# PERANCANGAN SISTEM INFORMASI PEMESANAN PRODUK KONVEKSI DI PERSADA KONVEKSI TINGKIR SALATIGA

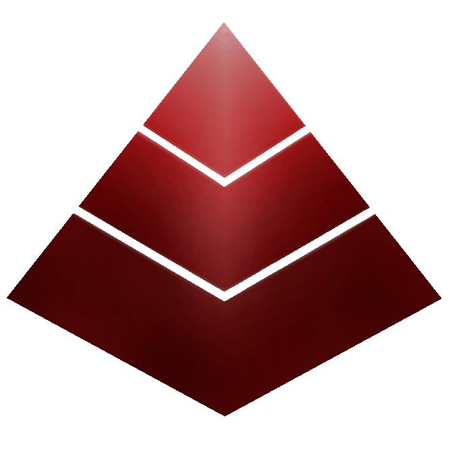
# BERBASIS WEBSITE

**TUGAS AKHIR**

Diajukan untuk menempuh Ujian Akhir Program Diploma Tiga Jurusan Manajemen Informatika Fakultas Ilmu Komputer

**MAR’ATUS SHOLIKHA**

**NIM 02041911022**



**UNIVERSITAS NASIONAL PASIM**

**BANDUNG**

**2021**

**BAB 1**

**PENDAHULUAN**

* 1. **Latar Belakang**

Seiring dengan perkembangan zaman, kemajuan teknologi informasi juga semakin pesat dari masa ke masa, begitupun dengan persaingan dalam dunia bisnis yang semakin ketat, para pelaku usaha dan bisnis dituntut untuk dapat menerapkan teknologi pada zaman sekarang untuk kemajuan bisnis mereka.

Konveksi adalah salah satu usaha industri yang bergerak pada bidang pembuatan pakaian jadi seperti kemeja, kaus, jaket, celana dan sebagainya. Adanya konveksi sangat mempengaruhi perkembangan ekonomi masyarakat sekitar, konveksi membutuhkan tenaga manusia dan mesin untuk proses pembuatan pakaian, mesin yang digunakan yaitu mesin jahit, mesin obras, mesin border, dan sebagainya. Jumlah pegawai dalam konveksi rata – rata adalah 10 – 20 orang, tergantung pada besarnya jumlah pemesanan setiap konveksi.

Persada Konveksi merupakan salah satu usaha konveksi yang berada di Kec. Tingkir, Kota Salatiga, Jawa Tengah. Persada Konveksi menawarkan berbagai macam produk pakaian. Pakaian yang diproduksi mencakup kemeja, kaos, celana, jaket dan lainnya. Persada Konveksi memasarkan produksinya dengan menerima pesanan yang bermacam - macam, seperti pesanan seragam sekolah, seragam kerja, seragam PKK dan sebagainya.

Berdasarkan hasil wawancara yang dilakukan, dalam melakukan proses pemesanan antara pihak konveksi dan *customer*, *customer* harus menjelaskan deskripsi produk pesanan secara rinci, baik menjelaskan secara langsung maupun melalui whatsapp, email, sms atau telepon. Hal tersebut seringkali menimbulkan kekurangan dimana informasi tidak sesuai dengan informasi yang dibutuhkan pihak konveksi, sehingga keduanya harus melakukan konfirmasi untuk mendapatkan informasi yang lebih jelas.

Berdasarkan hal tersebut maka Persada Konveksi membutuhkan media pemesanan berupa *website* sebagai media pemesanan dan pemasaran*.* Dengan adanya media tersebut maka proses pemesanan akan lebih efektif dan efisien, *customer* juga mendapatkan informasi produk yang lebih detail serta dapat memperluas jangkauan pemasaran Persada Konveksi. *Website* memberikan informasi mengenai Persada Konveksi dan produk – produk yang dihasilkan kepada masyarakat luas. Maka dari itu penulis tertarik untuk melakukan penelitian yang ditulis dalam Laporan Tugas Akhir dengan judul: **“PERANCANGAN SISTEM INFORMASI PEMESANAN PRODUK KONVEKSI DI PERSADA KONVEKSI TINGKIR SALATIGA BERBASIS WEBSITE”.**

* 1. **Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka pokok permasalahan yang akan dibahas pada Tugas Akhir ini adalah:

1. Bagaimana merancang sistem informasi yang dapat melakukan proses pemesanan produk pada Persada Konveksi berbasis *web* dengan menggunakan bahasa pemrograman java dan menggunakan *framework spring-boot*.
2. Bagaimana merancang aplikasi manajemen dan pemasaran berbasis *web* yang dapat digunakan untuk mempermudah proses pemesanan dan promosi produk.
3. Bagaimana mempermudah masyarakat dalam mengakses sistem informasi pemesanan konveksi di Persada Konveksi Tingkir.
   1. **Maksud dan Tujuan Penelitian** 
      1. **Maksud penelitian**

Maksud dari penelitian ini adalah membangun sebuah sistem informasi pemesanan produk pada Persada Konveksi berbasis *web* sehingga dapat mempermudah pekerja konveksi dalam melayani pemesanan dari *customer*.

* + 1. **Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Membangun sistem informasi yang dapat melayani proses pemesanan peroduk Persada Konveksi.
2. Membangun sistem informasi untuk memudahkan pekerja konveksi mengelola pemesanan produk.
3. Membangun sistem informasi untuk memperluas jangkauan pemasaran produk Persada Konveksi.
   1. **Batasan Masalah**

Sistem informasi pemesanan produk yang dibuat ini memiliki beberapa batasan masalah sebagai berikut:

1. Sistem yang dibahas melingkupi informasi tentang profil Persada Konvaksi, katalog produk, formulir pemesanan, *progress* proses produksi pesanan dan laporan pemesanan.
2. Pesanan mulai diproduksi dan akan dikonfirmasi oleh admin jika cutomer telah melakukan pembayaran minimal sebesar 20% dari keseluruhan total pembayaran.
3. Sistem dibangun menggunakan Bahasa Pemrograman Java dengan *framework Spring-Boot* dan DBMS SQLyog.
   1. **Kegunaan Penelitian**
4. Manfaat bagi peneliti
5. Belajar mengimplementasikan konsep manajemen dan pemasaran ke dalam sistem informasi.
6. Menambah pengetahuan dan wawasan penulis dalam membangun program aplikasi yang dapat diterapkan dalam suatu perusahaan.
7. Sebagai syarat kelulusan Diploma Tiga pada jurusan Manajemen Informatika, Fakultas Ilmu Komputer, Universitas Nasional Pasim, Bandung.
8. Manfaat bagi Persada Konveksi
9. Memudahkan Persada Konveksi dalam mengembangkan dan mengelola konveksi.
10. Adanya laporan pemesanan yang detail sehingga pemilik dapat memantau proses produksi dengan baik.
11. Bagi pelanggan
12. Memberikan informasi produk yang lengkap dan jelas kepada masyarakat.
13. Memberikan kemudahan dalam proses pemesanan bagi para *customer* yang ingin melakukan pemesanan.
14. Memberikan informasi mengenai progress produksi pesanan kepada *customer*.
    1. **Metode penelitian**
       1. **Metode Pengumpulan Data**
15. Wawancara

Penulis bertanya secara langsung kepada pihak pengelola konveksi mengenai proses pemesanan produk konveksi yang dilakukan saat ini.

1. Observasi

Penulis melakukan pengumpulan data dengan cara mempelajari dan mengamati objek secara langsung untuk mendapatkan bukti yeng mendukung penelitian.

1. Studi Pustaka

Penulis mengumpulkan informasi yang relevan dengan topik yang menjadi objek penelitian baik dari buku, karya ilmiah, internet dan sebagainya.

1. Studi Literatur

Studi Literatur adalah serangkaian kegiatan yang berkenaan dengan metode pengumpulan data Pustaka, membaca dan mencatat serta mengolah data penelitian.

* + 1. **Model Proses**

Model proses yang digunakan dalam pembangunan sistem ini adalah dengan menggunakan model proses *Relational Unified Process* (RUP). RUP adalah suatu kerangka kerja proses pengembangan perangkat lunak iteratif yang dibuat oleh *Relational Software*, suatu divisi dari IBM sejak 2003. RUP bukanlah suatu proses tunggal dengan aturan konkrit, melainkan suatu kerangka proses yang dapat diadaptasi dan dimaksudkan untuk disesuaikan oleh organisasi pengembang dan tim proyek perangkat lunak yang akan memilih elemen proses dengan kebutuhan mereka.

Dalam metode *Relational Unified Process* (RUP) memiliki empat tahap pengembangan perangkat lunak yaitu:

1. Permulaan (*Inception*)

Merupakan tahap untuk mengidentifikasi sitem yang akan dibangun. Aktivitas yang dilakukan pada tahap ini adalah mencakup analisis sistem *existing,* perumusan sistem target, penentuan arsitektur *global target,* identifikasi kebutuhan, perumusan persyaratan(fungsionl, performansi, keamanan, GUI, dll), perumusan kebutuhan pengujian (level unit, integrasi, sistem, performansi, fungsionalitas, keamanan, dll), UML diagram, dan pembuatan dokumentasi.

1. Elaboration

Merupakan tahap untuk melakukan desain secara lengkap berdasarkan hasil analisis pada tahap *inception*. Aktifitas yang dilakukan pada tahap ini antara lain mencakup pembuatan desain arsitektur subsistem, desain database, desain user interface, pemodelan diagram UML (diagram sequence, class, component, deployment, dll) dan pembuatan dokumentasi.

1. Construction

Merupakan tahap untuk mengimplementasikan hasil desain dan melakukan pengujian hasil implementasi. Aktivitas yang dilakukan pada tahap ini antara lain mencakup pengujian hasil analisis dan desain, pendataan kebutuhan implementasi lengkap (berpedoman pada identifikasi kebutuhan di tahap analisis), penentuan *coding pattern* yang digunakan, pembuatan program, pengujian, optimasi program, pendataan berbagai kemungkinan pengembangan atau perbaikan lebih lanjut, dan pembuatan dokumentasi.

1. Transition

Merupakan tahap untuk menyerahkan sistem aplikasi kepada user, yang umumnya mencakup pelatihan dan *beta testing* aplikasi.

* + 1. **Metode Pendekatan Sistem**

Metode pembangunan sistem informasi yang akan digunakan oleh penulis adalah dengan menggunakan pendekatan berorientasi objek dan analisis menggunakan *Unified Modelling Language* (UML).

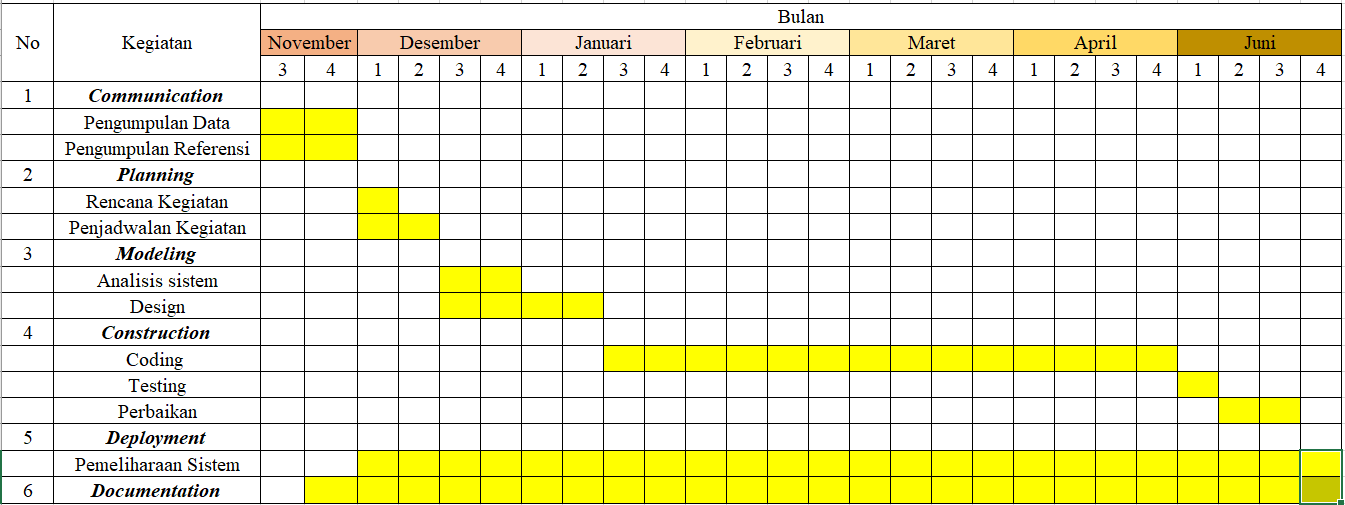
UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson.

UML menjadi alat yang akan membantu para developer dalam mengembangkan sistem informasi. Developer dapat dengan mudah membaca alur program yang akan dibangun.

Hasil analisis dengan pendekatan OOP (*Object Oriented Programming*) akan digambarkan oleh penulis dalam bentuk:

* Use Case Diagram
* Squence Diagram
* Class Diagram
* Activity Diagram
  1. **Lokasi Dan Waktu Penelitian**

Penulis melakukan penelitian di Konveksi Persada, yang beralamat di Tingkir Tengah, Kec. Tingkir, Kota Salatiga, Jawa Tengah 50746. Adapun rincian kegiatan penyusunan tugas akhir yang direncanakan penulis adalah sebagai berikut:



**Gambar 1.2** Jadwal kegiatan penelitian

**BAB 2**

**TINJAUAN PUSTAKA**

**2.1. Perancangan**

Perancangan merupakan penentuan proses dan data yang diperlukan oleh sistem baru. Perancangan sistem memberikan gambaran rancangan bangun yang lengkap sebagai pedoman bagi *programmer* dalam mengembangkan aplikasi.

Menurut Soetam Rizky (2011: 140) perancangan adalah sebuah proses untuk mendefinisikan suatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya.

**2.2. Sistem Informasi**

**2.2.1. Pengertian Sistem**

Menurut Azhar Susanto (2013: 22) sistem adalah kumpulan atau grup dari sub sistem/bagian/komponen atau apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan dapat bekerja sama untuk mencapai satu tujuan tertentu.

Menurut Sutarman (2009: 5) sistem adalah kumpulan elemen yang saling berinteraksi dalam kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama.

Menurut Jogiyanto (2009: 34) sistem dapat didefinisikan dengan pendekatan prosedur dan komponen. Sistem dan prosedur adalah suatu kesatuan yang tidak bisa dipisahkan satu dengan yang lain. Suatu sistem yang baru dapat terbentuk jika didalamnya ada beberapa prosedur yang mengikutinya.

Berdasarkan pendapat para ahli tersebut, dapat disimpulkan bahwa sistem adalah kumpulan dari setiap komponen sistem yang saling berhubungan untuk mencapai suatu tujuan.

**2.2.2. Informasi**

Informasi adalah sebuah fakta atau data yang telah diolah menjadi sesuatu yang berguna dan bernilai sehingga bermanfaat untuk penerima informasi.

Menurut Anggraeni dan Irviani (2017: 13) informasi adalah sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima.

Dalam Trimahardhika dan Sutinah (2017: 250) informasi merupakan suatu data yang telah diolah, diklasifikasikan dan diinterpretasikan serta digunakan untuk proses pengambilan keputusan.

**2.2.3. Pengertian Sistem Informasi**

Menurut John F. Nash (1995:8) Sistem Informasi adalah kombinasi dari manusia, fasilitas atau alat teknologi, media, prosedure dan pengendalian yang ditujukan untuk mengatur jaringan komunikasi yang penting, proses transaksi tertentu dan rutin, membantu manajemen dan pemakai intern dan ekstern dan menyediakan dasar untuk pengambilan keputusan yang tepat.

Menurut Rommey (1997:16) Sistem Informasi yang diselenggarakan cara untuk mengumpulkan, memasukan, mengolah, dan menyimpan data dan terorganisir cara untuk menyimpan, mengelola, mengendalikan dan melaporkan informasi dengan cara yang suatu organisasi dapat mencapai tujuan yang telah ditetapkan.

Menurut Kertahadi(2007) Sistem Informasi adalah alat untuk menyajikan informasi sedemikian rupa sehingga bermanfaat bagi penerimanya. Tujuannya adalah untuk memberikan informasi dalam perencanaan, memulai, pengorganisasian, operasional sebuah perusahaan yang melayani sinergi organisasi dalam proses mengendalikan pengambilan keputusan.

Berdasarkan pengertian menurut para ahli diatas dapat disimpulkan bahwa sistem informasi adalah alat yang mampu menghasilkan data yang telah diolah secara terorganisir untuk memberikan hasil output yang berguna bagi pengguna.

**2.3. Pemesanan Produk Konveksi**

**2.3.1. Pengertian pemesanan**

Pemesanan adalah proses yang biasa dilakukan antara konsumen dan produsen sebelum terjadinya pembelian barang oleh konsumen.

Menurut KBBI kata pemesanan mempunyai arti proses, perbuatan, cara memesan, atau memesankan.

Menurut Edwin dan Chris (1999:1) pemesanan dalam arti umum adalah perjanjian pemesanan tempat anatara 2 (dua) pihak atau lebih, perjanjian pemesanan tempat tersebut dapat berupa perjanjian atas pemesanan suatu ruangan, kamar, tempat duduk dan lainnya, pada waktu tertentu dan disertai dengan produk jasanya.

**2.3.2. Pengertian Produk**

Menurut Kotler dan Amstrong (2008:266) produk merupakan sebagai segala sesuatu yang dapat ditawarkan kepada pasar agar menarik perhatian, akuisi, penggunaan atau kosumsi yang dapat memuaskan suatu keinginan atau kebutuhan.

Menurut Staton (1996:222) produk adalah kumpulan dari atribut-atribut yang nyata maupun tidak nyata, termasuk di dalamnya kemasan, warna, hargaa, kualitas, dan merk ditambah dengan jasa dan reputaasi penjualannya.

Menurut Tjiptono (1999:95) secara konseptual produk adalah pemahaman subyektif dari produsen atas sesuatu yang bisa ditawarkan sebagai usaha untuk mencapai tujuan organisasi melalui pemenuhan kebutuhan dan keinginan konsumen, sesuai dengan kompetensi dan kapasistas organisasi serta daya beli.

Dari ketiga definisi produk tersebut dapat disimpulkan bahwa produk adalah segala sesuatu mempunyai warna, harga, kualitas dan merk yang ditawarkan dari produsen kepada konsumen sebagai usaha mencapai tujuan pemenuhan kebutuhan dan keinginan konsumen.

**2.3.3. Pengertian Konveksi**

Menurut Sri Wening dan Sicilia Savitri (1994:128) konveksi adalah usaha dibidang busana jadi yang dibuat secara besar-besaran.

Konveksi adalah industri kecil skala rumah tangga sebagai tempat pembuatan pakaian jadi seperti kemeja, kaos, jaket, seragam dan sebagainya. Barang diproduksi berdasarkan ukuan standar S, M, L, XL dalam jumlah yang banyak.­­­­­­­­

**2.4. Pengertian Sistem Informasi Pemesanan**

Sistem informasi pemesanan adalah alat untuk menyajikan informasi untuk pengambilan keputusan pada perencanaan, proses, pengendalian kegiatan, pembuatan, cara memesan yang bertujuan untuk mewujudkan kepuasan konsumen.

**2.5. Sistem Informasi Berbasis WEB**

**2.5.1. Pengertian WEB**

Dalam Wikipedia situs web / *website* adalah sekumpulan halaman web yang saling berhubungan yang umumnya berada pada peladen yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi.

Menurut Abdullah (2015:1) Website dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa text, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet.

Menurut Bekti (2015:35) Website merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terikat, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Penulis dapat menyimpulkan bahwa web adalah layanan berupa kumpulan halaman yang berisi informasi yang disediakan melalui koneksi internet.

**2.6. Perancangan Basis Data**

**1.6.1. Pengertian Data**

Menurut Bernard (2012:130) data adalah fakta kasar mengenai orang, tempat, kejadian dan sesuatu yang penting diorganisasikan.

Menurut Williams dan Sawyer(2007:25) data terdiri dari fakta – fakta dan angka – angka yang diolah menjadi informasi.

Menurut Inmon (2005) data adalah kumpulan dari fakta, konsep, atau instruksi pada penyimpanan yang digunakan untuk komunikasi, pernaikan dan diproses secara otomatis yang mempresentasikan informasi yang dapat dimengerti oleh manusia.

Dari pengertian di atas penulis dapat menyimpulkan, data adalah suatu fakta yang dapat diolah menjadi informasi yang berguna.

**1.6.2. Pengertian Basis Data**

Menurut Connolly (2002:14) database adalah sekumpulan data yang saling berhubungan secara logis dan dirancang untuk memenuhi kebutuhan informasi dalam suatu organisasi.

Menurut Kusrini (2007) basis data merupakan kumpulan data yang berelasi. Data sendiri merupakan fakta mengenai obyek, orang lain – lain. Data dinyatakan dengan nilai(deretan karakter, angka atau symbol).

Dari pengertian diatas penulis menyimpulkan bahwa basis data adalah kumpulan data yang saling berhubungan untuk memenuhi kebutuhan informasi.

**1.6.3. Database Management System**

Menurut Connoly, T.M. (2010) *database management system* (DBMS) adalah sistem perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara, dan mengontrol akses database.

Menurut Laudon, K.C. & Laudon, Jane.P. (2012) *database management system* (DBMS) adalah perangkat lunak yang memungkinkan sebuah organisasi untuk memusatkan data, mengelola secara efisien, dan memberikan akses ke data disimpan oleh program aplikasi.

Berdasarkan pengertian di atas, penulis dapat menyimpulkan bahwa database management system (DBMS) adalah sistem perangkat lunak uang digunakan untuk menyimpan dan mengelola data.

**2.7. Perangkat Desain**

**2.7.1. UML (*Unified Modeling Language*)**

Menurut Rosa dan Shalahudin (2015:133) UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang digunakan banyaak di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan design, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

Menurut Ariani R. Sukamto (2017) “UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram teks-teks pendukung”.

Berdasarkan pengertian di atas penulis dalam penulis dapat menyimpulkan bahwa UML (*Unified Modeling Language*) adalah bahasa visual untuk mendefinisikan *reqirement,* analisis dan design serta menggambarkan arsitektur dalam pemrograman berorientasi objek menggunakan diagram teks-teks pendukung.

**2.7.2. Use Case Diagram**

Menurut Rosa dan Shalahudin (2015: 155) **Use Case Diagram** adalahpemodelan untuk melakukan (behaviour) sistem informasi yang akan dibuat. Use Case diagram berfungsi untuk mengetahui apa saja yang ada di dalam sebuah sistem dan siapa saja yang yang berhubungan atau menggunakan sistem tersebut. Berikut symbol – symbol yang ada pada diagram *use case* (Rosa dan Shalahudin, 2014: 156):

**Tabel 2.1 Simbol-simbol Use Case Diagram**

|  |  |
| --- | --- |
| Simbol | Deskripsi |
| Use Case | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama use case. |
| Aktor | Proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dai aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal *frase* nama aktor. |
|  | Komunikasi antara aktor dan *use cas*e yang berpartisispasi pada *use case* atau *use case* memiliki interaksi dengan aktor. |
| Ekstensi / *extend* | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan depat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip inheritance pada pemrograman brorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan. |
| Generalisasi | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |

**2.7.3. Sequence Diagram**

Menurut Sukamto & Shalahuddin (2015:165) Diagram Sequence menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

Menurut Satzinger (2011:26) Sequence Diagram merupkan diagram yang menunjukkan aliran fungsionalitas dalam use case. Sequence adalah satu dari dua interaksi diagram yang mengilustrasikan objek – objek yang berhubungan dengan use case dan message atau pesan – pesannya.

Berikut symbol – symbol yang ada pada diagram *use case* (Rosa dan Shalahudin, 2014: 156):

**2.2. Simbol-simbol Sequence Diagram**

|  |  |
| --- | --- |
| Simbol | Deskripsi |
| Aktor | Proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dai aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal *frase* nama aktor. |
| Garis hidup | Menyatakan kehidupan suatu objek |
| Objek  Nama objek | Menyatakan objek yang berinterkasi pesan. |
| Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. |
| Pesan tipe *create* | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. |
| Pesan tipe *call* | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri. Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi atau metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi. |
| Pesan tipe *send* | Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi. |
| Pesan tipe *return* | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian. |
| Pesan tipe *destroy* | Menyatakan bahwa suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada *create* maka ada *destroy.* |

**2.7.4. Activity Diagram**

Diagram aktivitas atau *activity diagram* adalah diagram yang menggambarkan aliran kerja suatu sistem beserta relasi yang terkait dalam suatu sistem tersebut.

Berikut adalah symbol – symbol yang ada pada diagram aktivitas (Rosa dan Shalahuddin 2013: 162-163):

**2.2. Simbol-simbol Sequence Diagram**

|  |  |
| --- | --- |
| Simbol | Deskripsi |
| Status awal | Status awal aktivitas sistem, sebuah diagram memiliki sebuah status awal. |
| Aktivitas | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| Percabangan / *Decision* | Asosiasi percabangan dimana jika ada satu pilihan aktivitas atau lebih dari satu. |
| Penggabungan / *Join* | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| Status akhir | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuahstatus akhir |
| *Swimlane* | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

**2.7.5. Class Diagram**

Menurut Rosa dan Shalahuddin (2015: 141) diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Berikut adalah symbol-simbol yang ada pada *class diagram* (Rosa dan Shalahuddin, 2015: 146)

|  |  |
| --- | --- |
| Simbol | Deskripsi |
| Kelas | Kelas pada struktur sistem |
| Antarmuka / *interface* | Sama dengan *interface* dalam pemrograman berorientasi objek (OOP) |
| Asosiasi | Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity* |
| Asosiasi berarah | Relasi antarkelas dengan makna kelas satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.* |
| Generalisasi | Relasi antar kelas dengan makna generalisasi spesialisasi (umum-khusus). |
| Kebergantungan | Relasi antarkelas dengan makna kebergantungan antarkelas. |
| agregasi | Relasi antarkelas dengan makna semua bagian (*whole-part*). |

**2.8. Perangkat Pemrograman**

**2.8.1. Bahasa Java**

Menurut definisi *Sun Microsystem*, di dalam buku M. Shalahuddin dan Rosa A.S. (2010: 1) Java adalah nama sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada computer yang berdiri sendiri (standalone) ataupun pada lingkungan jaringan.

Menurut Arie (2010: 1) *Java* merupakan perangkat lunak produksi Sun Microsystem Inc. Untuk pemrograman beberapa tujuan (*multipurpuse*), dapat berjalan di beberapa sistem operasi (*multiplatform*), mudah dipelajari dan powerfull.

Java memiliki tiga komponen penting yaitu:

1. Java Development Kit (JDK)

Sebagai komponen inti dalam java. Komponen ini menyediakan semua tools, executables, binaries yang diperlakukan untuk Menyusun, mendebug, dan mengeksekusi sebuah program java.

1. Java Virtual Machine (JVM)

Java Virtual Machine bertugas untuk mengkonversi *byte code* menjadi kode yang lebih spesifik.

1. Java Runtime Environment (JRE)

Java Runtime Environment merupakan kumpulan lingkungan untuk menjalankan program java.



**Gambar 2.1** Logo Java

**2.8.2. Framework Spring**

Framework Spring merupakan framework open source berbasis Java yang menyediakan infrastruktur yang komprehensif dalam mengembangkan aplikasi Java dengan mudeh dan cepat. Framework Spring pertama kali ditulis dan dirilis olej Rod Johnson dengan lisansi Apache 2.0 pada bulan Juni 2003. Spring dapat membantu programmer untuk mengembangkan aplikasi sistem berbasis JVM.

Beberapa fitur yang disediakan Spring Framework adalah sebagai berikut:

1. Container atau disebut juga Dependency Injection
2. Basis Web
3. Transaction Management
4. JDBC Exception Handling
5. Object Relational Mapping (ORM)

**2.8.3. HTML, CSS dan Javascript**

1**.** HTML (*HyperText Markup Language*)

Menurut Sibero (2013: 19) *HyperText Markup Language* atau HTML adalah bahasa yang digunakan pada dokumen web sebagai bahasa untuk pertukaran document *web*.

Menurut Arief (2011: 23) HTML atau *HyperText Markup Language* merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan dihalaman *web*.

Menurut Nugroho (2013: 5) HTML adalah kependekan dari (*HyperText Markup Language*), merupakan sebuah bahasa Scripting yang berguna untuk menuliskan halaman Web.

Dari beberapa pendapat diatas dapat disimpulkan bahwa HTML (*HyperText Markup Language*) adalah salah satu format bahasa yang digunakan untuk membuat dokumen dan aplikasi halaman web.

2. CSS (Cascading Style Sheet)

Menurut Prasetio (2014: 252) menyatakan bahwa CSS adalah salah satu teknologi yang digunakan untuk memperindah tampilan halaman website (situs).

Menurut Sulistiyawan, dkk (2008: 32) mengemukakan bahwa *cascading style sheet* adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur style suatu dokumen.

Berdasarkan pengertian diatas penulis dapat menyimpulkan bahwa *cascading style sheet* (CSS) adalah sebuah teknologi yang digunakan untuk memperindah dan mengatur tampilan suatu dokumen.

**2.8.4. Database MySQL**

Database atau basis data adalah sekumpulan informasi yang disimpan di dalam komputer secara sistematik yang digunakan untuk memperoleh informasi dari basis data tersbut. Database mempunyai beberapa fungsi sebagai berikut:

1. mengelompokkan data untuk mempermudah indentifikasi data
2. Menghindari redudansi/pengulangan data
3. Mempermudah akses, penyimpanan data, mengubah dan menhapus data

**2.8.5. Xampp Control Panel**

Xampp adalah web server bersifat open source yang bisa digunakan pada sistem operasi cross-platform seperti Windows, Linux, Solaris, dan juga MacOS. XAMPP merupakan singkatan dari X (cross platform), A (Apache), M (MySQL), P (PHP), dan P(Perl) yang merupakan program – program tersedia didalamnya.

**BAB III**

**OBJEK PENELITIAN DAN ANALISIS SISTEM**

**3.1. Sejarah**

Persada Konveksi adalah bidang usaha yang bergerak dalam bidang industri konveksi di Kota Salatiga. Persada Konveksi didirikan oleh Ibu Hanik dan Ibu Ana pada tanggal 10 Mei 2007. Mereka merupakan saudara kandung yang bersama sama membangun usaha konveksi untuk mendukung produk lokal yang diminati oleh penduduk Indonesia. Selain pemasaran produk di Jawa, terutama Salatiga dan Kabupaten Semarang, mereka juga memasarkan produk konveksi ke beberapa pulau selain lain, yaitu Kalimantan, Sulawesi, dan Sumatera. Selain itu, mereka juga mempunyai tujuan agar mereka mampu memberikan peluang pekerjaan untuk masyarakat sekitar.

Dalam perkembangannya, Persada Konveksi menerima layanan untuk pembuatan seragam, jaket, kaos, pakaian olahraga dan jasa border.Dari tahun ke tahun Persada Konveksi dapat berkembang dengan baik dan jangkauan pemsaran semakin luas. *Customer* Persada Konveksi berasal dari berbagai kalangan masyarakat diantaranya perorangan, sekolah, organisasi siswa sekolah, organisasi mahasiswa, yayasan dll.

Di area Salatiga terdapat banyak konveksi lain yang juga berjalan dalam bidang industri konveksi, hal itu menyebabkan persaingan konveksi disini sangat ketat, sehingga setiap konveksi mempunyai tuntutan untuk memberikan pelayanan dan hasil produksi yang berbeda. Persada Konveksi memberikan jaminan bahwa produk yang mereka buat adalah produk yang mampu bersaing dalam hal kualitas dan kuantitas. Selain itu Persada Konveksi juga menerima pesanan sesuai keinginan konsumen sehingga memberikan hasil yang yang baik untuk *customer* .

**3.2. Visi dan Misi Perusahaan**

**3.2.1. Visi**

Visi yang dimiliki oleh Persada Konveksi adalah menjadi perusahaan terkemuka yang terpercaya dengan kualitas dan pelayanan terbaik untuk konsumen,

**3.2.2. Misi**

a. Memberikan layanan dan produk terbaik kepada pelanggan konveksi

b. Memberikan produk yang inovatif kepada pelanggan

c. Menciptakan tata kelola konveksi yang efektif dan efisien

d. Mensejahterkan karyawan

**3.3. Struktur Organisasi Persada Konveksi**

Struktur organisasi merupakan komponen-komponen dalam organisasi atau perusahaan yang berfungsi sebagai pengelompokan tugas pada anggota sesuai keahlian masing-masing anggota untuk mencapai hasil yang diinginkan dalam sebuah organisasi.

Struktur organisasi Persada Konveksi dapat dilihat dari gambar dibawah ini:

Struktur Organisasi Persada Konveksi

Pemimpin Perusahaan

Bagian Administrasi

Bagian Produksi

Bagian Pemasaran

Finishing

Penjahit

Bagian Bordir

**Gambar 3.1** Struktur Organisasi Persada Konveksi

**3.3.1. Tugas dan Wewenang**

a. Tugas dan Wewenang Pemimpin Perusahaan

1. Merencanakan masa depan perusahaan
2. Mengambil Keputusan
3. Menentukan harga
4. Melakukan pemasaran dan perekrutan karyawan dan menentukan tarif gaji karyawan
5. Melakukan kegiatan yang berkaitan dengan produsen dan mitra kerja.

b. Tugas Pemasaran

1. menerima pesanan
2. menawarkan produk kepada konsumen

c. Tugas Produksi

1. membuat pola dan model pakaian
2. memotong kain
3. menjahit kain menjadi pakaian sesuai pesanan

d. Finishing

1. merapikan pakaian setelah produksi
2. menyetrika dan mengemas pakaian

**3.4. Analisis Sistem**

Menurut Yakub (2012:142) Analisis Sistem adalah sebagai suatu proses untuk memahami sistem yang ada, dengan menganalisa jabatan dan uraian tugas, proses bisnis, ketentuan atau aturan, masalah dan mencari solusinya.

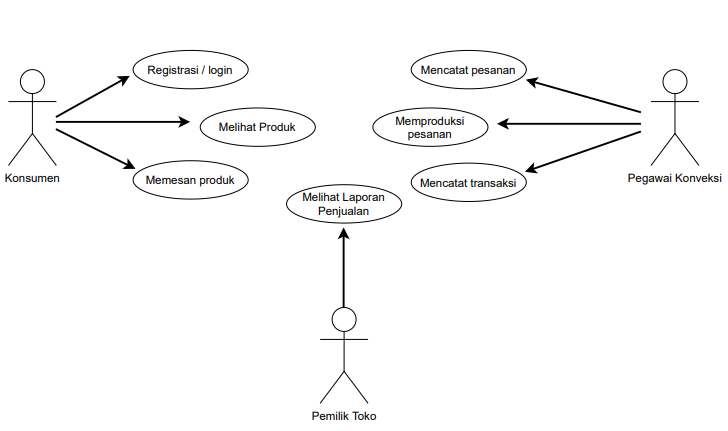
**3.4.1. Analisis Sistem Berjalan**

Berikut prosedur sistem berjalan pada Persada Konveksi:

1. Pelanggan mengunjungi konveksi untuk melihat produk yang sudah ada, pegawai konveksi akan memberikan informasi mengenai harga dan detail produk, pelanggan juga dapat memesan produk sesuai keinginan pelanggan sendiri dengan menjelaskan detail produk kepada pemimpin konveksi.
2. Setelah pelanggan menentukan produk yang diinginkan, pegawai konveksi mencatat pesanan pelanggan. Dan memberikan kwitansi total yang harus dibayar oleh pelanggan dengan ketentuan dp sebesar 50% terlebih dahulu dari total harga barang yang dipesan.
3. Setelah pesanan sudah selesai diproduksi pegawai konveksi akan menginformasikan kepada pelanggan bahwa pesanan sudah bisa diambil. Pelanggan datang ke toko untuk mengambil pesanan dan membayar sisa jumlah harga yang harus dilunasi dengan membawa kwitansi sebelumnya.
4. Pegawai konveksi mencatat transaksi penjualan ke dalam rekap penjualan yang sudah lunas.

**3.4.1. Use Case Sistem Berjalan**

Analisis system berjalan Konveksi Persada dalam Use Case Diagram adalah sebagai berikut:

****

**Gambar 3.2** *Use Case* Sistem yang Berjalan

**3.4.2. Analisis Kelemahan Sistem Berjalan**

1. *Performance*

a. Kelemahan sistem lama:

Pencatatan pemesanan pelanggan masih dilakukan secara manual sehingga rentan terjadi kesalahan dan kemungkinan kehilangan catatan juga besar. Pencatatan secara manual juga menyebabkan sulit untuk merekap penjualan.

b. Sistem yang akan diajukan:

Pelanggan memasukan pemesanan beserata detail pesanan ke dalam perangkat lunak yang akan disediakan sehingga pegawai tidak perlu mencatat detail pesanan secara manual.

2. *Information*

1. Kelemahan sistem lama:

Sistem yang berjalan pada Persada Konveksi masih menggunakan penjualan konvensional dan belum terpublikasi secara luas karena belum menerapkan sistem online sehingga pelanggan harus mendatangi konveksi untuk mendapatkan informasi mengenai ketentuan pemesanan produk, jika melalui pesan seperti SMS dan telepon informasi yang diberikan biasanya kurang maksimal.

Selain itu bagi pegawai konveksi harus mencari catatan pesanan dalam buku pesanan ketika ada informasi yang dibutuhkan.

1. Sistem yang akan diajukan:

Pelanggan dapat melihat detail produk, dapat mengetahui jika konveksi mempunyai produk terbaru. dan melakukan pemesanan dengan mudah di dalam perankat lunak.

Selain itu rekap pemesanan dan penjualan tercatat secara detail di dalam perangkat lunak.

3. *Economic*

a. Kelemahan sistem lama:

Pencatatan terhadap pesanan dan penjualan masih dilakukan secara manual dan membutuhkan buku – buku. Catatan dalam buku rentan terjadi kerusakan atau kehilangan dan sering membutuhkan buku pengganti.

1. Sistem yang akan diajukan:

Semua data tersimpan di dalam database dan dapat di buka sewaktu – waktu jika diperlukan

4. *Control*

a. Kelemahan sistem lama:

Pelanggan harus menanyakan secara langung kepada pegawai konveksi mengenai *progress* produksi pesanan. Dan pemilik / pegawai juga harus melakukan pengecekan dengan mengecek langsung pada bagian produksi, hal tersebut akan kurang efektif jika dilakukan secara berulang – ulang.

b. Sistem yang diajukan

Kontrol pada proses produksi pesanan dapat dilihat melalui perangkat lunak. baik oleh pelanggan maupun pegawai konveksi.

5. *Efficiency*

a. Kelemahan sistem lama

Pegawai konveksi harus berulang - ulang berdiskusi untuk menjelaskan detail produk kepada setiap konsumen yang datang, hal tersebut akan menyita waktu pegawai untuk mengerjakan pekerjaan produksi pesanan.

b. Sistem yang diajukan

Konsumen dapat melihat detail produk yang disediakan oleh konveksi melalui sistem perangkat lunak, jika terdapat hal yang kurang jelas baru dapat dikonsultasikan dengan pegawai, sehingga pegawai dapat memaksimalkan dalam proses pengerjaan produksi.

6. *Service*

a. Kelemahan sistem lama

Persada Konveksi hanya melayani pesanan bagi yang datang mengunjungi konveksi.

b. Sistem yang di ajukan

Pelanggan dapat memesan secara online melalui perangkat lunak tanpa harus mengunjungi konveksi secara langung.

**3.5. Analisis Kebutuhan Sistem**

Analis sistem dapat diartikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam komponen-komponen pembentuknya yang lebih sederhana untuk mengetahui bagaimana komponen-komponen tersebut bekerja sama dan saling berinteraksi untuk mencapai tujuan sistem.

**3.5.1. Analisis Kebutuhan Fungsional**

Analisis kebutuhan fungsional merupakan analisis yang dilakukan untuk memberikan gambaran mengenai kebutuhan, permasalahan, dan prosedur yang sedang berjalan. Adapun analisis kebutuhan sistem secara fungsional Konveksi Persada adalah sebagai berikut:

1. Sistem dapat diakses oleh 2 user yaitu pelanggan dan pegawai konveksi / pemilik konveksi.
2. Sistem aplikasi yang dibangun menyajikan rekap penjualan yang dapat diakses oleh pemilik konveksi.
3. Sistem aplikasi yang dibangun dapat menampilkan data produk, data pesanan, penjualan, dan pelanggan.
4. Sistem aplikasi dapat menampilkan informasi mengenai presentasi proses pengerjaan pesanan untuk pelanggan dan dapat di ubah oleh pegawai konveksi jika terdapat peningkatan proses pengerjaan.
5. Aplikasi yang dibangun dapat melakukan validasi terhadap penginputan data produk, data pemesanan, data penjualan, dan data pelanggan.

**3.5.2. Analisis Kebutuhan Non-Fungsional**

Analisis kebutuhan non-fungsional dilakukan untuk mengetahui spesifikasi kebutuhan sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras atau hardware, analisis perangkat lunak atau software serta analysis pengguna atau user.

**3.5.2.1. Analisis Kebutuhan Perangkat Keras / *Hardware***

Untuk merancang dan membuat media informasi berbasis web dibutuhkan perangkat keras agar program aplikasi yang dibuat dapat berjalan degan baik. Spesifikasi yang digunakan adalah PC dengan spesifikasi sebagai berikut:

1. Programmer
2. AMD A9-9425 RADEON R5, 5 COMPUTE CORES 2C+3G 3.10 GHz
3. 4.00 GB (3.89 GB usable)
4. VGA Resolusi minimal 512 MB
5. Monitor 1024 x 768
6. Keyboard, Mouse, Printer standard

**3.5.2.2. Analisis Kebutuhan Perangkat Lunak / Software**

Spesifikasi perangkat lunak yang dibutuhkan untuk mendukung aplikasi yang akan dibangun yaitu sebagai berikut:

1. Operasional

* Minimal menggunakan sistem operasi windows 7
* Google Chrome

1. *Sofware* Pendukung Pembangunan Sistem

* XAMPP
* SQLyog Enterprise

1. Bahasa atau *Scripting*

* Java untuk pembuatan aplikasi web
* HTML untuk pembuatan desain interface aplikasi web
* CSS untuk memperinda desain interface aplikasi web
* SQL *language* untuk memenipulasi data dalam database

1. Keamanan

* Sistem dilengkapi dengan form login untuk menyaring hak akses

**BAB IV**

**PERANCANGAN SISTEM**

**4.1. Perancangan Sistem**

Perancangan sistem adalah Menyusun atau mendesain suatu sistem yang berisi Langkah-langkah operasi dalam proses pengolahan data dan proses prosedur-prosedur untuk mendukung operasi sistem. Tujuan dari perancangan sistem adalah memetakan semua kebutuhan sistem sehingga sistem berjalan dengan baik, memenuhi kebutuhan para pemakai sistem seta memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada para programmer dan para ahli yang terlibat dalam pengembangan sistem.

Perancangan sistem ini mencakup *use case diagram*, *activity diagram, sequence diagram,* dan *class diagram* yang depat menjelaskan aliran data yang diproses hingga menghasilkan informasi yang diinginkan.

**4.1.1. Identifikasi Aktor**

Pada sistem Informasi pemesanan Persada Konveksi ini terdapat 3 pengguna atau actor. Adapun penjeasan dari pengguna atau actor adalah sebagai berikut:

**Tabel 4.1** Indentifikasi Aktor

|  |  |  |
| --- | --- | --- |
| **No.** | **Aktor** | **Deskripsi** |
| 1 | Pegawai (Administrator) | Pengguna yang mempunyai hak akses untuk mengelola *website* secara keseluruhan. Pengguna ini dapat mengakses setiap data master, seperti data produk, data pelanggan, data pemesanan, data transaksi dan data penjualan. |
| 2 | Pemilik | Pemilik memiliki hak akses untuk melihat data pelanggan, data pemesanan, data penjualan, serta mencetak atau mengunduh laporan penjualan. |
| 3 | Pelanggan | Pelanggan dapat melihat produk, dan melakukan pemesanan, |

**4.1.2. Identifikasi Use Case**

Pada Sistem Informasi Pemesanan Konveksi Persad aini terdapat beberapa *use case* untuk mengetahui fungsi dalam sistem yang akan dibangun. Adapun penjelasan dari *use case* tersebut adalah sebagai berikut:

**Table 4.2.** Identifikasi *Use Case*

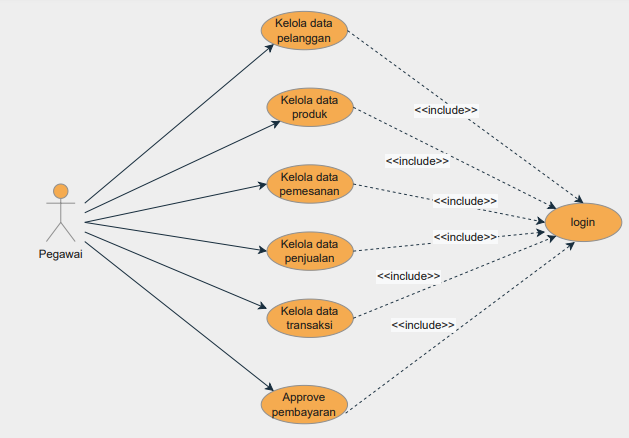
|  |  |  |
| --- | --- | --- |
| **No.** | **Aktor** | **Deskripsi** |
| 1 | Registrasi | [Pemilik, Pegawai, Pelanggan] Aktivitas yang diperlukan untuk membuat akun agar dapat mengakses aplikasi. |
| 2 | Login | [Pemilik, Pegawai, Pelanggan] Aktivitas untuk mengatur hak akses pengguna dan sebagai validasi untuk mendapatkan hak akses sistem. |
| 3 | Kelola data pelanggan | [Pegawai] Aktivitas untuk mengelola data pelanggan baik aksi tambah, ubah dan hapus data pelanggan. |
| 4 | Kelola data produk | [Pegawai] Aktivitas untuk mengelola data produk baik aksi tambah, ubah dan hapus data produk. |
| 5 | Kelola data pemesanan | [Pegawai] Aktivitas untuk mengelola data pemesanan baik aksi tambah, ubah dan hapus data. |
| 6 | Kelola data transaksi | [Pegawai] Aktivitas untuk mengelola data transaksi untuk aksi tambah dan ubah data transaksi. |
| 7 | Lihat Produk | [Pelanggan] Aktivitas untuk melihat produk |
| 8 | Lihat Detail Produk | [Pelanggan] Aktivitas untuk melihat detail produk |
| 9 | Approve Pembayaran | [Pegawai] Aktivitas untuk menyetujui bahwa pelanggan telah melakukan pembayaran |
| 14 | Melihat laporan penjualan | [Pemilik] Aktivitas untuk memeriksa penjualan konveksi |
| 15 | Kelola data pegawai | [Pemilik] Aktivitas untuk mengelola data pegawai untuk aksi tambah dan ubah data. |

**4.2. Metode Perancangan Sistem**

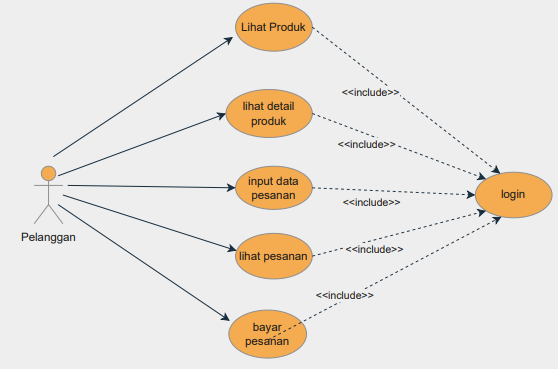
Untuk membantu proses perancangan sistem berorientasi objek, maka digunakan 4(empat) diagram yaitu *use case diagram*, *activity diagram, sequence diagram* dan *class diagram* sebagai berikut:

**4.2.1. *Use Case***

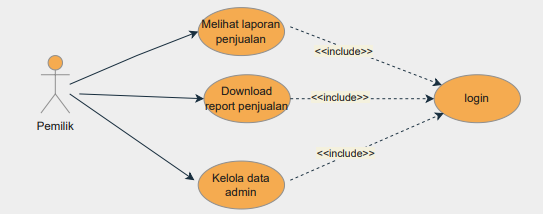
**4.2.1.1. Use Case Diagram**

****

**Gambar 4.1** Use Case Pegawai



**Gambar 4.2** Use Case Pelanggan

****

**Gambar 4.3** Use Case Pemilik

* + - 1. ***Use Case Spesification***

1. *Use Case Spesification Login*

**Tabel 4.3** *Use Case Spesification* *Login*

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Login | |
| *Actor* | Pemilik konveksi, Pegawai, dan pelanggan | |
| *Description* | Dalam proses ini sistem melakukan proses authentication dengan menyesuaikan *username* dan *password* yang dimasukkan oleh user dengan data berada di dalam database. Dan sistem juga melakukan proses *authorization*/menentukan hak akses yang dimiliki oleh user. | |
| *Pre-condition* | Pengguna dalam keadaan logout dan sistem menampilkan *form login* | |
| *Basic Flow* | Aktor | Sistem |
| 1. *Input username* dan *password* | Sistem melakukan validasi *username* dan *password* yang dimasukkan user |
| Alternative Flow | Jika Username dan password salah maka sistem akan menampilkan pesan “Username dan Password salah!”  Jika Username atau password kosong maka sistem akan menampilkan pesan “Username dan Password tidak boleh kosong” | |
| *Requirement* | Berhasil login sebagai pegawai atau pelanggan | |
| *Post-condition* | Sistem menampilkan halaman utama | |

2. *Use Case Spesification* Kelola Data Produk

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Kelola Data Produk | | |
| *Actor* | Pegawai | | |
| *Description* | *Use Case* ini digunakan untuk mengelola data produk yang dapat diproduksi oleh konveksi. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, ubah dan hapus data produk. | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. Klik *button* data produk | 1. Sistem menampilkan halaman data produk |
|  | Tambah | 1. Klik *button* tambah | 1. Sistem menampilkan form tambah data |
|  |  | 1. Mengisi *form* tambah data dan klik *button save* | 1. Sistem melakukan validasi pada data yang diinputkan sebelum data dimasukan di dalam *database*. |
|  | Ubah | 1. Klik *button* data produk | 1. Menampilkan *form* yang berisi detail data yang akan diubah. |
|  |  | 1. Menginputkan perubahan data kemudian klik *button save* | 1. Sistem melakukan validasi data yang telah diinputkan user dan menyimpan data ke dalam *database.*   (*Alternatif flow* 6.1) |
|  | Hapus | 1. Klik button hapus pada tampilan. | 1. Sistem akan menampilkan konfirmasi hapus data |
|  |  | 1. Klik ok pada layer konfirmasi hapus data. (*alternatif flow* 5.1) | 1. Sistem akan menghapus data |
| *Alternative Flow* | 5.1 Mengklik button cancel (Kembali ke Langkah 2)  6.1 Data yang diinputkan tidak sesuai dengan peraturan sistem (Kembali ke langkah 4) | | |
| *Requirement* | Berhasil login sebagai pegawai | | |
| *Post-condition* | Sistem melakukan Data Manipulation Language (DML) pada database sesuai dengan aksi yang dilakukan aktor (tambah, ubah dan hapus data) | | |

3. *Use Case Spesification* Kelola Data Pesanan

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Kelola Data Pesanan | | |
| *Actor* | Pegawai dan Pelanggan | | |
| *Description* | *Use Case* ini digunakan untuk mengelola data pesanan yang diinputkan oleh pelanggan dan pesanan yang diterima oleh konveksi. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, ubah dan hapus data pesanan. | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. Klik *button* data pesanan | 1. Sistem menampilkan halaman data pesanan |
|  | Tambah | 1. Klik *button* pesan | 1. Sistem menampilkan form tambah data untuk pesanan |
|  |  | 1. Mengisi *form* tambah data pesanan dan klik *button* pesan | 1. Sistem melakukan validasi pada data yang diinputkan sebelum data dimasukan di dalam *database*. |
|  | Ubah | 1. Klik *button* data pesanan | 1. Menampilkan *form* yang berisi detail data yang akan diubah. |
|  |  | 1. Menginputkan perubahan data kemudian klik *button save* | 1. Sistem melakukan validasi data yang telah diinputkan user dan menyimpan data ke dalam *database.*   (*Alternatif flow* 6.1) |
|  | Hapus | 1. Klik button batalkan pesanan pada tampilan. (*Alternatif flow* 3.1) | 1. Sistem akan menampilkan konfirmasi hapus pesanan |
|  |  | 1. Klik ok pada layar konfirmasi hapus data. (*alternatif flow* 5.1) | 1. Sistem akan menghapus data |
| *Alternative Flow* | 3.1. Pesanan hanya dapat dibatalkan oleh admin dengan menghapus data pesanan.  5.1 Mengklik button cancel (Kembali ke Langkah 2)  6.1 Data yang diinputkan tidak sesuai dengan peraturan sistem (Kembali ke langkah 4) | | |
| *Requirement* | Berhasil login sebagai pegawai atau pelanggan | | |
| *Post-condition* | Sistem melakukan Data Manipulation Language (DML) pada database sesuai dengan aksi yang dilakukan aktor (tambah, ubah dan hapus data) | | |

4. *Use Case Spesification* *Approve* Pembayaran

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Approve Pembayaran | | |
| *Actor* | Pegawai | | |
| *Description* | *Use Case* ini digunakan oleh pegawai untuk menyetujui bahwa pelanggan telah melakukan pembayaran. | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. Klik *button edit* pada menu transaction   (*alternatif* 1.1) | 1. Sistem menampilkan form transaksi pesanan. |
|  |  | 1. Pilih status transaksi menjadi *approve.* | 1. Sistem menampilkan detail transaksi pesanan dimana status telah berubah menjadi *approved*. |
| *Alternative Flow* | 1.1 Klik *button reject* dan menambahkan catatan | | |
| *Requirement* | User login sebagai pegawai | | |
| *Post-condition* | Sistem melakukan Data Manipulation Language (DML) pada database sesuai dengan aksi yang dilakukan aktor (ubah data) | | |

5. *Use Case Spesification* Lihat Produk

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Lihat Produk | | |
| *Actor* | Pelanggan | | |
| *Description* | *Use Case* ini digunakan oleh pelanggan untuk melihat produk | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. User klik *button* produkpada *navbar* | 1. Sistem menampilkan data produk yang disediakan konveksi |
| *Alternative Flow* | - | | |
| *Requirement* | User login sebagai pelanggan | | |
| *Post-condition* | Data produk berhasil ditampilkan | | |

6. *Use Case Spesification* Lihat Detail Produk

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Lihat Detail Produk | | |
| *Actor* | Pelanggan | | |
| *Description* | *Use Case* ini digunakan oleh pelanggan untuk melihat detail produk | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. User klik *button* produkpada *navbar* | 1. Sistem menampilkan data produk yang disediakan konveksi |
|  |  | 1. User mengklik *link* detail produk | 1. Menampilkan halaman berisi detail produk |
| *Alternative Flow* | - | | |
| *Requirement* | User login sebagai pelanggan | | |
| *Post-condition* | Detail produk berhasil ditampilkan | | |

7. *Use Case Spesification* Input Data Pesanan

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Input Data Pesanan | | |
| *Actor* | Pelanggan | | |
| *Description* | *Use Case* ini digunakan oleh pelanggan untuk Menginputkan detail pesanan produk | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. User klik *button* produkpada *navbar* | 1. Sistem menampilkan data produk yang disediakan konveksi |
|  |  | 1. User mengklik *button* detail produk | 1. Menampilkan halaman berisi detail produk |
|  |  | 1. *User* mengklik *button* buatpesanan | 1. Menampilkan form detail pesanan |
|  |  | 1. *User* menginputkan detail pesanan dan klik pesan | 1. Melakukan validasi pada data yang telah dimasukan *user* sebelum dimasukan ke dalam database.(Alternatif flow 8.1) |
| *Alternative Flow* | 8.1 Data yang diinputkan tidak sesuai dengan *requirements* sistem(Kembali ke Langkah 6) | | |
| *Requirement* | User login sebagai pelanggan | | |
| *Post-condition* | Berhasil menyimpan data pesanan ke dalam database | | |

8. *Use Case Spesification* Lihat *Progress* pesanan

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Lihat *Progress* pesanan | | |
| *Actor* | Pelanggan | | |
| *Description* | *Use Case* ini digunakan oleh pelanggan untuk melihat *progress* pesanan yang berjalan. Informasi yang diberikan berupa presentase. | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. User klik pesanan saya | 1. Sistem menampilkan data pesanan pelanggan |
|  |  | 1. User mengklik *button* detail pesanan | 1. Menampilkan halaman berisi detail pesanan dan infomasi *progress* pesanan |
| *Alternative Flow* | - | | |
| *Requirement* | User login sebagai Pelanggan | | |
| *Post-condition* | *Progress* pesanan berhasil ditampilkan | | |

9. *Use Case Spesification* Lihat Laporan Penjualan

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Lihat Laporan Penjualan | | |
| *Actor* | Pemilik | | |
| *Description* | *Use Case* ini digunakan oleh pemilik konveksi untuk melihat laporan penjualan | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. User klik *button* penjualanpada *navbar* | 1. Sistem menampilkan data penjualan. |
|  |  | 1. User mengklik *button* detail penjualan | 1. Menampilkan halaman berisi detail penjualan |
| *Alternative Flow* | - | | |
| *Requirement* | User login sebagai pemilik | | |
| *Post-condition* | Data penjualan berhasil ditampilkan | | |

10. *Use Case Spesification* Kelola Data Pegawai

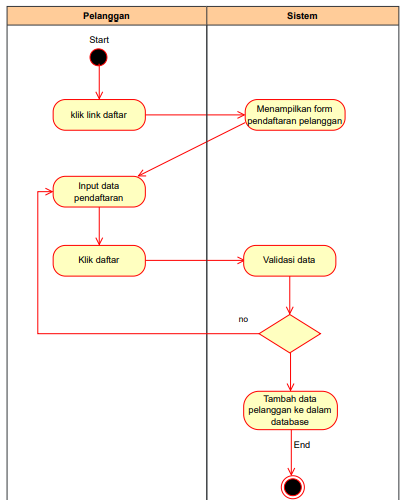
|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Kelola Data Pegawai | | |
| *Actor* | Pegawai | | |
| *Description* | *Use Case* ini digunakan untuk mengelola data pegawai Persada Konveksi. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, ubah dan hapus data pegawai. | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. Klik *button* data Pegawai | 1. Sistem menampilkan halaman data Pegawai |
|  | Tambah | 1. Klik *button* tambah | 1. Sistem menampilkan form tambah data |
|  |  | 1. Mengisi *form* tambah data dan klik *button save* | 1. Sistem melakukan validasi pada data yang diinputkan sebelum data dimasukan di dalam *database*. |
|  | Ubah | 1. Klik *button* ubahdata pegawai | 1. Menampilkan *form* yang berisi detail data yang akan diubah. |
|  |  | 1. Menginputkan perubahan data kemudian klik *button save* | 1. Sistem melakukan validasi data yang telah diinputkan user dan menyimpan data ke dalam *database.*   (*Alternatif flow* 6.1) |
|  | Hapus | 1. Klik button hapus pada tampilan. | 1. Sistem akan menampilkan konfirmasi hapus data |
|  |  | 1. Klik ok pada layer konfirmasi hapus data. (*alternatif flow* 5.1) | 1. Sistem akan menghapus data |
| *Alternative Flow* | 5.1 Mengklik button cancel (Kembali ke Langkah 2)  6.1 Data yang diinputkan tidak sesuai dengan peraturan sistem (Kembali ke langkah 4) | | |
| *Requirement* | Berhasil login sebagai pegawai | | |
| *Post-condition* | Sistem melakukan Data Manipulation Language (DML) pada database sesuai dengan aksi yang dilakukan aktor (tambah, ubah dan hapus data) | | |

11. *Use Case Spesification* Kelola Data Pelanggan

|  |  |  |  |
| --- | --- | --- | --- |
| *Use Case Name* | Kelola Data Pelanggan | | |
| *Actor* | Pegawai | | |
| *Description* | *Use Case* ini digunakan untuk mengelola data pelanggan Persada Konveksi. Fungsi-fungsi yang ada di dalam *use case* ini yaitu tambah, ubah dan hapus data pegawai. | | |
| *Pre-condition* | *User* dalam keadaan login | | |
| *Basic Flow* | Aksi | Aktor | Sistem |
|  | 1. Klik *button* data pelanggan | 1. Sistem menampilkan halaman data pelanggan |
|  | Tambah | 1. Klik *button* tambah | 1. Sistem menampilkan form tambah data |
|  |  | 1. Mengisi *form* tambah data dan klik *button save* | 1. Sistem melakukan validasi pada data yang diinputkan sebelum data dimasukan di dalam *database*. |
|  | Ubah | 1. Klik *button* ubahdata pelanggan | 1. Menampilkan *form* yang berisi detail data yang akan diubah. |
|  |  | 1. Menginputkan perubahan data kemudian klik *button save* | 1. Sistem melakukan validasi data yang telah diinputkan user dan menyimpan data ke dalam *database.*   (*Alternatif flow* 6.1) |
|  | Hapus | 1. Klik button hapus pada tampilan. | 1. Sistem akan menampilkan konfirmasi hapus data |
|  |  | 1. Klik ok pada layer konfirmasi hapus data. (*alternatif flow* 5.1) | 1. Sistem akan menghapus data |
| *Alternative Flow* | 5.1 Mengklik button cancel (Kembali ke Langkah 2)  6.1 Data yang diinputkan tidak sesuai dengan peraturan sistem (Kembali ke langkah 4) | | |
| *Requirement* | Berhasil login sebagai pegawai | | |
| *Post-condition* | Sistem melakukan Data Manipulation Language (DML) pada database sesuai dengan aksi yang dilakukan aktor (tambah, ubah dan hapus data) | | |

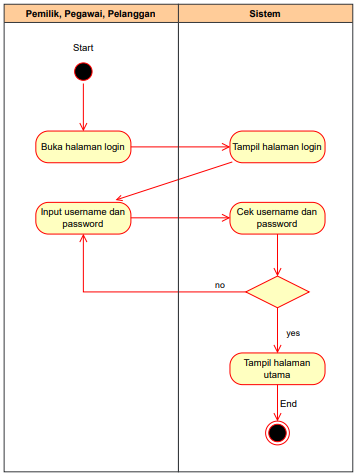
* + 1. **Activity Diagram**

1. *Activity Diagram* Registrasi



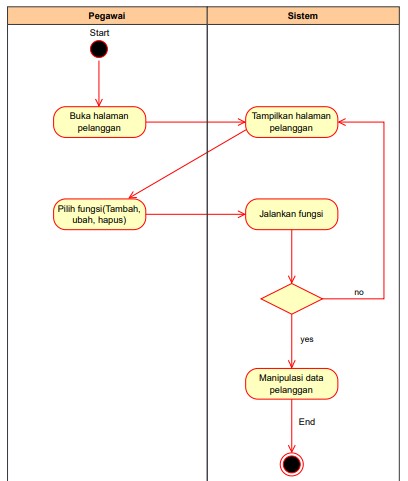
**Gambar 4.4** *Activity Diagram Registrasi*

1. *Activity Diagram* Login



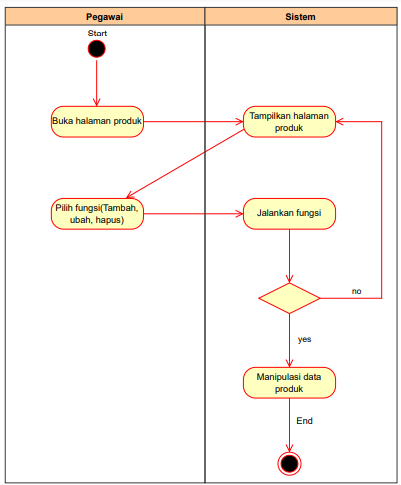
**Gambar 4.5** *Activity Diagram Login*

1. *Activity Diagram* Kelola Data Pelanggan



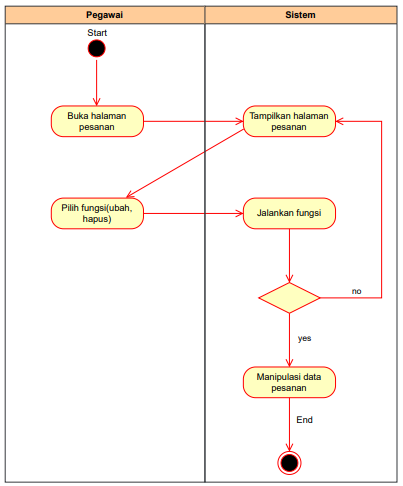
**Gambar 4.6** *Activity Diagram* Kelola Data Pelanggan

1. *Activity Diagram* Kelola Data Produk

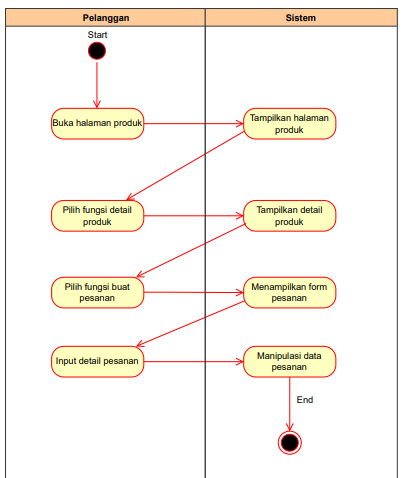


**Gambar 4.7** *Activity Diagram* Kelola Data Produk

1. *Activity Diagram* Kelola Data Pesanan

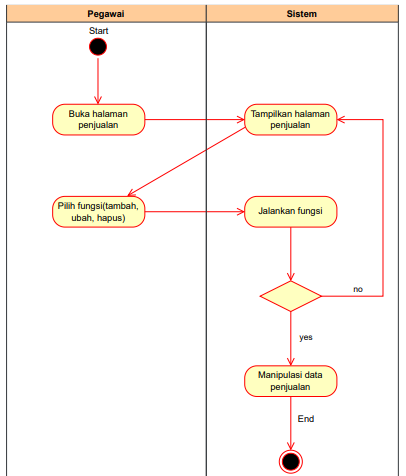


**Gambar 4.8** *Activity Diagram* Kelola Data Pesanan oleh Pegawai



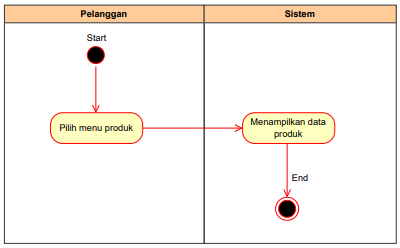
**Gambar 4.9** *Activity Diagram* Kelola Data Pesanan oleh Pelanggan

1. *Activity Diagram* Kelola Data Penjualan

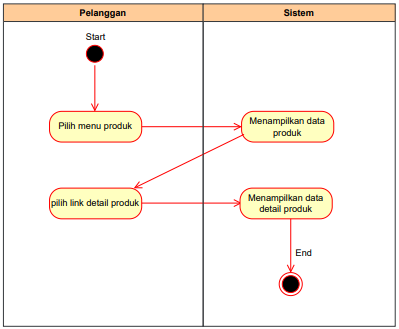


**Gambar 4.10** *Activity Diagram* Kelola Data Penjualan

1. *Activity Diagram* Lihat Produk

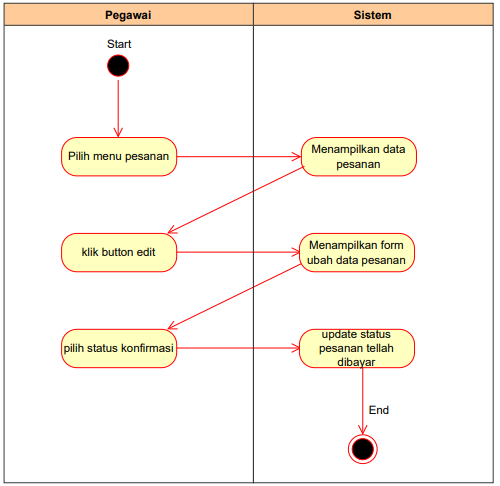


**Gambar 4.11** *Activity Diagram* Lihat Produk



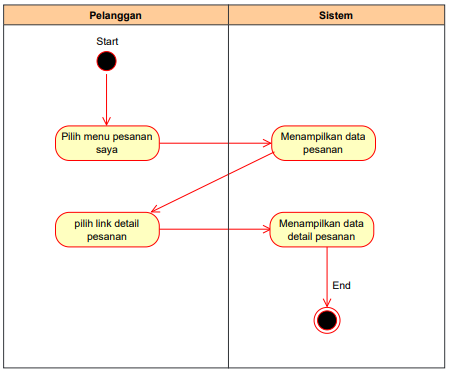
**Gambar 4.12** *Activity Diagram* Lihat Detail Produk

1. *Activity Diagram* *Approve* Pembayaran



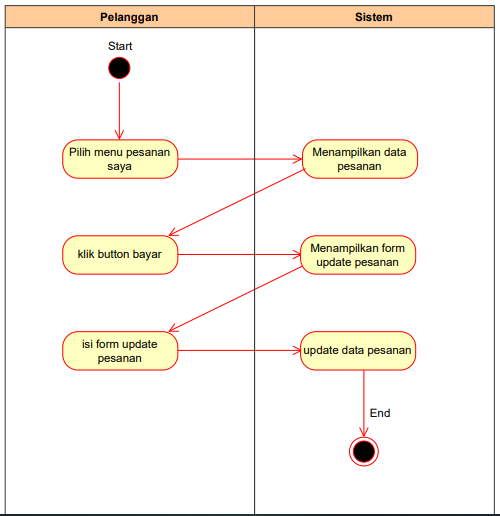
**Gambar 4.13** *Activity Diagram Approve* Pembayaran

1. *Activity Diagram* Lihat Progress Pesanan



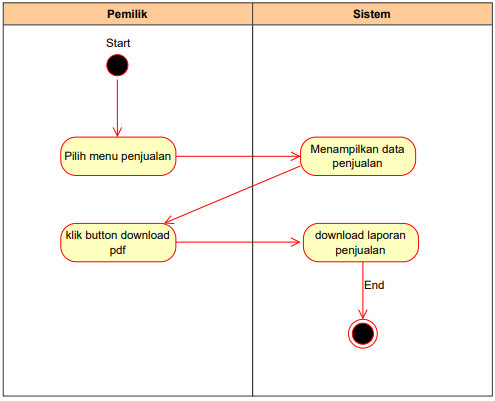
**Gambar 4.14** *Activity Diagram* Lihat Progress Pesanan

1. *Activity Diagram* Bayar Pesanan



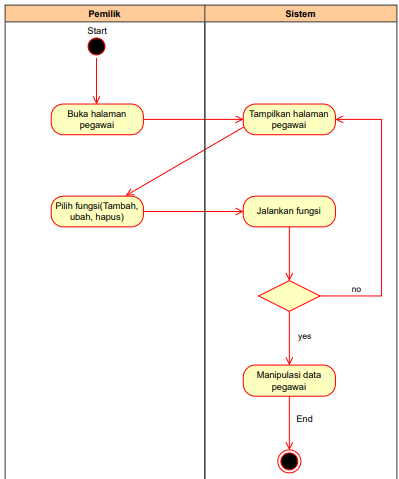
**Gambar 4.15** *Activity Diagram* Bayar Pesanan

1. *Activity Diagram* Lihat Laporan Penjualan

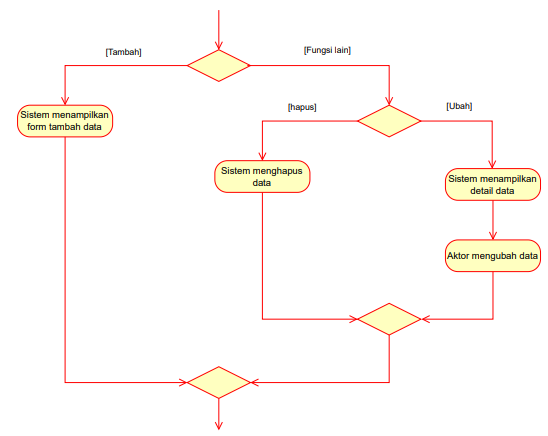


**Gambar 4.16** *Activity Diagram* Lihat Laporan Penjualan

1. *Activity Diagram* Kelola Data Pegawai

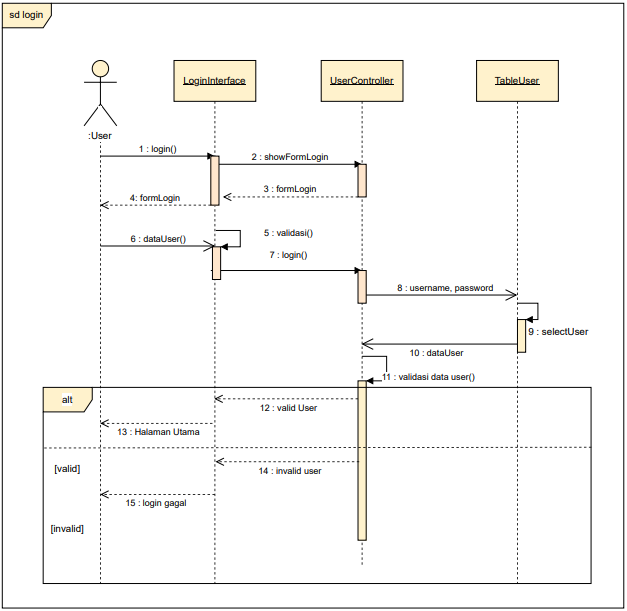


**Gambar 4.17** *Activity Diagram* Kelola Data Pegawai

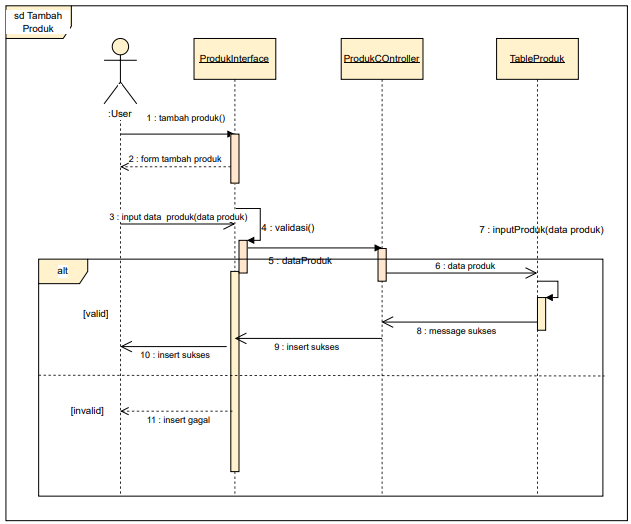
1. *SubActivity* Diagram Jalankan Fungsi

**Gambar 4.18** *SubActivity* Diagram Jalankan Fungsi

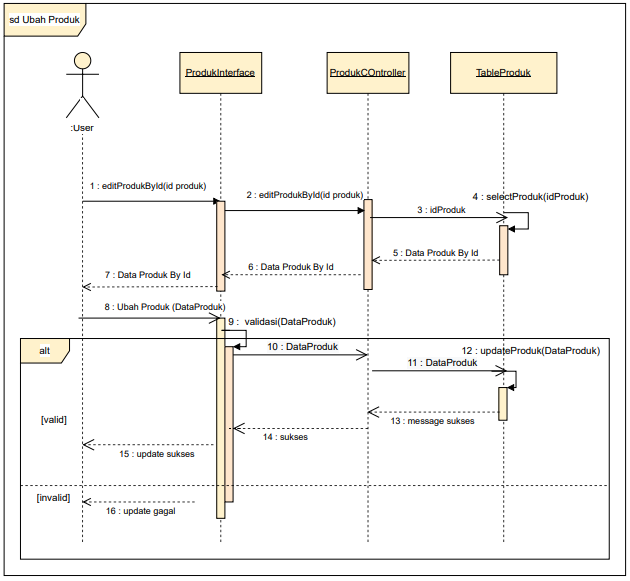
* + 1. **Sequence Diagram**



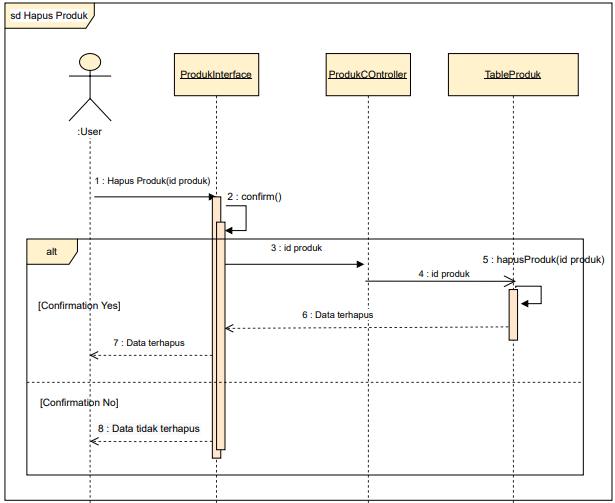
**Gambar 4.19** *Sequence Diagram Login*

**

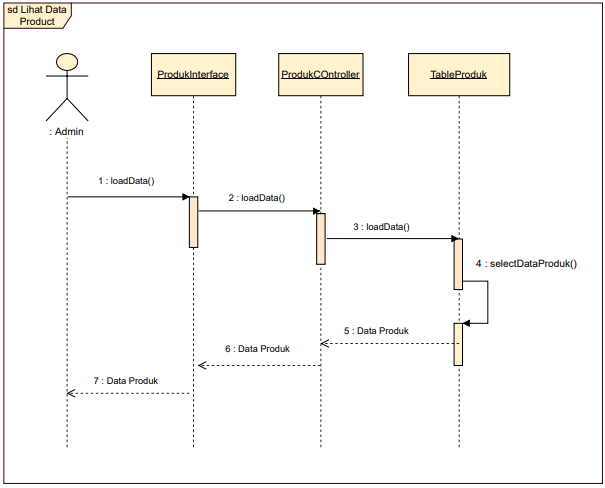
**Gambar 4.20** *Sequence Diagram* Tambah Produk



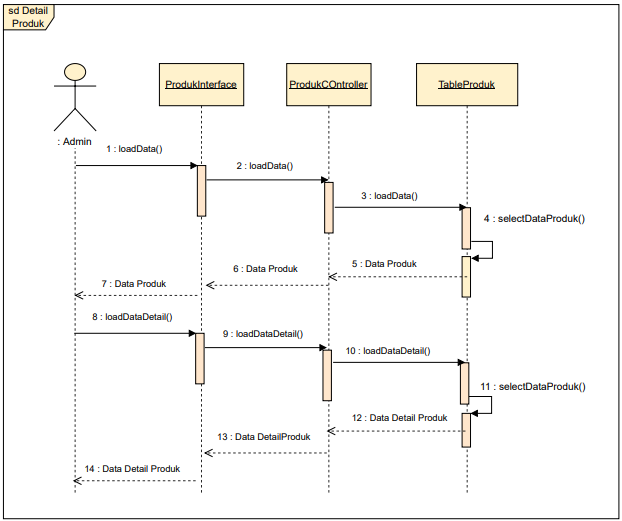
**Gambar 4.21** *Sequence Diagram* Ubah Produk



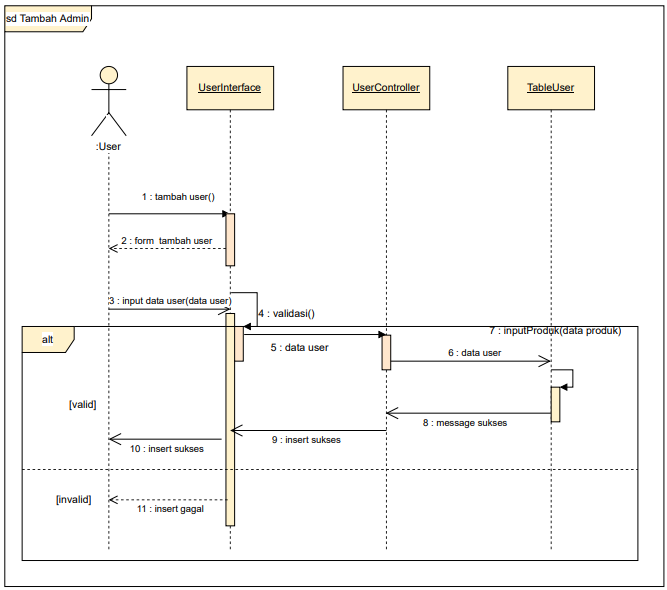
**Gambar 4.22** *Sequence Diagram* Hapus Produk



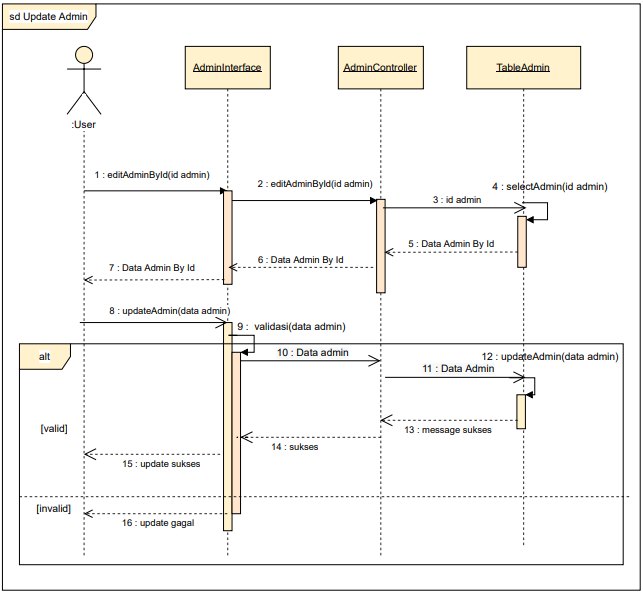
**Gambar 4.23** *Sequence Diagram* Lihat Data Produk

**

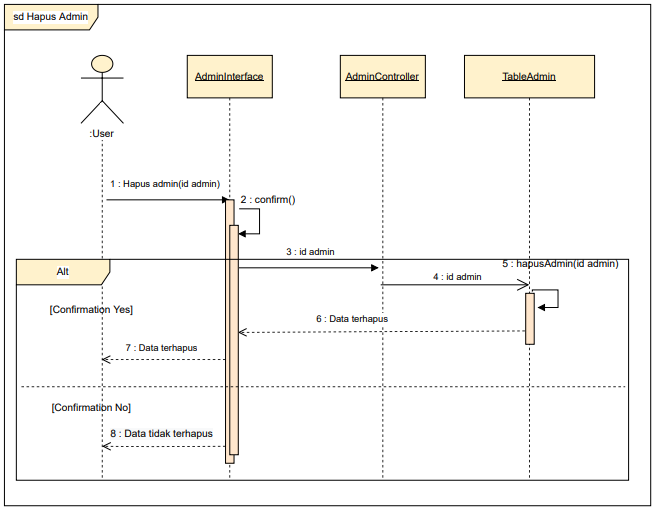
**Gambar 4.24** *Sequence Diagram* Lihat Detail Produk

**

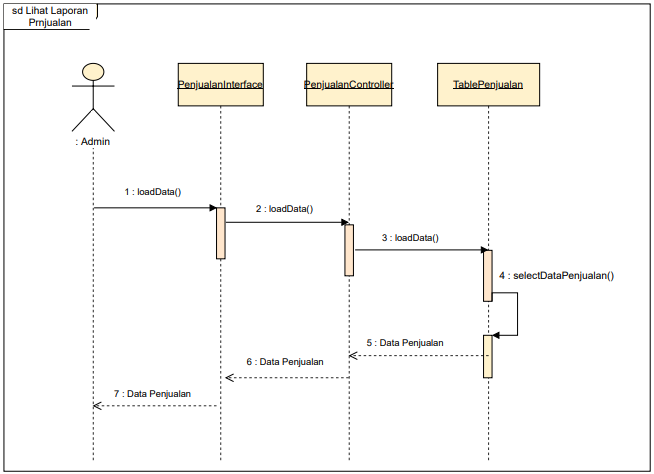
**Gambar 4.25** *Sequence Diagram* Lihat Tambah Admin



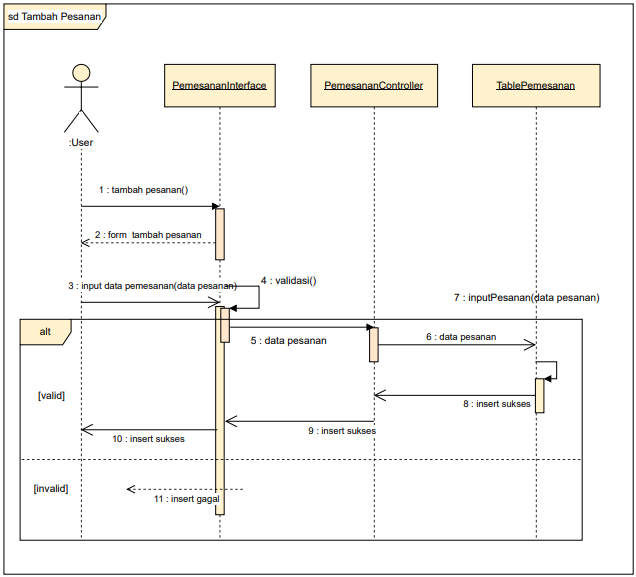
**Gambar 4.26** *Sequence Diagram* Lihat Update Admin



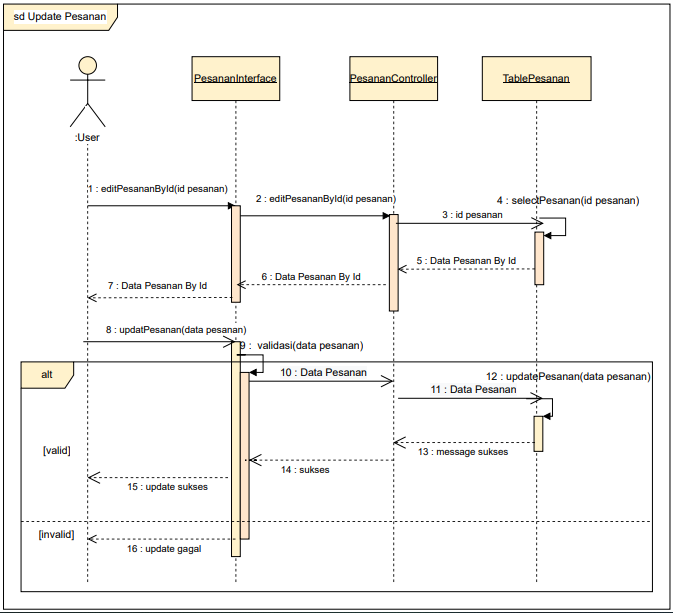
**Gambar 4.27** *Sequence Diagram* Lihat Hapus Admin



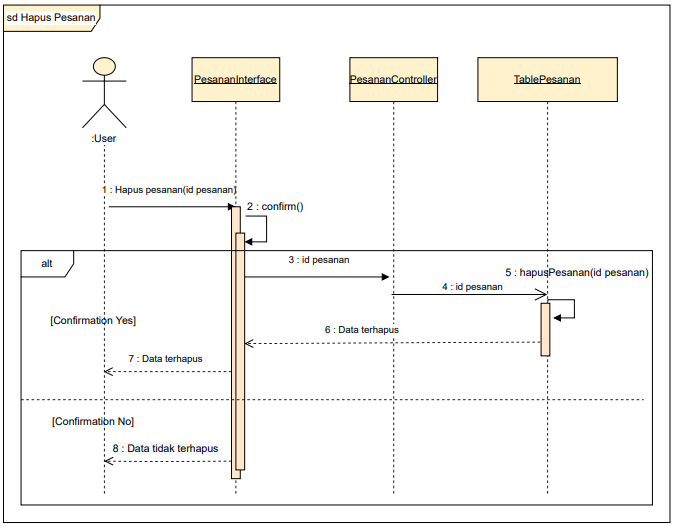
**Gambar 4.28** *Sequence Diagram* Lihat Laporan Penjualan



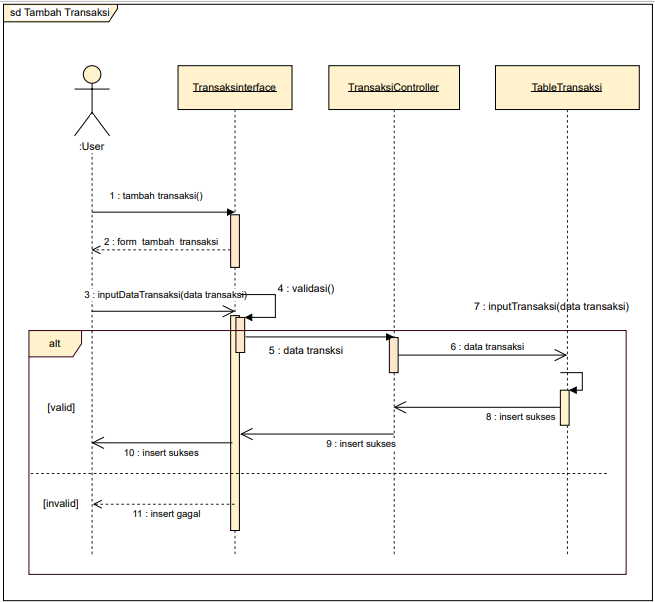
**Gambar 4.29** *Sequence Diagram* Tambah Pesanan



**Gambar 4.30** *Sequence Diagram* Ubah Pesanan



**Gambar 4.31** *Sequence Diagram* Hapus Pesanan



**Gambar 4.32** *Sequence Diagram* Tambah Transaksi

* + 1. Class Diagram