilkan informasi-informasi transaksi penjualan produk.	
	2. Membuat dan mencetak rekapitulasi transaksi penjualan produk.

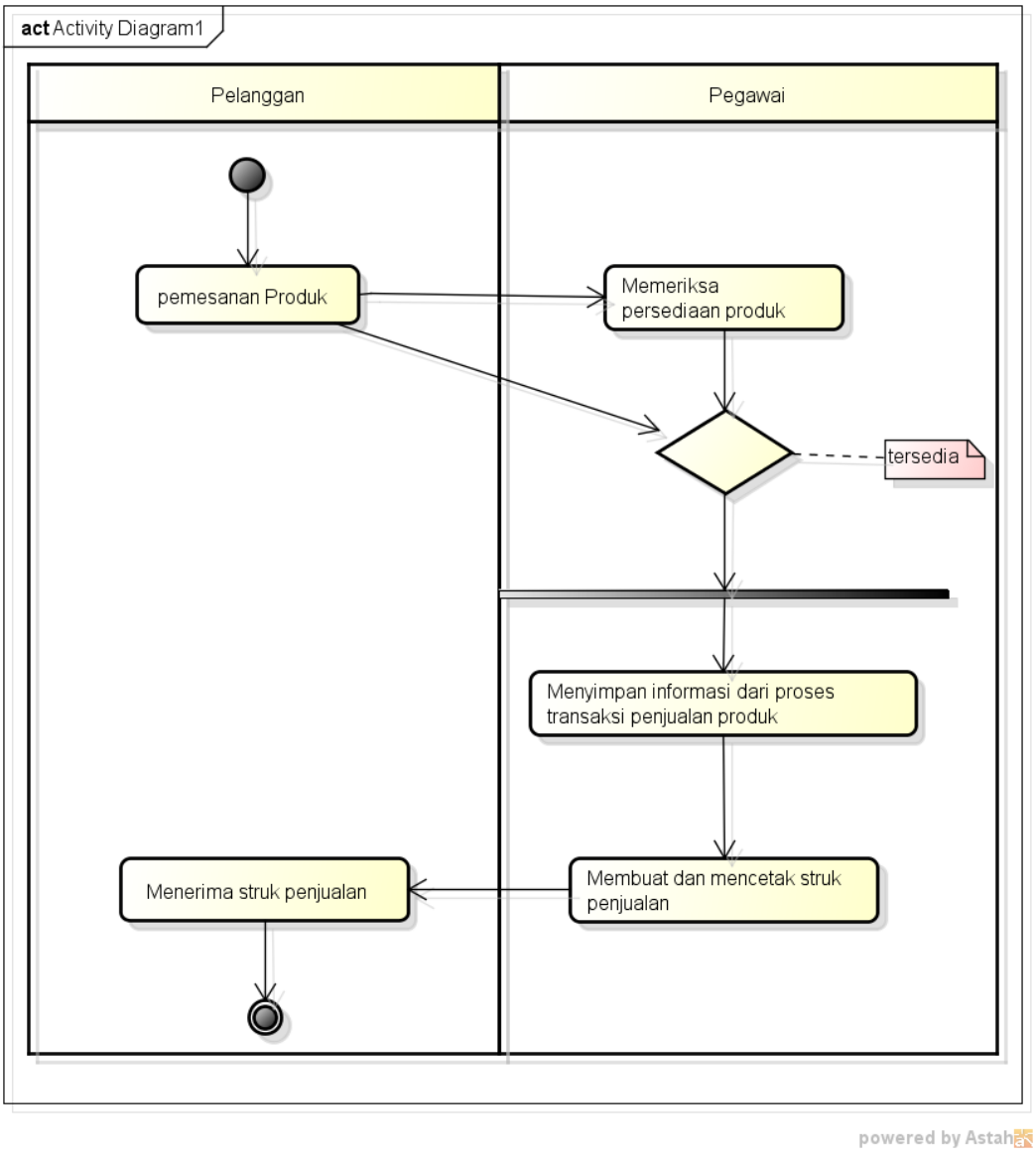
4.1.1.3 Activity Diagram

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam suatu sistem. Agar dapat lebih memahami tentang sistem yang akan dibuat, maka perlu dibuatkan *activity diagram t*entang sistem yang sedang berjalan, yaitu:



Gambar 4.2. *Activity Diagram* Melihat produk yang sedang berjalan pada Distro Celtic bandung.

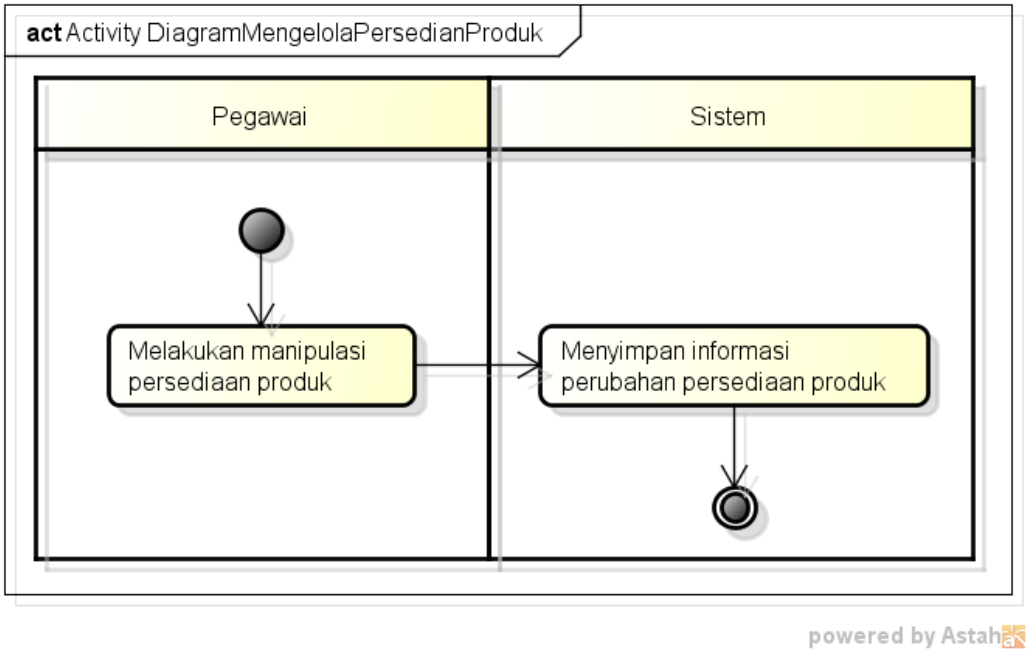
Pada *activity diagram* lihat produk yang sedang berjalan ini, menggambarkan dimana pelanggan datang langsung ke distro untuk melihat-lihat produk yang tersedia dan terpajang di dalam distro Celtic Bandung.



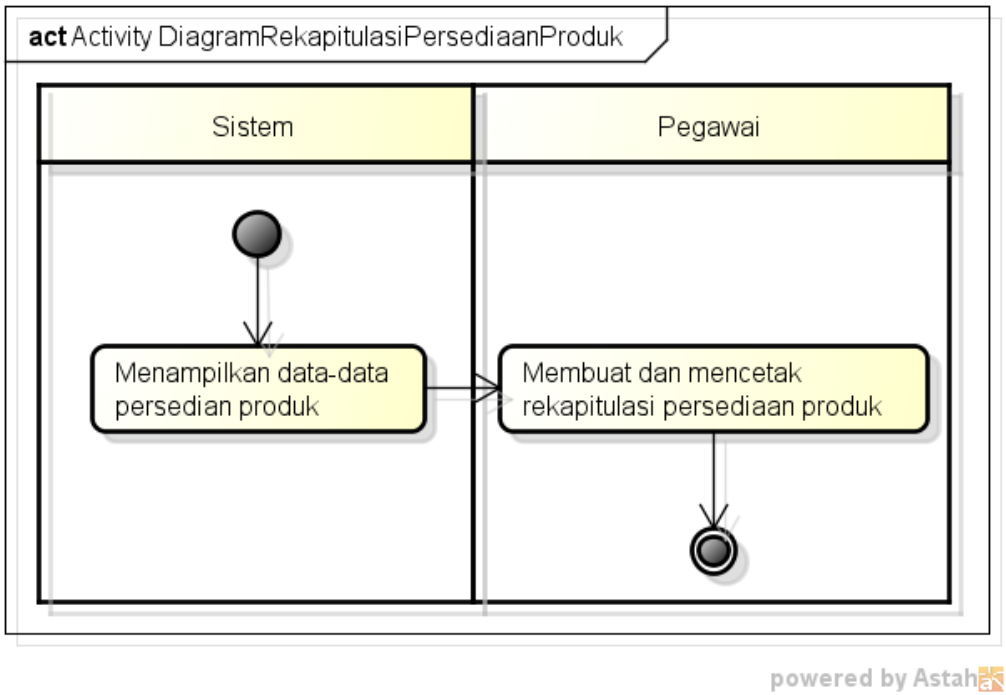
Gambar 4.3. *Activity Diagram* Transaksi Penjualan produk yang sedang berjalan pada distro Celtic Bandung.

Dalam *activity diagram* transaksi penjualan produk yang sedang berjalan ini, menggambarkan bagaimana pelanggan melakukan pemesanan suatu produk

dan pegawai distro memeriksa persedian produk yang dipesan. Apabila produk tersedia maka transaksi dilanjutkan dan pegawai menyimpan semua informasi dari proses transaksi penjualan lalu membuat dan mencetak struk untuk kemudian diserahkan kepada pelanggan.

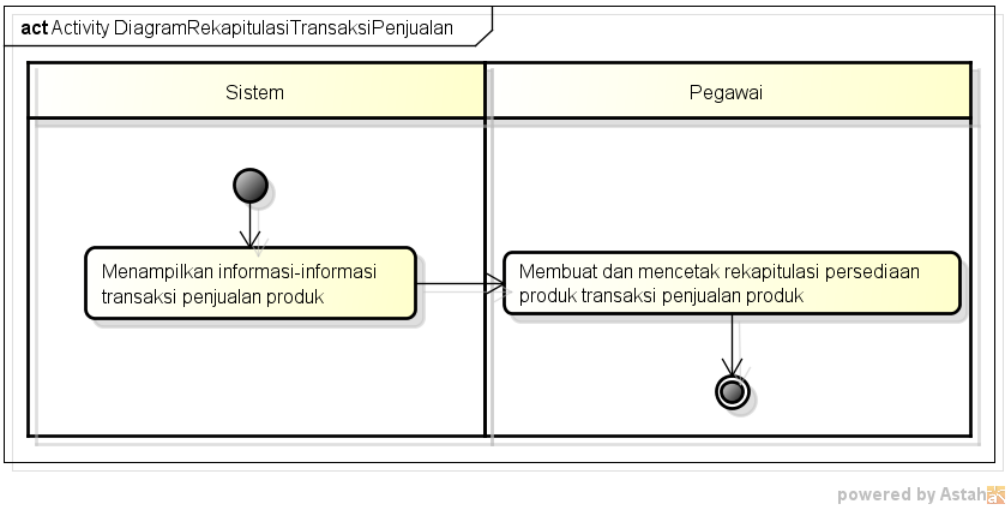


Gambar 4.4. *Activity Diagram* Mengelola persedian produk yang sedang berjalan pada distro Celtic Bandung



Gambar 4.5. *Activity Diagram* rekapitulasi persediaan produk yang sedang berjalan pada distro Celtic Bandung.

Pada *activity diagram* ini, sistem menampilkan data-data persediaan produk dan pegawai membuat dan mencetak laporan persediaan produk.



Gambar 4.6. *Activity Diagram* Rekapitulasi transaksi penjualan produk yang Sedang Berjalan pada distro Celtic Bandung.

Dimana sistem menampilkan informasi-informasi transaksi penjualan produk lalu pegawai membuat dan mencetak laporan transaksi penjualan produk.

4.1.2 Evaluasi Sistem Yang Sedang Berjalan

Evaluasi terhadap sistem yang sedang berjalan dimaksudkan guna memperoleh solusi terbaik terhadap perubahan sistem yang lebih baik, dan evaluasi ini dilakukan setelah tahap proses analisis terhadap sistem yang sedang berjalan. Baik analisis terhadap prosedur yang ada, juga beberapa permasalahan yang penulis temukan baik itu secara langsung maupun tidak langsung. Sehingga dapat berdampak terhadap kinerja sistem yang ada. Dan bila masalah-masalah itu ditinjau lebih jauh lagi akan berpengaruh terhadap pengembangan sistem nantinya. Berdasarkan analisa sistem dan hasil wawancara yang telah dilakukan, maka dapat disimpulkan bahwa :

Tabel 4.6 Evaluasi Sistem yang Sedang Berjalan

No	Permasalahan	Bagian	Rancangan Permasalahan
1	Pengelolaan data produk yang kurang efektif dan efesien.	Pegawai	Membangun system informasi yang dapat membantu dalam melakukan Pengelolaan data produk secara efektif dan efesien.
2	Sistem penjualan yang mengharuskan pelanggan datang langsung ke distro apabila ingin melakukan pemesanan.	Pelanggan	Membangun system penjualan yang dapat memudahkan pelanggan dalam melakukan pemesanan

3	Lama proses pembuatan rekapitulasi	Pegawai	Membuat sistem yang mampu membantu petugas dalam Pembuatan rekapitulasi secara cepat dan tepat.

4.2 Perancangan Sistem

Perencanaan sistem adalah proses dimana sebuah sistem dirancang dan nantinya untuk dilakukan pengembangan sistem. Perencanaan sistem ini menyangkut estimasi dari kebutuhan-kebutuhan fisik, tenaga kerja dan dana yang dibutuhkan untuk mendukung pengembangan sistem serta untuk mendukung operasinya setelah diterapkan.

Perancangan sistem dapat diartikan sebagai :

1. Tahap setelah analisis dari siklus pengembangan sistem.
2. Pendefinisian atas kebutuhan-kebutuhan fungsional.
3. Persiapan untuk rancang bangun implementasi.
4. Menggambarkan sistem yang akan dibentuk, berupa penggambaran perencanaan, pembuatan sketsa, pengaturan dari beberapa elemen terpisah ke dalam satu kesatuan yang utuh dan berfungsi.
5. Konfigurasi komponen *software* dan *hardware* sistem.

4.2.1 Tujuan Perancangan Sistem

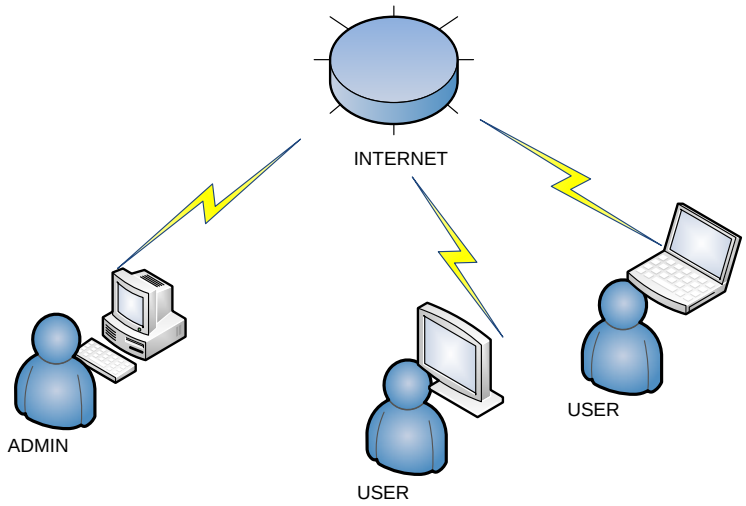
Perancangan sistem yang akan diaplikasikan ini bertujuan untuk memberikan gambaran secara umum kepada pengguna tentang sistem yang akan dibangun dan mengidentifikasikan komponen-komponen sistem informasi yang

akan didesain secara rinci. Tujuan perancangan sistem informasi penjualan online ini akan diuraikan sebagai berikut :

1. Sistem Informasi Penjualan pada Distro Celtic Bandung berbasis web, ditujukan untuk dapat memfasilitasi dan memudahkan pelanggan-pelanggan yang hendak membeli ataupun sekedar melihat-lihat produk Celtic namun tidak dapat datang langsung ke distro.
2. Dengan adanya Sistem Informasi Penjualan Pada Distro Celtic Berbasis Web, diharapkan dapat meningkatkan transaksi penjualan dan sekaligus sebagai media promosi yang lebih luas.

4.2.2 Gambaran Umum Sistem yang Diusulkan

Sistem yang diusulkan untuk pengolahan data penjualan produk Distro Celtic dilakukan secara *online*, dimana para *user* dapat mengaksesnya langsung tanpa harus hadir ke lokasi, dalam hal ini DistroCeltic. Sistem yang diusulkan pada Web ini bertindak sebagai *administrator* sedangkan *user* adalah para konsumen. Berikut adalah gambaran umum sistem yang diusulkan.



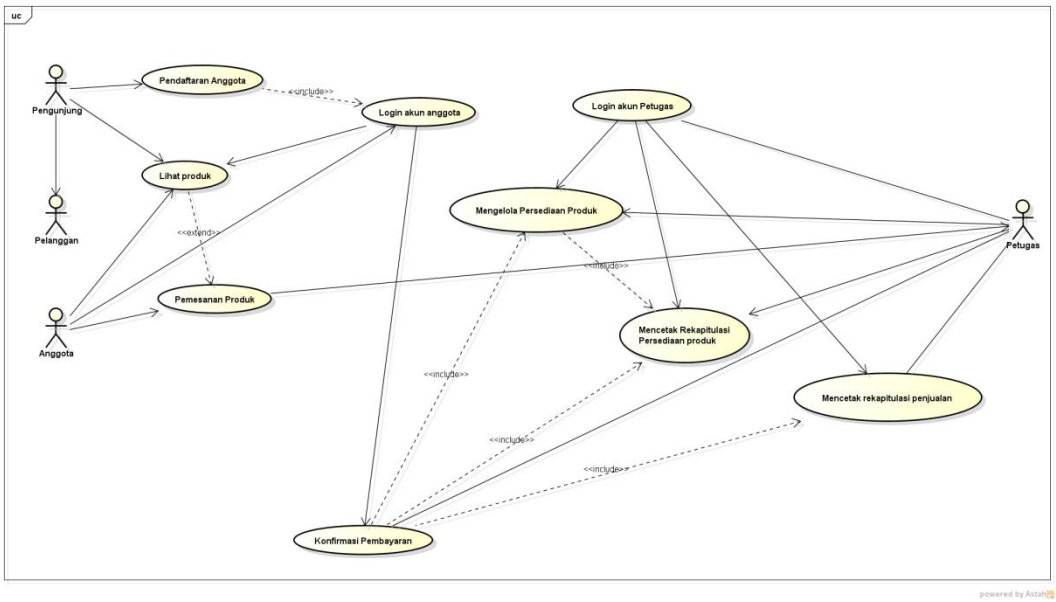
Gambar 4.7 Gambaran Umum Sistem yang Diusulkan.

4.2.3 Perancangan Prosedur Yang Diusulkan

Perancangan ini mencakup *use case diagram, activity diagram, collaboration diagram, class diagram, object diagram, sequence diagram, component diagram dan deployment diagram* yang menghasilkan sistem lebih baik. Proses yang dirancang diuraikan menjadi beberapa bagian yang dapat membentuk sistem tersebut menjadi satu kesatuan komponen.

4.2.3.1 Uses Case

Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun, berikut adalah *use case* diagram dari sistem informasi penjualan yang diusulkan.



Gambar 4.8. *Uses Case* Diagram Sistem Informasi Penjualan pada Distro Celtic berbasis web Yang Diusulkan.

4.2.3.2 Skenario Uses case

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. berikut skenario *use case* yang diusulkan.

1. Nama *Use Case* : Pendaftaran Anggota

   Aktor : Pengunjung

   Tujuan : Melakukan pendaftaran akun anggota

   Deskripsi : Dimana pengunjung melakukan proses pendaftaran yang kemudian data akunnya akan disimpan oleh sistem ke *database*

Tabel 4.7. Tabel Skenario *Use case* melakukan pendaftaran Anggota.

Pengunjung	Sistem
1. Masuk ke halaman pendaftaran akun anggota	
2. Melakukan pendaftaran	
	3. Menyimpan data akun anggota ke *database*

2. Nama *use case* : Lihat Produk

   Aktor : Pelanggan (pengunjung,anggota)

   Tujuan : Melakukan Pendaftaran akun anggota

   Deskripsi : Proses dimana pelanggan melihat-lihat produk

Tabel 4.8 Skenario *Use case* Lihat Produk.

Pelanggan	Sistem
1. Masuk ke halaman produk	
	2. Menampilkan daftar produk yang tersedia di website.

3. Nama *Use Case* : *Login* akun anggota

   Aktor : Anggota

   Tujuan : Melakukan *login* Anggota

   Deskripsi : Proses anggota melakukan *login* dengan mengisi form *login* dan akan di verivikasi oleh sistem.

Tabel 4.9 Skenario *Use case* Proses *Login* Anggota.

Anggota	Sistem
1. Masuk ke halaman *login* akun.	
2. Melakukan *login* akun anggota dengan mengisi email dan kata sandi.	
	3. Memverifikasi akun anggota.
	4. Menampilkan halaman utama anggota.

4. Nama *Use case* : Pemesanan Produk

   Aktor : Anggota

   Tujuan : Melakukan pemesanan produk

Deskripsi : Proses dimana anggota melakukan pemesanan produk kemudian data pesanan tersebut akan di verivikasi oleh system.

Tabel 4.10 Skenario *Use case* Proses Pemesanan Produk.

Anggota	**Sistem**
1. Masuk ke halaman produk dan memilih produk yang akan di pesan	
2. Mengisi form pemesanan	.
	3. Memverivikasi data pesanan.
	4. Menyimpan data pesanan ke database

5. Nama Use Case : Konfirmasi Pembayaran

   Aktor : Anggota

   Tujuan : Melakukan konfirmasi pembayaran berdasarkan pemesanan

   Deskripsi : Proses dimana anggota melakukan konfirmasi pembayaran setelah melakukan transfer.

Tabel 4.11 Skenario *Use case* Proses Konfirmasi Pembayaran.

Anggota	Sistem
1. Masuk ke halaman konfirmasi pembayaran berdasarkan pemesanan.	
2. Mengisi form konfirmasi.	
	3. Memverivikasi data konfirmasi
	4. Menyimpan data konfirmasi ke *database*.

6. Nama *use case* : *Login* akun petugas

   Aktor : Petugas

   Tujuan : Melakukan *Login* akun petugas

   Deskripsi : Proses dimana petugas melakukan *login* terhadap system

Tabel.4.12 Skenario *Use case* Proses *Login* Petugas

Petugas	Sistem
1. Masuk ke halaman login akun Petugas.	
2. Melakukan *login* akun petugas dengan mengisi *usename* dan kata sandi.	
	3. Memverifikasi akun petugas.
	4. Menampilkan halaman utama petugas.

7. Nama use case : Mengelola persedian produk

   Aktor : Petugas

   Tujuan : Memanipulasi Persedian Produk

   Deskripsi : Proses dimana dilakukannya perubahan data produk oleh petugas

Tabel 4.13. Skenario *Use case* Mengelola Persediaan Produk

Petugas	Sistem
1. Masuk ke halaman produk.	
2. Memanipulasi data-data atau informasi produk.	
	3. Menyimpan perubahan data produk ke *database*.

8. Nama use case : Mencetak rekapitulasi persedian produk

   Aktor : Petugas

   Tujuan : Membuat dan mencetak rekapitulasi persediaan produk

   Deskripsi : Proses dimana petugas membuat laporan persedian produk dan mencetaknya.

Tabel 4.14. Skenario *Use case* Mencetak Rekapitulasi Persedian Produk.

Petugas	Sistem
1. Masuk ke halaman produk dan membuat rekapitulasi persediaan produk.	
	2. Menampilkan data-data produk.
	3. Mencetak rekapitulasi persediaan Produk.

9. Nama use case : Mencetak rekapitulasi transaksi penjualan produk

   Aktor : Petugas

   Tujuan : Membuat dan mencetak rekapitulasi transaksi penjualan produk

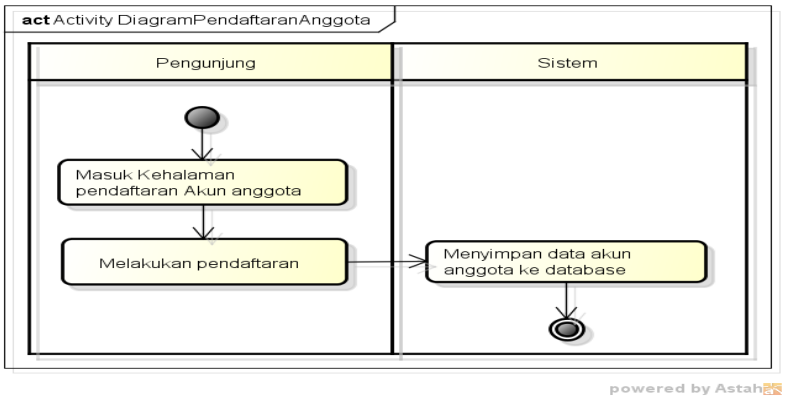
   Deskripsi : Proses dimana petugas membuat rekapitulasi transaksi penjualan produk dan mencetaknya

Tabel 4.15. Skenario *Use case* Mencetak rekapitulasi transaksi penjualan

Petugas	Sistem
1. Masuk ke halaman transaksi penjualan dan membuat rekapitulasi transaksi penjualan.	
	2. Menampilkan data-data transaksi penjualan.
	3. Mencetak rekapitulasi transaksi penjualan.

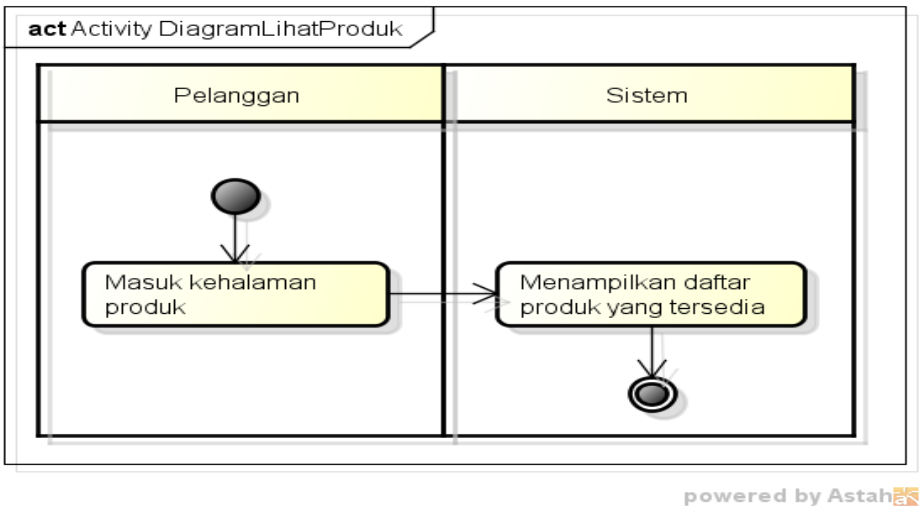
4.2.3.3 Activity Diagram

.*Activity diagram* berikut ini memperlihatkan secara rinci aliran data secara logika tanpa mempertimbangkan lingkungan fisik dimana data mengalir. Berikut *activity diagram* yang diusulkan pada distro Celtic.



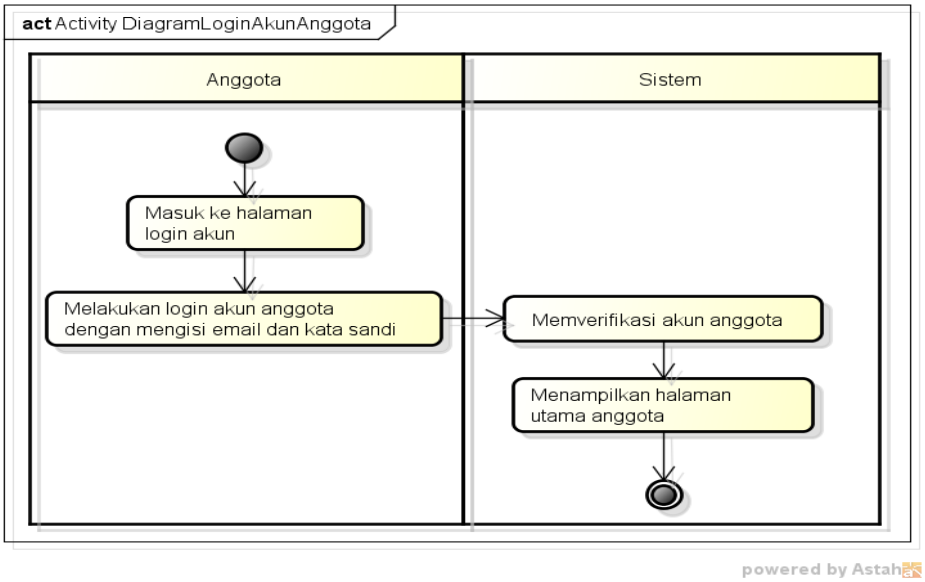
Gambar 4.9. *Activity Diagram* Pendaftaran anggota pada distro Celtic Berbasis Web yang Diusulkan.

Pada *activity diagram* pendaftaran anggota ini, dimana pegawai masuk ke halaman pendaftaran akun anggota dan melakukan pendaftaran dengan cara mengisi form pendaftaran, lalu sistem menyimpan data akun anggota ke *database*.



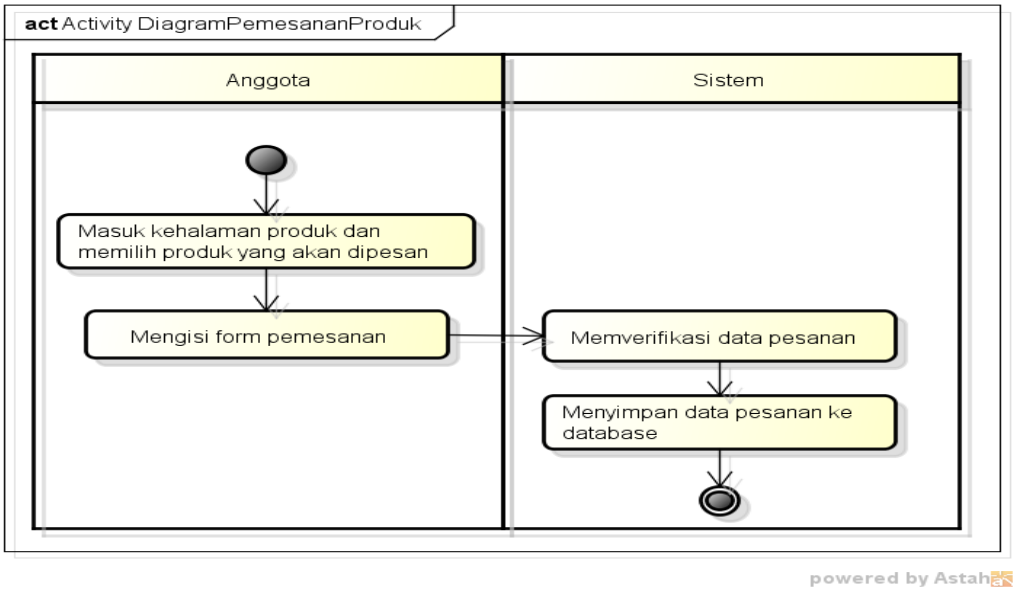
Gambar 4.10. *Activity Diagram* lihat Produk pada distro Celtic Berbasis Web yang Diusulkan.

Dalam diagram ini, menggambarkan dimana pelanggan masuk ke halaman produk dan sistem secara otomatis menampilkan daftar produk yang tersedia.



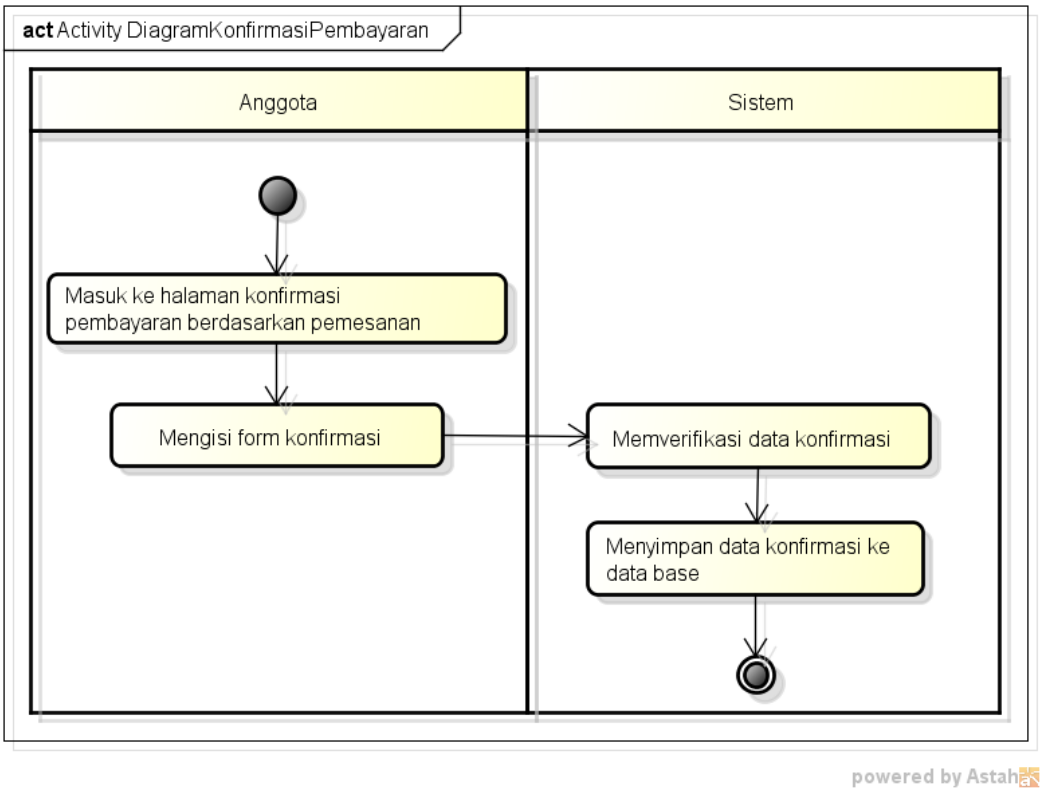
Gambar 4.11. *Activity Diagram* Login akun anggota pada distro Celtic Berbasis Web yang Diusulkan.

Pada *activity diagram login* akun anggota ini, menggambarakan bagaimana anggota masuk ke halaman *login* anggota dan melakukan *login* anggota dengan mengisi *email* dan kata sandi, kemudian sistem memverifikasi akun anggota tersebut dan menampilkan halaman utama anggota.



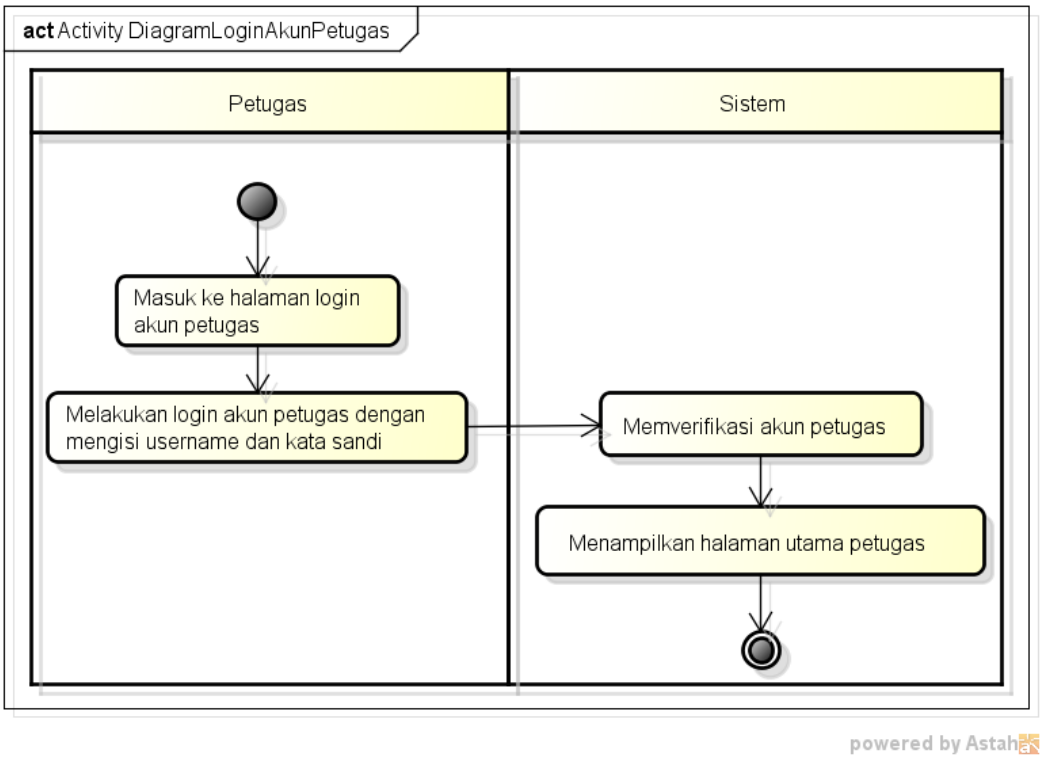
Gambar 4.12. *Activity Diagram* Pemesanan Produk pada distro Celtic Berbasis Web yang Diusulkan.

Pada *activity diagram* ini, menggambarkan dimana anggota masuk kehalaman produk dan memilih produk yang akan dipesan lalu mengisi form pemesanan. Kemudian sistem memverifikasi data pesanan, kemudian menyimpan data pesanan tersebut ke *database*.



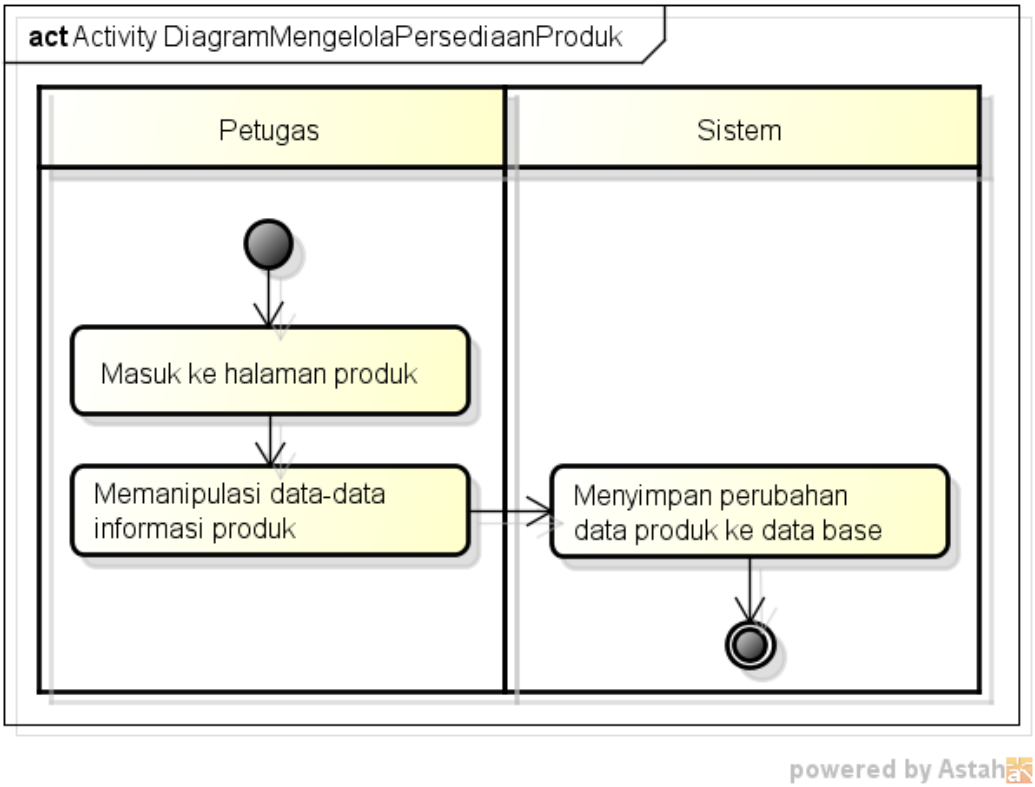
Gambar 4.13. *Activity Diagram* Konfirmasi Pembayaran pada distro Celtic Berbasis Web yang Diusulkan.

Anggota masuk kehalaman konfirmasi pembayaran berdasarkan pemesanan, kemudian mengisi form konfirmasi dan sistem memverifikasi data konfirmasi kemudian menyimpan data konfirmasi tersebut ke *database*.



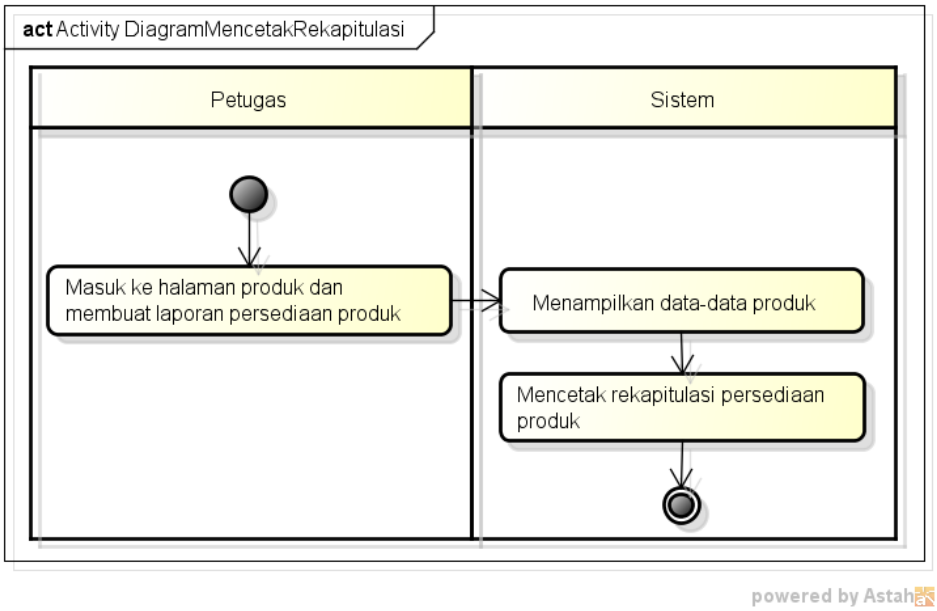
Gambar 4.14. *Activity Diagram Login* Petugas pada distro Celtic Berbasis Web yang Diusulkan.

Petugas masuk ke halaman *login* petugas dan melakukan *login* akun petugas dengan mengisi *username* dan kata sandi, kemudian sistem memverifikasi akun petugas tersebut lalu menampilkan halaman petugas.



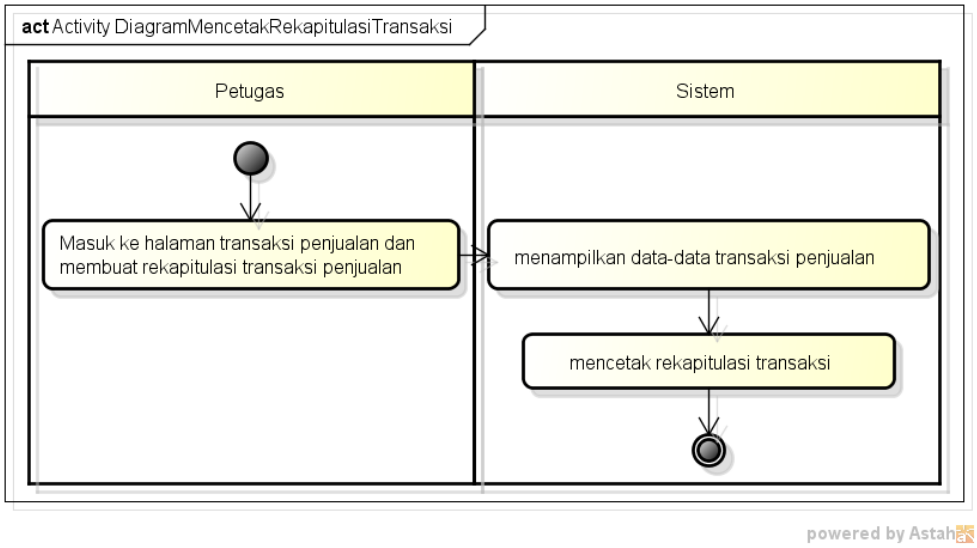
Gambar 4.15. *Activity Diagram* Mengelola persediaan produk pada distro Celtic Berbasis Web yang Diusulkan.

Petugas masuk ke halaman produk dan memanipulasi data-data/ informasi produk, kemudian sistem menyimpan perubahan data produk tersebut ke *database*.



Gambar 4.16. *Activity Diagram* Mencetak rekapitulasi persediaan produk pada distro Celtic Berbasis Web yang Diusulkan.

Petugas disini masuk ke halaman produk dan membuat laporan persediaan produk, kemudian sistem menampilkan data-data produk dan mencetak laporan persediaan produk.



Gambar 4.17. *Activity Diagram* Mencetak Rekapitulasi Transaksi Penjualan pada distro Celtic Berbasis Web yang Diusulkan.

Petugas masuk ke halaman transaksi penjualan dan membuat laporan penjualan, kemudian sistem menampilkan data-data transaksi penjualan dan mencetak laporan transaksi.

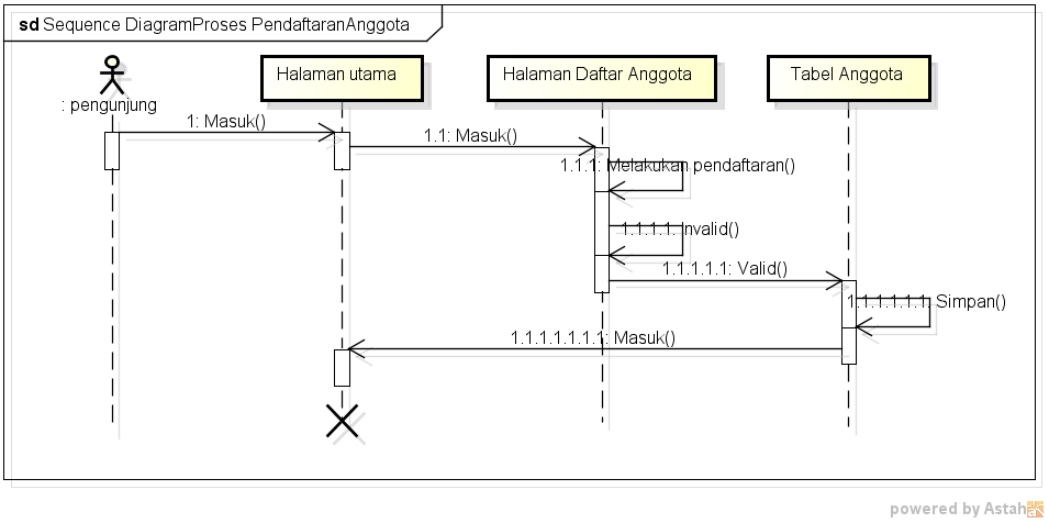
4.2.3.4 Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. Tetapi pada dasarnya sequence Diagram selain digunakan dalam lapisan abstraksi model objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Komponen utama *sequence* diagram terdiri atas objek yang dituliskan dengan kotak segiempat bernama pesan diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan proses vertikal.

Berikut adalah *sequence diagram* yang ada pada sistem penjualan Produk distro Celtic, yaitu:

1. **Sequence Diagram Pendaftaran Member**

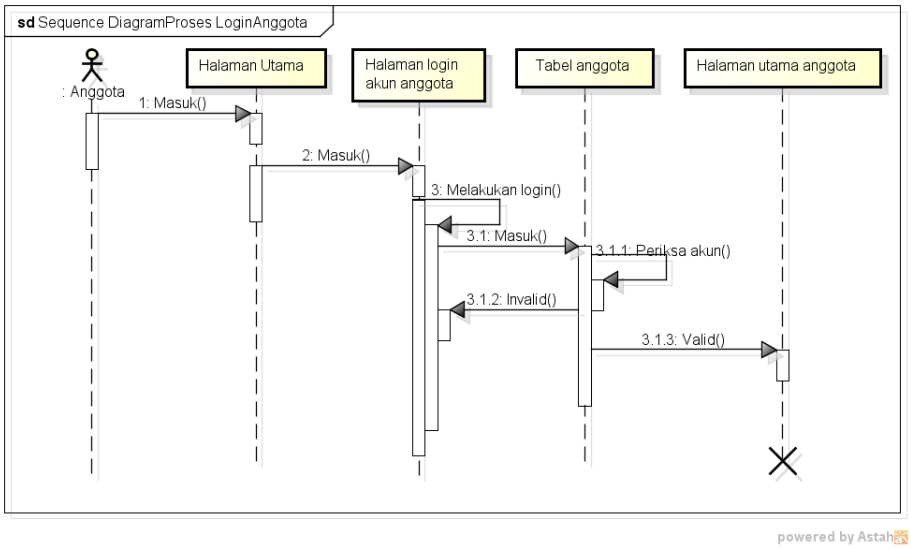
   Pengunjung masuk ke halaman utama web distro Celtic, apabila pengunjung ingin dapat melakukan transaksi maka harus mendaftar terlebih dahulu. Dengan cara masuk ke halaman daftar anggota dan melakukan pendaftaran dengan mengisikan data di form pendaftaran. Kemudian sistem akan memvalidasi data tersebut dan menyimpanya ke *database*.



Gambar 4.18. *Sequence Diagram* Proses Pendaftran Anggota yang diusulkan.

2. Sequence Diagram Untuk Login Anggota

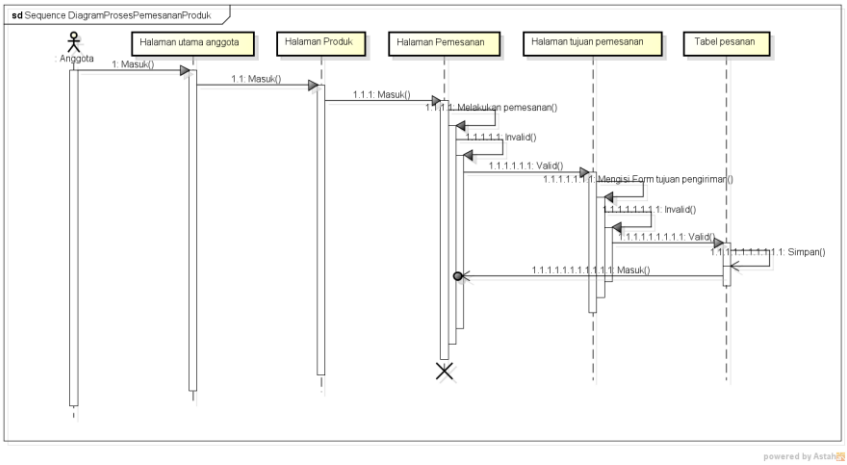
Anggota masuk kehalaman utama, kemudian masuk ke halaman *login* akun anggota untuk melakukan proses *login*. Kemudian sistem akan memvalidasi akun *login* tersebut, sehingga jika valid anggota dapat masuk ke halaman utama anggota untuk melakukan proses selanjutnya.



Gambar 4.19. *Sequence Diagram* Proses Login Anggota yang diusulkan.

3. Sequence Diagram Untuk Pemesanan Produk

Pada proses ini dimana anggota masuk ke halaman utama anggota setelah melakukan proses login terlebih dahulu, kemudian masuk ke halaman produk dan melakukan pemesanan dihalaman pemesanan. Kemudian sistem akan memvalidasi pemesanan tersebut, jika valid maka akan langsung diarahkan kehalaman tujuan pemesanan untuk mengisi form tujuan pengiriman, lalu sistem melakukan validasi untuk tujuan pemesanan.jika valid maka data pemesanan akan tersimpan kedalam table pemesanan.

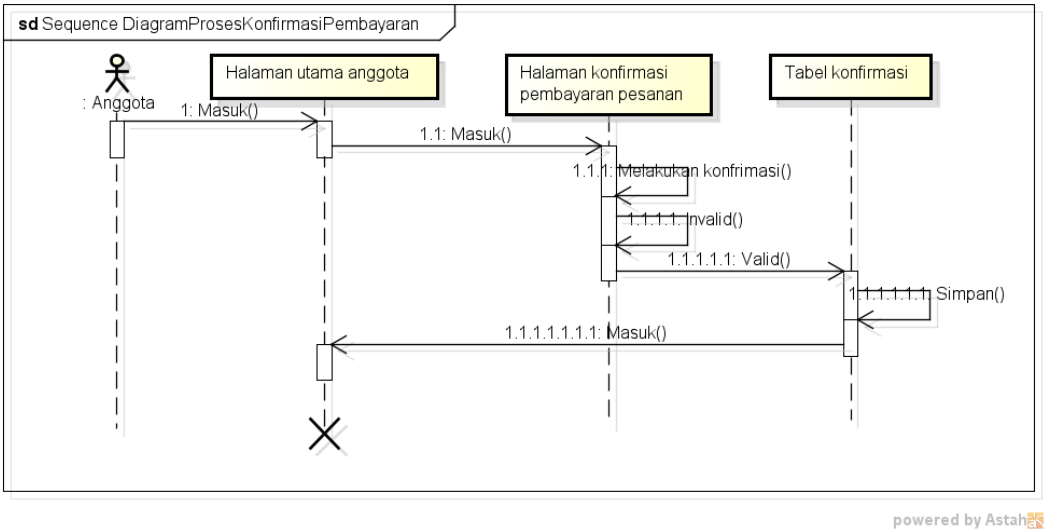


Gambar 4.20. *Sequence Diagram* Proses Pemesanan Produk yang diusulkan.

4. Sequence Diagram Untuk Konfirmasi Pembayaran oleh Anggota

Pada diagram ini menggambarkan bagaimana proses konfirmasi pembayaran yang dilakukan oleh anggota dimana anggota terlebih dahulu masuk kehalaman utama anggota, kemudian masuk kehalaman konfirmasi pembayaran pemesanan, dan mengisi form konfirmasi pemesanan.

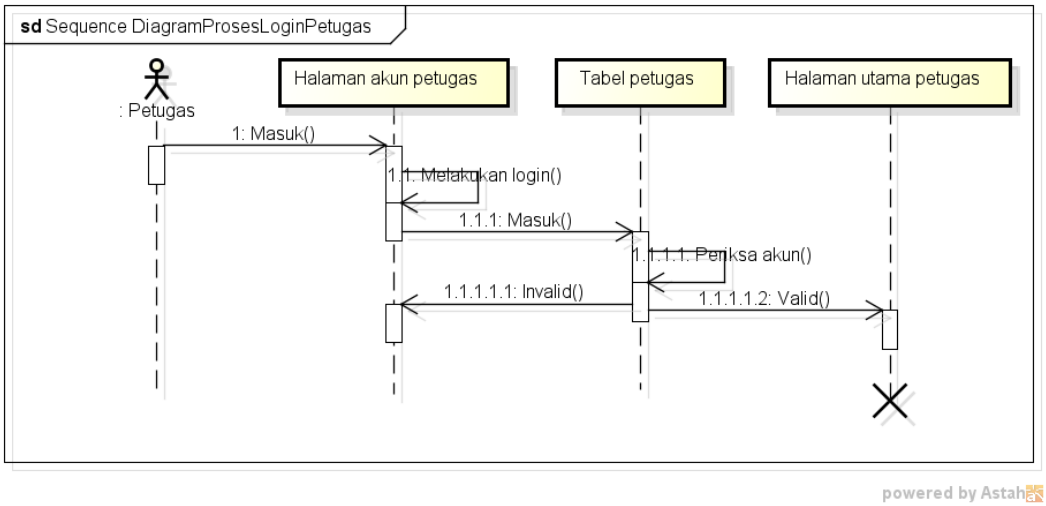
Kemudian sistem melakukan validasi, jika data yang di isi valid maka data tersebut akan tersimpan kedalam tabel konfirmasi.



Gambar 4.21. *Sequence Diagram* Proses Konfirmasi Pembayaran oleh Anggota yang diusulkan.

5. Sequence Diagram Untuk Login Petugas

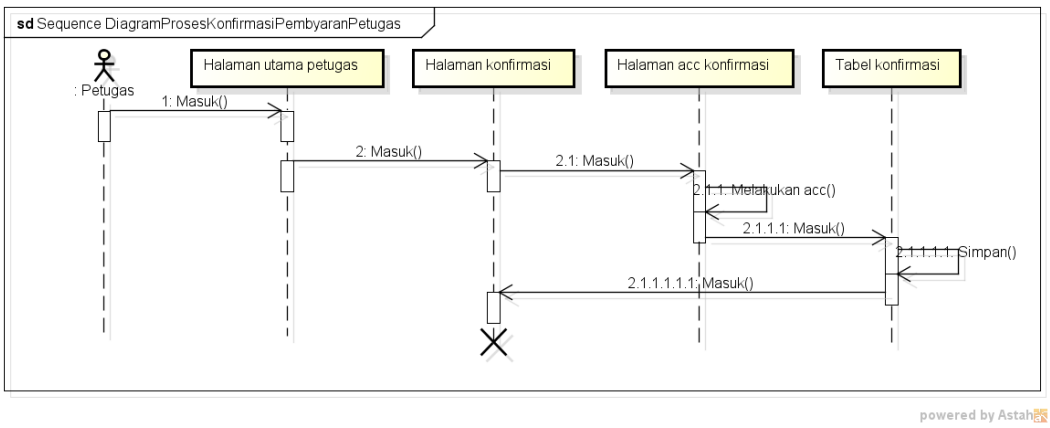
Dimana petugas masuk ke halaman *login* akun petugas untuk melakukan proses *login*. Kemudian sistem akan memvalidasi akun *login* tersebut, sehingga jika valid petugas dapat masuk ke halaman utama petugas untuk melakukan proses selanjutnya.



Gambar 4.22. *Sequence Diagram*Proses *Login* Petugas yang Diusulkan.

6. Sequence Diagram Untuk Konfirmasi Pembayaran oleh Petugas

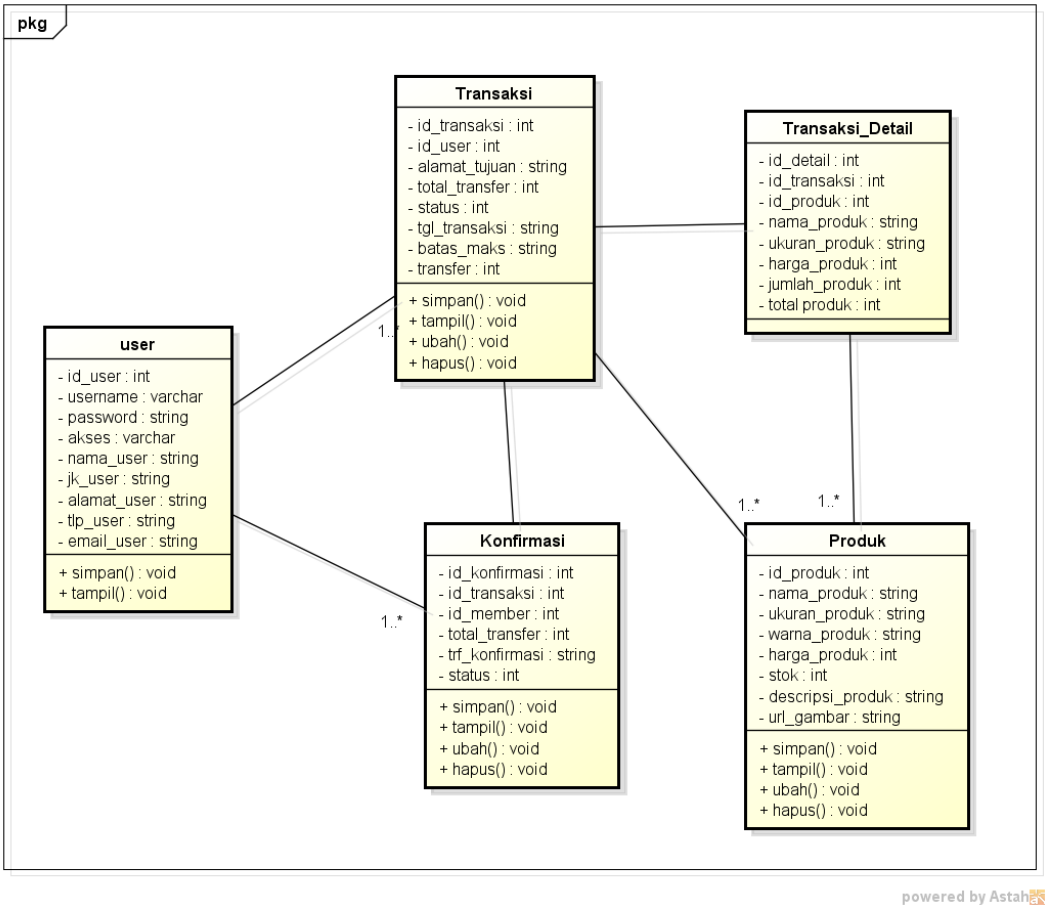
Petugas masuk kehalam utama petugas, kemudian masuk kehalaman konfirmasi hingga masuk ke halaman *acc* konfirmasi untuk melakukan konfirmasi pembayaran yang telah dilakukan oleh anggota. kemudian data yang tersimpan ke dalam tabel konfirmasi dan sistem langsung menampilkan halaman tabel konfirmasi.



Gambar 4.23. *Sequence Diagram* Proses Konfirmasi Pembayaran oleh Petugas yang Diusulkan.

4.2.3.5 Class Diagram

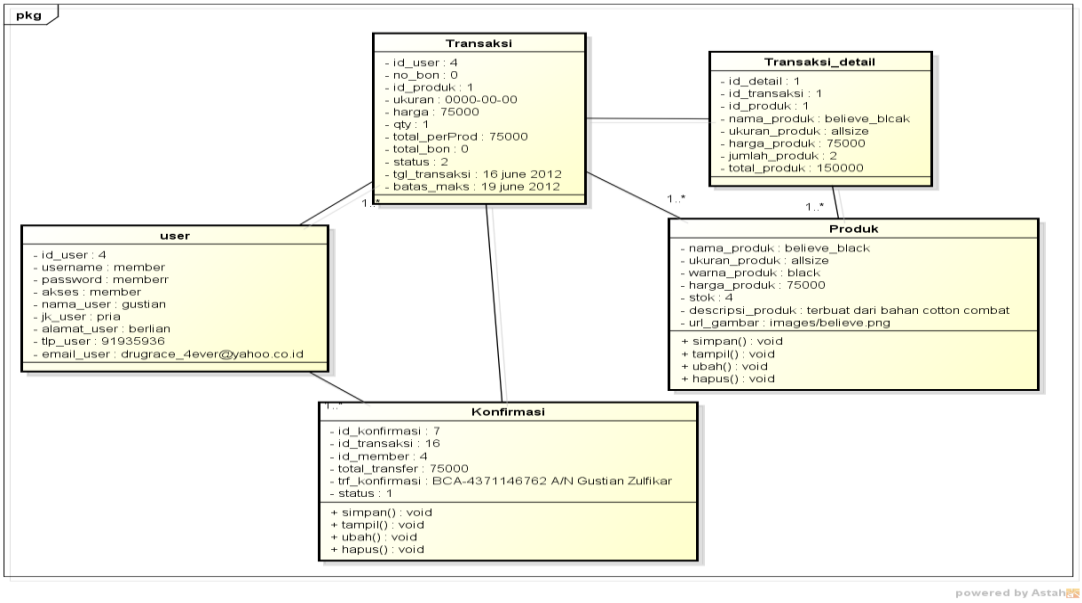
Class diagram bertujuan untuk menggambarkan struktur statis *class* di dalam sistem. *Class* berfungsi untuk merepresentasikan sesuatu yang ditangani oleh sistem. Dengan melihat karakteristik sistem pemasaran produk mulai dari bagian pendaftaran sampai dengan pemesanan beserta proses-proses yang terjadi, maka dapat dibuat C*lass Diagram* Berikut *Class Diagram* Sistem Informasi Penjualan produk di Distro Celtic.



Gambar 4.24. *Class Diagram* Sistem Informasi Penjualan pada Distro Celtic yang Diusulkan.

4.2.3.6 Object Diagram

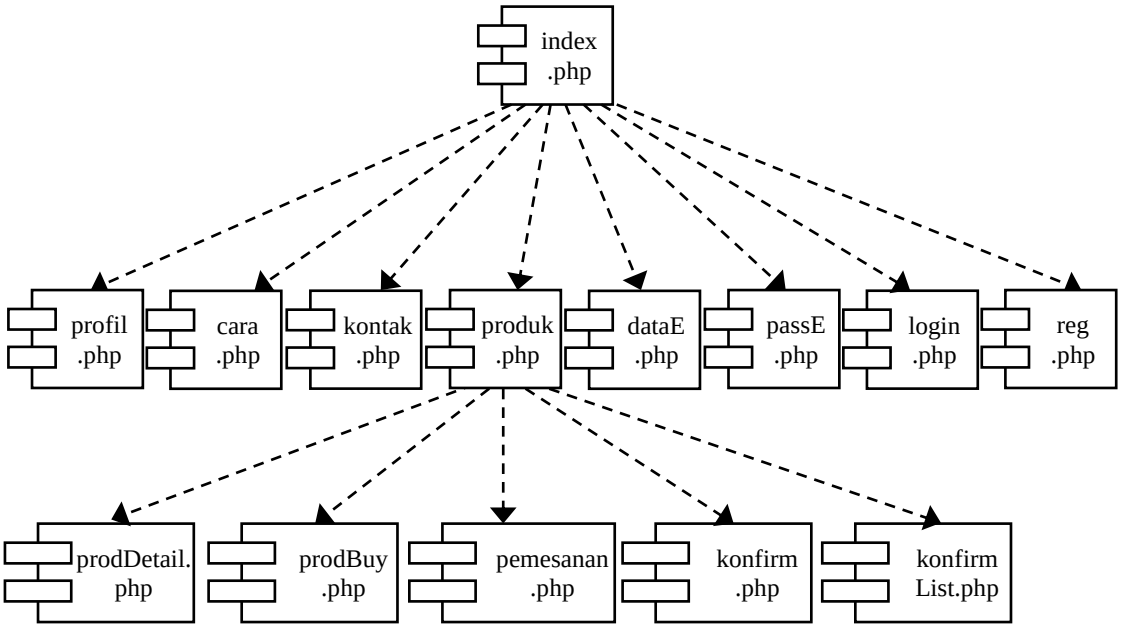
Diagram objek ialah merupakan suatu diagram yang mengambarkan objek-objek dalam suatu system. Bila pada kelas diagram terdapat banyak relasi antar kelas, namun pada diagram objek, hanya terdapat satu relasi antar kelas, yaitu asosiasi. Tidak hanya itu, pada diagram objek yang didefinisikan hanya nama objek dalam suatu kelas beserta atributnya saja, tanpa operasi yang dilakukan oleh objek tersebut.



Gambar 4.25. *Object Diagram* Sistem Informasi Penjualan pada Distro Celtic yang Diusulkan.

4.2.3.7 Component Diagram

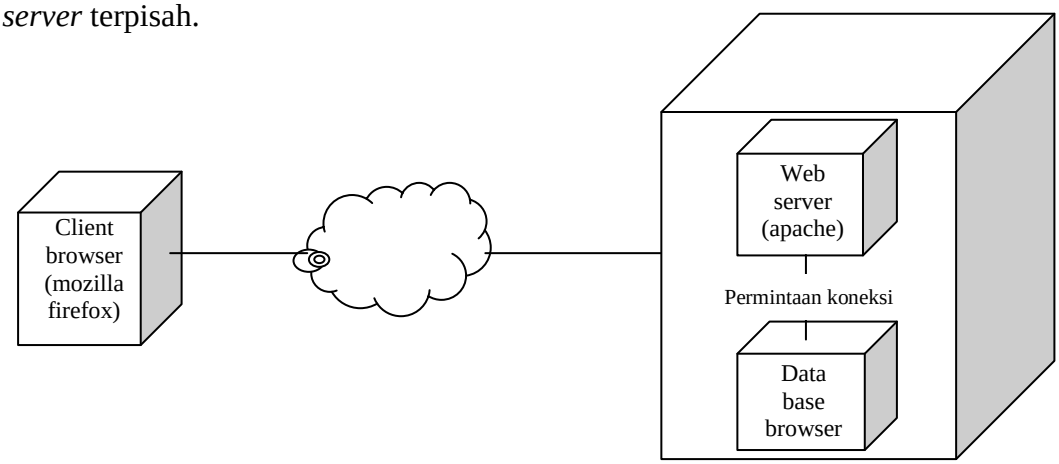
Component adalah unit fisik yang nyata yang menjadi bagian dari *deployment independent*. *Component* ini diimplementasikan meskipun pada sistem yang kecil. Perancangan menu pada aplikasi yang diusulkan digambarkan oleh gambar *component diagram* berikut.



Gambar 4.26. *Component Diagram* Sistem Informasi Penjualan pada Distro Celtic yang Diusulkan.

4.2.3.8 Deployment Diagram

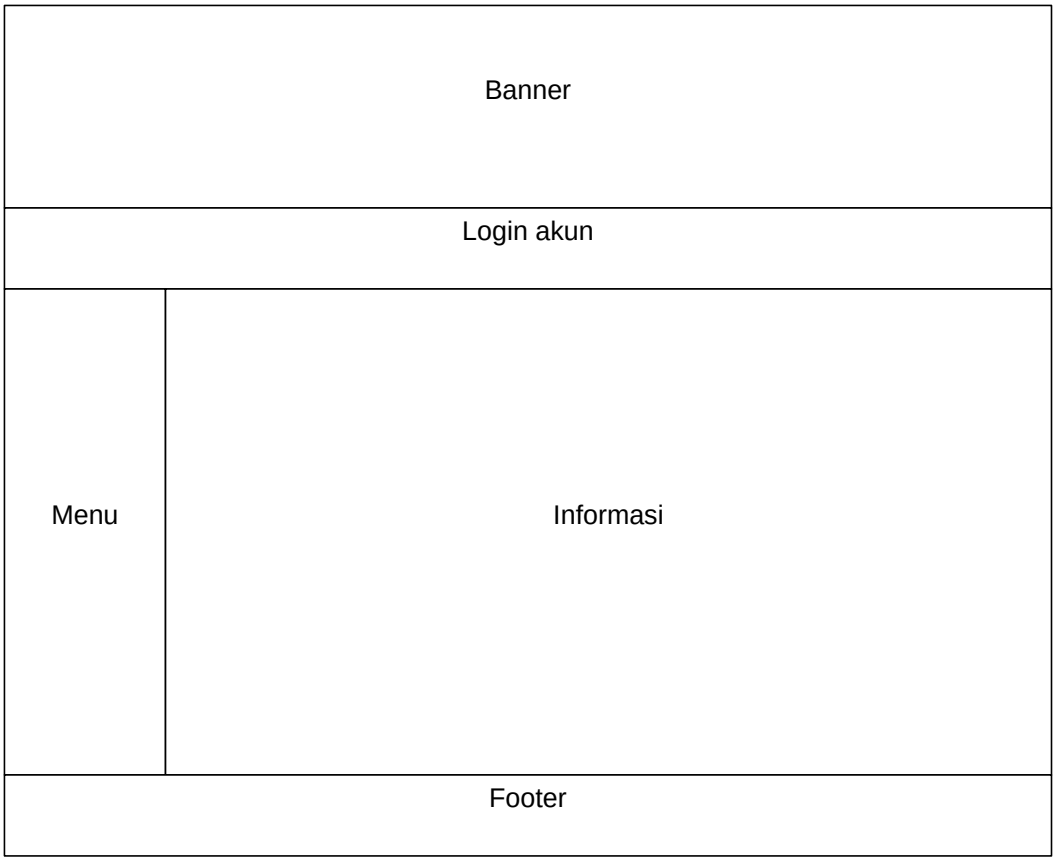
Asumsi yang digunakan pada perangkat lunak ini memakai sistem yang *stand alone*, artinya tidak ada satu komponenpun yang akan dibagi pakai *(sharing)* dengan aplikasi *web* yang lain, oleh karena itu tidak perlu *application server* terpisah.



Gambar 4.27. *Deployment Diagram* Sistem Informasi Penjualan pada Distro Celtic yang Diusulkan.

4.2.4 Perancangan Antar Muka

Perancangan ini merupakan desain utama sebagai *user interface* yang akan digunakan oleh sistem yang akan dirancang.



Gambar 4.28. Perancangan antar muka .

4.2.4.1 Perancangan Input

Perancangan input dirancang sebagai media dimana pengguna atau administrator dapat dengan mudah menyimpan data yang nantinya akan diperlukan baik itu oleh sistem maupun oleh pengguna itu sendiri. Perancangan input dari perangkat lunak yang akan dibangun adalah sebagai berikut :

1. Pendaftaran Anggota

Berikut merupakan perancangan dari formulir pendaftaran penggguna sebagai anggota yang nantinya akan digunakan untuk memudahkan proses selanjutnya.

Data Login	
Username :	
Password :	
Ulangi Password :	
Data User	
Nama :	
Jenis Kelamin :	Pria v
Alamat :	
Telepon :	
Email :	
	Daftar

Gambar 4.29. Perancangan Form *Input* Pendaftran Anggota

2. Login Anggota

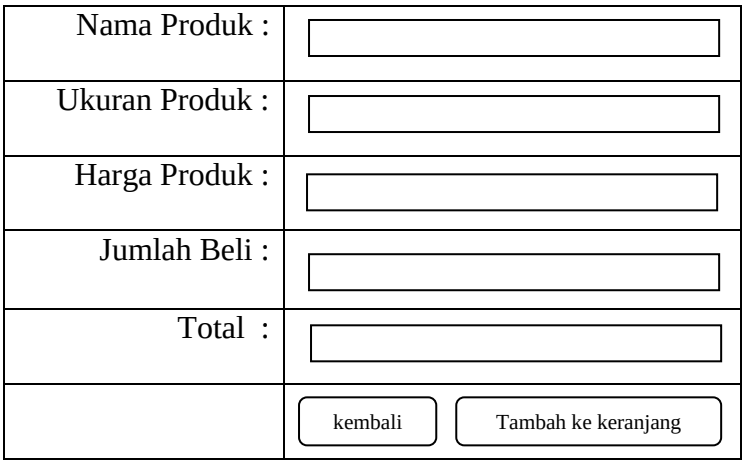
Perancangan form ini sebagai tahap selanjutnya dari pengguna yang telah menjadi anggota untuk dapat melanjutkan ke tahap selanjutnya dengan melakukan *login* terlebih dahulu dengan mengisikan *username* dan *password*.



Gambar 4.30. Perancangan Tampilan Halaman *Login* Anggota.

3. Pemesanan Produk

Berikut merupakan perancangan form dimana anggota dapat melakukan proses pemesanan dengan memilih produk yang akan dipesan terlebih dahulu dan menentukan detail dari produk dan pesanan tersebut.



Gambar 4.31. Perancangan Tampilan Halaman Pemesanan Produk.

4. Konfirmasi Pembayaran

Perancangan form ini bertujuan sebagai konfirmasi pembayaran dari anggota yang telah melakukan pembayaran atas produk yang telah dipesannya.

Jumlah yang Harus Anda Transfer Untuk Pesanan Anda Adalah Rp xxxx	
Total Transfer :	xxxxxx
Transfer Via :	xxxxx
	konfirmasikan

Gambar 4.32. Perancanagan Tampilan Konfirmasi Pembayaran Pesanan.

5. Form Login Petugas

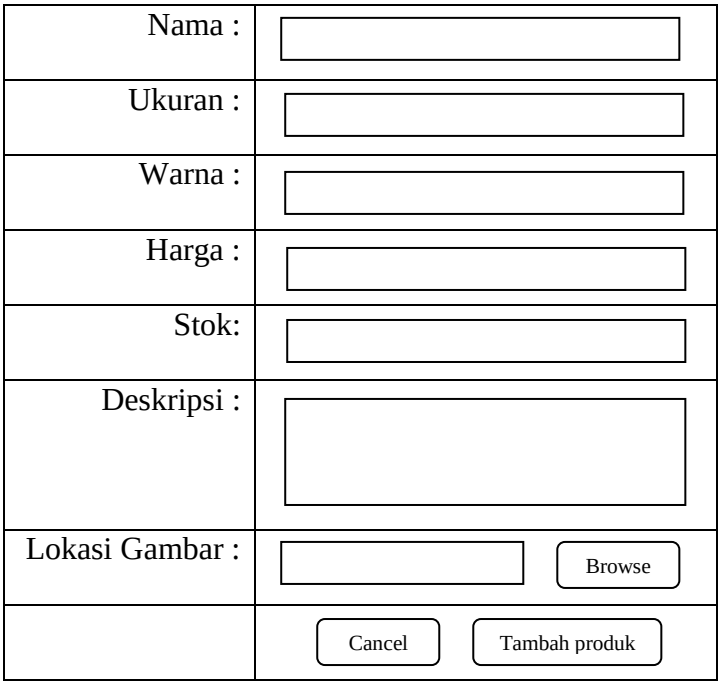
Sama halnya dengan perancangan form *login* anggota, form *login* petugas inipun bertujuan untuk memberikan hak akses ke pada petugas untuk melakukan pengolahan data dengan menginputkan *username* dan *password*.

Username	Password	Login

Gambar 4.33. Perancangan Tampilan Form *Login* Petugas.

6. Form Tambah Produk

Perancangan form input tambah produk ini bertujuan agar petugas dapat menambah jumlah dari suatu produk yang akan ditawarkan oleh distro Celtic.



Gambar 4.34 Perancangan Tampilan Form Tambah Produk.

4.2.4.1.1 Perancangan Output

Perancangan *output* merupakan rencana pembuatan antarmuka yang bertujuan untuk menampilkan keluaran program. Berikut ini adalah perancangan *output* pada sistem yang akan dirancang.

1. **Tampilan Produk**

   Perancangan *output* ini bertujuan untuk menampilkan daftar dari produk-produk baik yang ditawarkan kepada pelanggan atau pun sebagai *output* saja bagi petugas untuk melakukan manipulasi terhadap data tersebut.

Nama Produk		Nama Produk	
Gambar		Gambar	
Harga :	xxxx	Harga :	xxxx
	Detail - Buy		Detail - buy

Gambar 4.35. Perancanagan Tampilan Produk.

2. Tampilan Detail Produk

Perancangan *output* ini bertujuan untuk menampilkan daftar detail produk baik yang ditawarkan kepada pelanggan atau pun sebagai *output* saja bagi petugas untuk melakukan manipulasi terhadap data tersebut.

Nama Produk	
Gambar	Ukuran : xx
	Warna : xxxx
	Harga : xxxx
	Deskripsi : xxxx
	Stock : xx
	Buy

Gambar 4.36. Perancanagan Tampilan Detail Produk.

3. Tampilan Daftar Keranjang

Perancangan tampilan ini bertujuan untuk menampilkan semua daftar pembelanjaan yang belum di konfirmasi kan oleh anggota.

Aksi	Nama Barang	Ukuran	Harga	Jumlah	Total
xx	xxx	xxxx	xx	xx	xxxx
				Grand Total :	
Alamat Pengiriman :			Pembayaran : V		
			Check out		

Gambar 4.37 Perancangan Tampilan Daftar Keranjang.

4. Tampilan Daftar Pesanan Produk

Perancangan daftar pesanan produk ini bertujuan untuk menampilkan semua daftar pesanan yang telah dilakukan oleh anggota sebelum dilanjutkan ke tahap selanjutnya.

ID pemesanan	Tanggal	Alamat Pengiriman	Total Tertagih	Batas Transfer	Status Pesanan	Status Pembayaran
x	xxxx	xxxx	x	xxxx	xxxx	xxxx

Gambar 4.38. Perancangan Tampilan Daftar Pesanan Produk.

5. Tampilan Daftar Konfirmasi Pemesanan

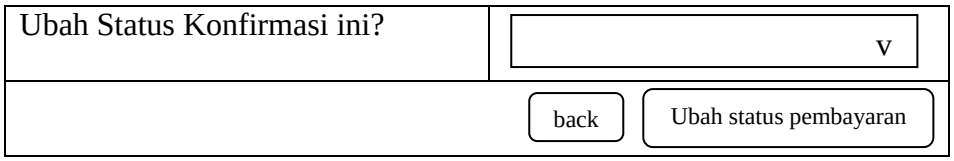
Perancangan tampilan ini bertujuan untuk menampilkan semua daftar pesanan baik yang sudah maupun yang belum dikonfirmasi oleh petugas.

ID Pemesanan	Jumlah yang telah ditransfer	Transfer Via (Bank)	Status
xx	xxxxx	xxxx	xxxx

Gambar 4.39 Perancangan Tampilan Daftar Konfirmasi Pesanan.

6. Tampilan Perubahan Status Kelola Pembayaran

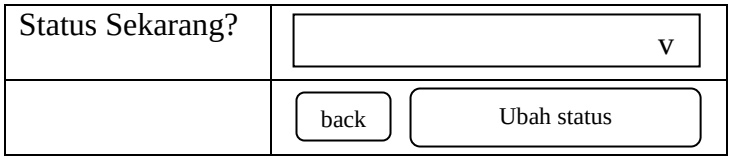
Perancangan tampilan ini bertujuan untuk menampilkan perubahan status kelola pembayaran yang sudah maupun yang belum masuk.pada form ini dilakukan oleh petugas.



Gambar 4.40 Perancangan Tampilan Perubahan Satus Kelola Pembayaran.

7. Tampilan Perubahan Status Kelola Penjualan

Perancangan tampilan ini bertujuan untuk menampilkan perubahan status produk yang telah di pesan oleh anggota,pada form ini dilakukan oleh petugas.



Gambar 4.41 Perancangan Tampilan Perubahan Satus Kelola Penjualan.