**Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis *Website*  untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering***

***SOFTWARE DESIGN DOCUMENTATION***

***Version:* <1.0>**

**Nashalu Aprilianti – 10110044**

**Fatimah Az-Zahra - 10110018**

**PROGRAM STUDI SISTEM INFORMASI**

**JURUSAN TEKNOLOGI INFORMASI DAN KOMPUTER**

**POLITEKNIK NEGERI SUBANG**

**2025**

# **DAFTAR ISI**

[**DAFTAR ISI** i](#_Toc200136792)

[**1.** **PENDAHULUAN** 1](#_Toc200136793)

[**1.1.** **Latar Belakang** 1](#_Toc200136794)

[**1.2.** **Tujuan** 2](#_Toc200136795)

[**1.3.** **Ruang Lingkup** 3](#_Toc200136796)

[**1.4.** **Definisi, Istilah dan Singkatan** 3](#_Toc200136797)

[**1.5.** **Referensi** 5](#_Toc200136798)

[**2.** **DESKRIPSI UMUM** 6](#_Toc200136799)

[**2.1.** **Perspektif Produk** 6](#_Toc200136801)

[**2.2.** ***Software Interface*** 7](#_Toc200136802)

[**2.3.** ***Hardware Interface*** 8](#_Toc200136803)

[**2.4.** **Manfaat Produk** 8](#_Toc200136804)

[**2.5.** **Karakteristik *Use Case*** 9](#_Toc200136805)

[**2.6.** **Batasan-Batasan** 10](#_Toc200136806)

[**2.7.** **Asumsi dan Ketergantungan** 10](#_Toc200136807)

[**3.** **ARSITEKTUR SISTEM** 11](#_Toc200136808)

[**3.1.** **Kebutuhan Fungsional** 11](#_Toc200136809)

[**3.1.1.** **Pendahuluan** 11](#_Toc200136812)

[**3.1.2.** **Input** 12](#_Toc200136813)

[**3.1.3.** **Proses** 14](#_Toc200136814)

[**3.1.4.** **Output** 16](#_Toc200136815)

[**3.2.** **Kebutuhan Non Fungsional** 17](#_Toc200136816)

[**4.** **PEMODELAN PERANGKAT LUNAK** 18](#_Toc200136817)

[**4.1.** **Use Case Diagram** 18](#_Toc200136818)

[**4.1.1.** **Deskripsi Aktor** 19](#_Toc200136819)

[**4.1.2.** **Deskripsi *Use Case* Diagram** 19](#_Toc200136820)

[**4.1.3.** **Scenario *Use Case* Diagram** 22](#_Toc200136823)

[**4.2.** **Activity Diagram** 74](#_Toc200136856)

# **PENDAHULUAN**

## **Latar Belakang**

Perkembangan pesat dalam teknologi informasi dan komunikasi di era digital telah mengakibatkan perubahan signifikan dalam banyak aspek kehidupan, termasuk sektor bisnis dan industri. Penggabungan teknologi dalam kegiatan bisnis tidak hanya meningkatkan efisiensi operasional tetapi juga menciptakan kesempatan baru untuk inovasi dan pertumbuhan. Internet yang dianggap lebih cepat dan efisien, sering kali memberikan nilai tambah kepada pengguna, serta informasi yang diinginkan dapat diakses dengan mudah. Sebagai hasilnya, banyak pengusaha menyusun model bisnis mereka untuk memenuhi tuntutan pasar dan menjadikannya keuntungkan [1] Salah satu metode untuk mencapai hal ini adalah dengan menerapkan sistem informasi yang terintegrasi, seperti Sistem Informasi Manajemen Pemesanan dan *Customer Relationship Management*  (CRM) berbasis web.

Pada usaha *Catering*, dihadapkan untuk memenuhi harapan yang dimiliki oleh pelanggan. Umumnya, usaha *catering* adalah bisnis yang berskala lokal dan fokus pada pasar di sekitarnya yang dapat diakses dengan transportasi darat [2] serta aktivitas yang memproduksi makanan yang akan dijual setelah pemesanan diterima, dengan tujuan akhir untuk memperoleh profit [3]. Fokus penelitian ini adalah untuk membantu Hanna *Catering*, sebuah usaha yang bergerak dalam penyediaan makanan, dalam memperbarui cara mereka mengelola pemesanan dan berinteraksi dengan pelanggan. Masalah yang dihadapi oleh Hanna *Catering* adalah bahwa pemesanan hanya dapat dilakukan melalui admin *WhatsApp* , pemesanan yang tidak teratur, dan pelanggan harus bertanya terlebih dahulu untuk melihat menu dan harga. Selain itu, usaha *Catering* yang masih menggunakan sistem manual dalam mengelola pemesanan, yang menyebabkan keterlambatan, kesalahan dalam pesanan, serta kesulitan dalam menjaga hubungan yang baik dengan pelanggan.

Penelitian ini bertujuan memberikan panduan bagi Hanna *Catering* untuk memanfaatkan sistem informasi berbasis *website* yang tidak hanya memudahkan proses pemesanan tetapi juga memungkinkan perusahaan untuk memonitor sekaligus menganalisis kebutuhan pelanggan dengan lebih efisien. Penerapan CRM yang efektif juga sangat penting karena dapat memperbaiki pengalaman pelanggan, yang dapat meningkatkan kepuasan dan loyalitas.

Implementasi CRM berbasis *website* diharapkan dapat memperkuat hubungan jangka panjang dengan pelanggan lewat layanan yang lebih personal, sekaligus mengurangi kesalahan yang sering muncul dalam proses pemesanan yang masih dilakukan secara manual. Dengan latar belakang ini, penelitian diharapkan dapat memberikan solusi yang relevan untuk Hanna *Catering* serta mempermudah pelanggan dalam melakukan pemesanan, sekaligus meningkatkan pengalaman dan kepuasan terhadap layanan yang diberikan.

## **Tujuan**

Tujuan Pengembangan Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis Website untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering* sebagai berikut :

1. Menggambarkan perancangan sistem untuk otomatisasi tugas-tugas manual dalam proses pencatatan pesanan untuk meningkatkan efesiensi dan mengurangi risiko kesalahan.
2. Mengilustrasikan perancangan sistem yang mengelola seluruh proses pemesanan, mulai dari menerima pemesanan, penjadwalan, pengelolaan menu hingga status pemesanan.
3. Mengambarkan perancangan sistem terkait *fitur*-*fitur* CRM yang dapat diintegrasikan ke dalam sistem pemesanan untuk meningkatkan interaksi dengan pelanggan, layanan dan pengelolaan data pelanggan.

## **Ruang Lingkup**

Adapun ruang lingkup dari sistem yang akan dibuat antara lain :

1. Sistem Informasi Manajemen Pemesanan berfokus pada menerima pesanan, mencatat detail pesanan, pengelolaan menu seperti penambahan, pengeditan, pengahapusan, mengatur harga dan deskripsi menu. Sistem membuat penjadwalan pesanan, pelacakan status pesanan seperti “Diterima”,”Diproses”,”Dikirim”,”Selesai”.
2. Perancangan sistem menyediakan *fitur* CRM yang mencakup pendaftaran dan pengelolaan data pelanggan secara ringkas, pencatatan riwayat pemesanan pelanggan, penawaran khusus yang ditargetkan, serta *fitur* untuk mengumpulkan umpan balik dan pengawasan kepuasan pelanggan.
3. Sistem ini mendukung proses pembayaran *online*.

## **Definisi, Istilah dan Singkatan**

|  |  |
| --- | --- |
| SRS : | *Software Requrement Specification* (Spesifikasi Kebutuhan Perangkat Lunak) |
| PHP : | PHP merupakan singkatan dari *Personal Home Page* (situs personal). PHP adalah bahasa pemrograman yang dikenal sebagai bahasa *scripting* dalam arti bahwa PHP adalah bahasa pemrograman yang tertanam dalam bahasa dan aplikasi lain. Misalnya, PHP termasuk dalam skrip HTML, bahasa asli *World Wide Web* [4] |
| MySQL : | MySQL adalah Suatu *software* atau program yang digunakan untuk membuat sebuah basis data yang bersifat *open source* [5]. |
| WEB : | *Website* atau situs dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*) [6]. |
| Laravel : | Laravel merupakan sebuah *Framework* PHP (*Hypertext Preprocessor*) yang dirilis dibawah lisensi MIT, dibangun dengan konsep MVC (*Model, View, Controller*). Laravel adalah *Framework* PHP yang dibuat oleh Taylor Otwell dan dirilis pertama kali pada tahun 2011 [7] |
| HTML : | *Hypertext Markup Language* adalah bahasa pemrograman untuk membuat halaman *Website*  yang ditampilkan pada web *browser*. |
| Xampp : | XAMPP adalah perangkat lunak gratis yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang terdiri sendiri (*localcost*), yang terdiri atas program *Apache* HTTP *Server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemograman PHP dan Perl [6]. |
| CRM : | *Customer Relationship Management* |
| ERD : | *Entity Relationship Diagram* |
| *Payment Gateway* : | Sistem pembayaran yang memfasilitasi proses pembayaran secara *online*. |

## **Referensi**

[1] N. Fitria, A. M. Harahap, and F. H. Sibarani, “SISTEM INFORMASI MANAJEMEN *CATERING* MENGGUNAKAN CUSTOMER RELATIONSHIP MANAGEMENT (CRM) BERBASIS WEB,” 2024.

[2] M. Rahmatuloh and M. R. Revanda, “Rancang Bangun Sistem Informasi Jasa Pengiriman Barang Pada Pt. Haluan Indah Transporindo Berbasis Web,” *Use Case*, vol. 14, no. 1, pp. 54–59, 2022.

[3] A. Nasution, “Strategi Pemasaran Bisnis Kuliner Menggunakan Influencer Melalui Media Sosial Instagram,” *Use Case*, vol. 6, no. 1, pp. 11–18, 2021.

[4] I. Murni, B. R. Lubis, and A. Ikhwan, “Pengamanan Pesan Rahasia dengan Algoritma Vigenere Cipher Menggunakan PHP,” *Use Case*, vol. 5, no. 2, pp. 3466–3476, 2023.

[5] Y. Trimarsiah and M. Arafat, “Analisis dan perancangan website sebagai sarana informasi pada lembaga bahasa kewirausahaan dan komputer akmi baturaja,” *Use Case*, pp. 1–10, 2017.

[6] A. Suprianto and A. A. F. Matsea, “Rancang bangun aplikasi pendaftaran pasien online dan pemeriksaan dokter di klinik pengobatan berbasis web,” *Use Case*, vol. 7, no. 1, 2018.

[7] A. H. Putra, D. Pramana, and N. L. P. Srinadi, “Sistem Manajemen Arsip Menggunakan Framework Laravel dan Vue. Js (Studi Kasus: BPKAD Provinsi Bali),” *Use Case*, vol. 13, no. 2, pp. 97–104, 2019.

# **DESKRIPSI UMUM**

Dalam deskripsi umum ini menggambarkan mengenai spesifikasi terkait Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis *Website*  untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering* yang akan dibangun. Dokumen ini berisi dekripsi terkait kebutuhan *software* dan *hardware,* manfaat produk dan batasan-batasan pada sistem yang akan dibangun.



## **Perspektif Produk**

*Website* Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis *Website*  untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering* adalah sebuah sistem informasi berbasis *Website*  dengan menerapkan CRM (*Customer Relationship Management*) yang dibangun untuk memudahkan pelanggan maupun perusahaan untuk mengelola pesanan serta dapat meningkatkan kepuasaan pelanggan Hanna *Catering*. Sistem ini nantinya akan dikelola oleh admin yang bertugas sebagai pengelola *Website*  maupun *WhatsApp.* Sistem ini memiliki beberapa *fitur* maupun menu terkait pengelolaan pemesanan seperti halaman utama, Menu *Catering*, Daftar Harga, Ulasan, Tentang Hana *Catering* selain itu ada *fitur* pemesanan, pembayaran, informasi kontak dan notifikasi pesanan.

## ***Software Interface***

Dalam pembuatan sistem berbasis *Website* ini, dibutuhkan *software*  untuk mendukung proses pembuatan Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis *Website* untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering*. *Software*  tersebut antara lain :

* + 1. Sistem Operasi (*User*): Semua tipe plat*form*, disarankan menggunakan *Google Chrome* dan hanya bisa diakses melalui *Website*.
    2. Data

*Database*: MySQL

*Thesaurus*: Bahasa Indonesia

* + 1. Bahasa pemrograman Web

Bahasa pemrograman: PHP, HTML, CSS dan Javascript

*Framework*: Laravel

* + 1. *Browser*

Semua *browser* dan disarankan menggunakan *Google Chrome* serta hanya dapat diakses melalui *Website*  karena belum tersedia versi *mobile.*

* + 1. Desain Antarmuka

Alat desain: Figma dan Canva

* + 1. *Software*  Pendukung Lain

*Software*  pendukung lainnya sebagai berikut:

1. Visual Studio Code
2. Microsoft Word
3. StarUML

## ***Hardware Interface***

*Hardware Interface* yang dibutuhkan untuk membantu proses pembangunan dari sistem ini yaitu :

1. *Keyboard* yang merupakan salah satu alat untuk proses mengimput informasi yang dibutuhkan oleh sistem. Baik berupa angka, karakter maupun simbol-simbol.
2. *Mouse* yang dapat membantu sistem untuk mengenali *input* dari pengguna dengan melakukan klik.
3. *Monitor* yang dapat membantu pengguna untuk mengetahui dan menampilkan apa saja yang menjadi *ouput* dari sistem dan menjadi perantara komunikasi antara sistem dengan pengguna.

## **Manfaat Produk**

Manfaat dari produk yang dapat diperoleh dengan menggunakan sistem *Website*  ini sebagai berikut :

1. Kemudahan akses dalam proses pemesanan

Pelanggan dapat melakukan pemesanan kapan saja dan dimana saja melalui perangkat mereka seperti komputer, laptop, tablet, smartphone dan tidak perlu lagi untuk datang langsung ke tokonya atau tanya-tanya terlebih dahulu untuk melihat menu yang tersedia. Proses pemesanan yang lebih efesien karena pelanggan dapat langsung memilih menu paket dan mengisi detail pemesanan.

1. Katalog Produk dan harga terstruktur

Pelanggan dengan mudah dapat melihat semua menu yang tersedia dengan deskripsi lengkap, gambar berkualitas tinggi, dan harga yang jelas.

1. Meningkatan kepuasaan pelanggan dengan sistem *feedback*

Sistem *feedback* memungkinkan pelanggan memberikan umpan balik, saran, atau keluhan pelanggan dengan mudah serta memberi rating *Catering*. Admin dapat menanggapi umpan balik pelanggan untuk menunjukan bahwa bisnis peduli dengan mereka dan berkomitmen untuk memberikan pengalaman yang lebih baik.

1. Pemasaran yang lebih efektif

Sistem dapat memberikan data tentang jumlah tampilan, klik, dan konversi. Ini memungkinkan bisnis mengoptimalkan strategi pemasaran mereka untuk mencapai hasil yang lebih baik.

1. Kemudahan dalam pembayaran dan transaksi

Sistem menyediakan pilihan pembayaran yang aman dan nyaman, transfer bank dan dompet *digital*. Pelanggan dapat dengan mudah melihat riwayat transaksi mereka, status pesanan, dan detail pembayaran.

## **Karakteristik *Use Case***

* 1. Admin

Admin bertanggung jawab atas seluruh pengelolaan keseluruhan sistem. Admin yang memiliki hak akses penuh ke seluruh menu dan fungsi yang ada di dalam sistem. Admin dapat mengelola data penjualan, pemesanan, data *user*.

* 1. Pelanggan

Pelanggan merupakan aktor utama yang dimana pelanggan dapat mengakses *Website* tersebut untuk melihat, memesan, memberikan ulasan dan melakukan pembayaran di *Website*  tersebut.

* + 1. Pemilik

Pemilik merupakan pihak yang memiliki bisnis dan sistem. Peran pemilik memiliki tingkat akses yang tinggi, meskipun mungkin tidak seluas admin dalam pengelolaan operasional sehari-hari. Pemilik berfokus pada pemantauan kinerja bisnis secara keseluruhan, pengambilan keputusan strategis, dan pengelolaan aspek keuangan.

## **Batasan-Batasan**

Batasan perancangan Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis *Website*  untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering* antara lain yaitu :

1. Sistem Informasi Manajemen Pemesanan berfokus pada pengelolaan data pemesanan, termasuk data pengguna pelanggan, data barang atau menu, dan laporan-laporan terkait. Serta Sistem dirancang untuk mempermudah pelanggan dalam melakukan pemesanan.
2. Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM berfokus pada penerapan CRM (*Customer Relationship Management*) yang dimana tersedia *fitur* *chatting* yang dapat memberikan wadah untuk berdiskusi antara pelanggan dan admin. Serta *fitur* dimana pelanggan dapat memberikan ulasan terkait pesanannya.

## **Asumsi dan Ketergantungan**

Asumsi dan ketergantungan yang digunakan dalam sistem *Website*  ini adalah :

* 1. Pengguna sistem diharuskan mengetahui bagaimana cara menggunakan maupun mengoperasikan teknologi internet maupun komputer terutama berbasis *Website* .
  2. Sistem *Website*  ini terhubung dengan akses internet dan perangkat yang mendukung akses *Website*.
  3. Keamanan data pelanggan terjamin.
  4. Pelanggan yang bersedia menggunakan sistem *online* untuk melakukan pemesanan dan pembayaran.

# **ARSITEKTUR SISTEM**

## **Kebutuhan Fungsional**



### **Pendahuluan**

Kebutuhan fungsional adalah kebutuhan yang harus dipenuhi untuk memastikan suatu sistem dapat beroperasi atau dalam kata lain kebutuhan tambahan yang mencangkup *input*, proses, dan *output*. Kebutuhan fungsional yang akan dikembangkan pada Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis *Website* untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Catering* yaitu :

1. Pengguna dapat mendaftar, membuat akun, dan *login*  untuk melacak pesanan.
2. Pengguna dapat memilih dan memesan menu *Catering*, memilih paket sesuai dengan kebutuhan acara, serta menentukan jumlah pesanan.
3. Sistem dapat memproses pembayaran secara aman menggunakan metode seperti transfer bank atau *e-wallet*.
4. Admin dapat mengelola, memproses, dan mengonfirmasi pesanan yang masuk.
5. Sistem mengirimkan konfirmasi pemesanan dan pengingat otomatis kepada pelanggan melalui notifikasi dan email.
6. Mengelola data pelanggan, memberikan rekomendasi berdasarkan riwayat pesanan, dan melacak interaksi pelanggan.

### **Input**

*Input* pada Proses *Register*

|  |  |
| --- | --- |
| Kolom email | Untuk masukkan email akun |
| Kolom *username* | Untuk masukkan *username* akun |
| Kolom *password* | Untuk masukkan *password* |
| Kolom nomor telepon | Untuk masukkan nomor telepon |
| Tombol *register* | Untuk *register* |

*Input* pada Proses *Login*

|  |  |
| --- | --- |
| Kolom *Username* | Untuk masukkan *username* akun |
| Kolom *Password* | Untuk masukkan *password* |
| Tombol *Login* | Untuk *Login* |

*Input* Proses Menu

|  |  |
| --- | --- |
| Kolom Nama Menu | Untuk masukkan nama menu |
| Kolom Harga Menu | Untuk masukkan harga menu |
| Kolom Deskripsi | Untuk masukkan deskripsi menu |
| Kolom Foto Menu | Untuk masukkan foto menu |
| Opsi Kategori | Untuk menentukan kategori produk |
| Tombol Simpan | Untuk menyimpan |
| Tombol Hapus | Untuk menghapus |
| Tombol Batal | Untuk membatalkan |
| Tombol Edit | Untuk mengedit |

*Input* Proses Pemesanan

|  |  |
| --- | --- |
| Kolom Nama Lengkap | Untuk masukkan nama pelanggan |
| Kolom Nomor Telepon | Untuk masukkan nomor telepon |
| Kolom Alamat Lengkap | Untuk memasukan alamat pelanggan |
| Opsi Keterangan Pengambilan | Untuk menentukan pengambilan pemesanan |
| Kolom Tanggal Pengambilan/ Pengantaran | Untuk memasukaan tanggal pengambilan/pengantaran |
| Tombol *Chekout* | Untuk menchekout pesanan |
| Kolom Kuantitas | Untuk memasukkan jumlah yang akan dipesan |
| Tombol *Cancel* | Untuk membatalkan pesanan |
| Tombol Tambah Keranjang | Untuk menambahkan menu ke keranjang |

*Input* Proses Pembayaran

|  |  |
| --- | --- |
| Kolom Nomor *Invoice* | Untuk memasukkan nomor invoice transaksi |
| Kolom Nama Pelanggan | Untuk memasukkan nama pelanggan |
| Opsi Keterangan Pembayaran | Untuk menentukan keterangan pembayaran seperti DP atau Pelunasan |
| Opsi Metode Pembayaran | Untuk menentukan metode pembayaran seperti transfer bank atau *e-wallet* |
| Kolom Status Pembayaran | Untuk mengupdate status pembayaran (*pending*/lunas) |
| Kolom Tanggal Pembayaran | Untuk memasukkan tanggal pembayaran dilakukan |
| Tombol Cek Status Pembayaran | Untuk memeriksa status pembayaran. |

### **Proses**

Pada halaman utama *website* Hanna *Catering* terdapat tampilan beranda, menu *Catering*, daftar harga, ulasan, tentang Hanna *Catering*. Untuk admin maka halaman utamanya langsung *login*.

Proses dimulai dengan *registrasi* dan *login*, dimana pelanggan diharuskan membuat akun untuk mengakses layanan, sedangkan admin dan pemilik bisa *login* untuk mengelola informasi pelanggan dan pesanan. Setelah itu, sistem akan merekam dan menyimpan informasi pelanggan, termasuk nama, nomor telepon, email, dan *password* didalam *database*.

Menu pemesanan, pelanggan dapat memilih item menu, menetapkan jumlah yang diinginkan, dan mengisi *form*ulir pemesanan seperti nama lengkap, nomor telepon, alamat lengkap, keterangan, tanggal diambil atau diantarkan kemudian pelanggan memilih tombol chekout.

Menu pembayaran, pelanggan dapat melakukan pembayaran dengan metode pembayaran bertahap, seperti pembayaran uang muka dan sisa pembayaran. Setelah pembayaran diterima, admin akan memproses pemesanan dan mengirimkan invoice kepada pelanggan.

Status pemesanan secara otomatis diperbarui, mulai dari Diproses, Dikirim, hingga Selesai, dan pelanggan akan menerima pemberitahuan melalui email atau notifikasi di*website* untuk memastikan transparansi dalam setiap proses transaksi.

Sistem menyediakan dukungan pelanggan, dimana pelanggan dapat mengajukan pertanyaan atau menyampaikan keluhan melalui *fitur chat*. Admin akan memberikan tanggapan cepat untuk meningkatkan kepuasan pengguna. Dengan bantuan CRM, pemilik dapat menganalisis data pelanggan untuk merumuskan strategi harga dan inovasi paket *Catering* yang lebih sesuai dengan permintaan pasar.

Menu ulasan, pelanggan dapat memberikan umpan balik seperti ulasan dan rating tentang kualitas makanan serta pelayanan. Umpan balik ini sangat berharga bagi admin dan pemilik untuk melakukan perbaikan dan meningkatkan standar layanan.

Sistem CRM ini menghasilkan laporan dan analisis penjualan, yang memungkinkan pemilik untuk melihat kinerja bisnis secara langsung. Dari laporan ini, pemilik dapat mengambil keputusan strategis, seperti menambah variasi menu, memperbaiki sistem operasional, atau merencanakan ekspansi usaha.

### **Output**

Pada sistem *website* hanna *Catering*, proses *login* menghasilkan *ouput* berupa autentikasi sehingga pelanggan, admin, dan pemilik akan masuk ke halaman masing-masing sehingga dapat mengakses seluruh informasi yang ada di halaman masing-masing.

Ketika pelanggan melakukan pemesanan, sistem akan menampilkan rincian pesanan, termasuk pilihan menu, jumlah porsi, total biaya, serta pilihan untuk pembayaran uang muka atau pelunasan. Setelah pembayaran terkonfirmasi, sistem akan mengeluarkan faktur digital yang dapat diunduh oleh pelanggan sebagai bukti transaksi. Selain itu, pelanggan akan menerima pemberitahuan mengenai status pesanan melalui *email* atau notifikasi di *website*nya, yang mencakup proses, pengiriman, dan penyelesaian.

Bagi admin, sistem menyediakan daftar pesanan yang masuk, mencakup informasi rinci tentang pesanan pelanggan, cara pembayaran, dan status pembayaran. Admin juga menerima laporan transaksi setiap hari, yang menunjukkan jumlah pesanan yang telah diproses, pembayaran yang tercatat, serta daftar pesanan yang menunggu konfirmasi. Sistem ini juga menyajikan laporan pengiriman, yang mencatat pesanan yang telah dikirim.

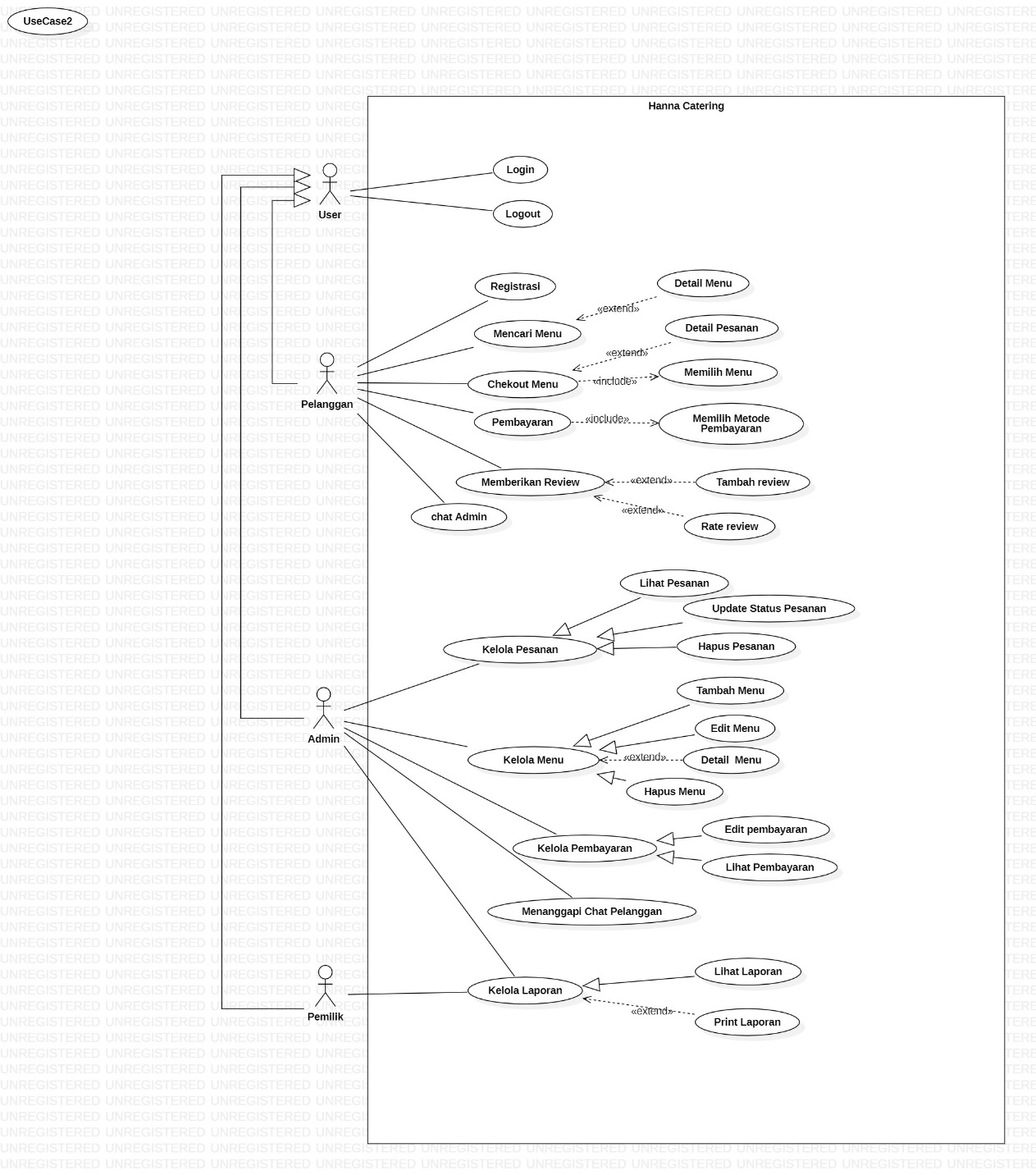
Bagi pemilik dapat menampilkan laporan keuangan, grafik penjualan bulanan, serta analisis tren pemesanan. Melalui laporan ini, pemilik dapat mengidentifikasi menu terlaris. Sistem ini juga menyediakan laporan umpan balik pelanggan, berisi ulasan dan penilaian dari pelanggan, yang membantu pemilik dalam meningkatkan kualitas layanan.

## **Kebutuhan Non Fungsional**

|  |  |
| --- | --- |
| **Parameter** | **Requirements** |
| *Availability* | Sistem harus tersedia 24/7 dengan *downtime* minimal, untuk memastikan pelanggan dapat mengaksesnya kapan saja kecuali saat dilakukan pemeliharaan atau perbaikan. |
| *Portability* | *Website*  harus dapat diakses di berbagai *platform* dan perangkat, termasuk desktop, laptop, tablet, dan ponsel. |
| *Ergonomy* | Sistem harus *User-friendly*, mudah digunakan, dengan navigasi yang jelas agar pengguna dapat memesan dengan mudah. |
| *Security* | Sistem harus menjamin tingkat keamanan yang tinggi, termasuk pengamanan data masuk dan perlindungan dari akses yang tidak diizinkan. |
| BahasaKomunikasi | Bahasa yang digunakan dalam *Website*  adalah Bahasa Indonesia untuk memudahkan pengguna lokal. |

# **PEMODELAN PERANGKAT LUNAK**

## **Use Case Diagram**

****

### **Deskripsi Aktor**

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1. | Admin | Admin bertanggung jawab atas seluruh pengelolaan keseluruhan sistem. Admin yang memiliki hak akses penuh ke seluruh menu dan fungsi yang ada di dalam sistem. Admin dapat mengelola data penjualan, pemesanan, data *User.* |
| 2. | Pelanggan | Pelanggan merupakan aktor utama yang dimana pelanggan dapat mengakses *Website* tersebut untuk melihat, memesan, memberikan ulasan dan melakukan pembayaran di *Use Case* tersebut. |
| 3. | Pemilik | Pemilik merupakan pihak yang memiliki bisnis dan sistem. Peran pemilik memiliki tingkat akses yang tinggi, meskipun mungkin tidak seluas admin dalam pengelolaan operasional sehari-hari. Pemilik berfokus pada pemantauan kinerja bisnis secara keseluruhan, pengambilan keputusan strategis, dan pengelolaan aspek keuangan. |

### **Deskripsi *Use Case* Diagram**

|  |  |  |
| --- | --- | --- |
| No | *Use Case* | Deskripsi |
|  | *Login* | Pengguna masuk ke dalam sistem dengan mengautetinfikasi diri menggunakan informasi *login.* |
|  | Registrasi | Pengguna daftar akun untuk mendapatkan akses ke sistem. |
|  | Mencari Menu | Mencari menu didalam sistem. |
|  | Detail Menu | Melihat detail menu didalam sistem. |
|  | *Chekout* Menu | Menchekout menu didalam sistem. |
|  | Memilih Menu | Memilih menu yang akan dipesan didalam sistem. |
|  | Detail Pesanan | Melihat detail pesanan yang akan dipilih. |
|  | Memilih Metode  Pembayaran | Memilih metode pembayaran yang ada di dalam sistem. |
|  | Pembayaran | Melakukan pembayaran didalam sistem. |
|  | Memberikan *Review* | Pelanggan memberikan *review* disistem |
|  | Tambah *Review* | Menambah *review* menu didalam sistem. |
|  | Rate *Review* | Merating menu didalam sistem. |
|  | *Chat* Admin | Menanyakan melalui *chat* terkait menu atau pesanan didalam sistem. |
|  | Lihat Pesanan | Melihat pesanan didalam sistem. |
|  | *Update* Status Pesanan | Admin mengupdate status pesanan seperti “diproses”,”selesai”. |
|  | Kelola Pesanan | Admin mengelola pesanan pelanggan. |
|  | Tambah Menu | Admin menambahkan menu didalam sistem. |
|  | Edit Menu | Admin mengedit menu didalam sistem. |
|  | Hapus Menu | Admin menghapus menu didalam sistem. |
|  | Detail Menu | Melihat detail menu didalam sistem. |
|  | Kelola Menu | Admin mengelola menu seperti menambahkan, menghapus, mengedit menu. |
|  | Edit Pembayaran | Admin mengedit pembayaran seperti membuat *invoice*, memasukan dp dan sisa pembayaran. |
|  | Kelola Pembayaran | Admin mengelola pembayaran didalam sistem. |
|  | Menanggapi *Chat*  Pelanggan | Admin menanggapi *chat* yang dikirimkan pelanggan. |
|  | Lihat Laporan | Admin dan pemilik dapat melihat laporan didalam sistem. |
|  | *Print* Laporan | Admin dan pemilik dapat mengeprintkan laporannya. |
|  | Kelola Laporan | Admin dan pemilik dapat mengelola laporan. |
|  | *Logout* | Pengguna keluar dari. |

### **Scenario *Use Case* Diagram**

Berikut merupakan beberapa *Use Case*pada Sistem Informasi Manajemen Pemesanan Beserta Penerapan CRM Berbasis untuk Meningkatkan Kepuasan Pelanggan pada Hanna *Use Case*:

1. *Login*

Tabel 4.1 Skenario Login

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-01 | |
| Nama *Use Case* | | *Login* | |
| Tujuan | | Melakukan *login* sebagai aktor yang memiliki hak akses ke dalam sistem | |
| Deskripsi | | *Use Case* ini menggambarkan proses otentikasi pengguna yang akan mengakses *system*. | |
| Nama aktor | | * Pelanggan * Admin * Pemilik | |
| Triger Event | | *User* membuka halaman *login* dan memasukkan kredensial (*username* dan *password*) lalu mengklik tombol "*Login*". | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin: bertanggung jawab untuk mengelola sistem dalam suatu layanan.  3. Pemilik : bertanggung jawab atas keberlangsungan usaha tersebut. | |
| Related *Use Case* | | Register : Sebelum dapat *login*, pengguna baru (pelanggan) yang belum memiliki akun harus melakukan registrasi terlebih dahulu. *Use Case* ini mencangkup pembuatan akun baru dengan memasukan informasi yang diperlukan untuk mendapatkan akses ke sistem. | |
| Skenario utama | | |  |
| Pre kondisi :  Aktor belum melakukan *login.* | |  | |
| Aktor | | Reaksi sistem | |
| * + - 1. *User* mengakses sistem. | |  | |
|  | | * + - 1. Menampilkan *form* *login*. | |
| * + - 1. Mengisi *form* *login*. | |  | |
|  | | * + - 1. Sistem memeriksa *Field* kosong pada *form* *login*. | |
|  | | * + - 1. Sistem mencocokan data *login* dengan data yang berada pada *database*.       2. Sistem melakukan validasi pengguna dan menampilkan notifikasi *login* berhasil.       3. Sistem menampilkan halaman utama. | |
| *Post*-kondisi:  • Jika *login* berhasil, pengguna akan diarahkan ke halaman utama sistem yang sesuai dengan hak akses mereka.  • Jika *login* gagal, pengguna akan menerima notifikasi kesalahan dan tetap berada di halaman *login* untuk mencoba lagi. | | | |
| **Exception** | |  | |
| 1. *Field* *Username* tidak diisi : Sistem akan memberikan alert bahwa *Field* *Username* harus diisi.  2. *Field* *Password* tidak diisi: Sistem akan memberikan alert bahwa *Field* *password* harus diisi.  3. *Field* *Username* dan *Password* tidak diisi : Sistem akan memberikan alert bahwa kedua *Field* harus diisi.  4. *Username* atau *password* yang diisi salah : sistem akan memberikan notifikasi bahwa *login* gagal *username* atau *password* yang dimasukkan tidak salah. | | | |

1. Register

Tabel 4.2 Skenario Register

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-02 | |
| Nama *Use Case* | | Register | |
| Tujuan | | Melakukan register sebagai pelanggan baru agar memiliki hak akses ke dalam sistem. | |
| Deskripsi | | Pelanggan baru membuat akun dengan mengisi *form*ulir registrasi, memasukkan nama lengkap, email, *Username*, dan *Password*. Sistem kemudian memvalidasi dan menyimpan informasi tersebut ke dalam *database*. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan baru ingin membuat akun dengan memasukkan informasi registrasi yang diperlukan. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin: bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : Setelah membuat akun, pelanggan baru akan melakukan *login* untuk masuk ke dalam sistem. | |
| Skenario utama | | |  |
| Pre kondisi :   * Menampilkan halaman registrasi. * Pengguna belum terdaftar dengan email yang sama. | | | |
| Aktor | | Reaksi sistem | |
| 1. Pengguna membuka halaman register. | |  | |
|  | | 1. Sistem menampilkan halaman register. | |
| 1. Pengguna mengisi data registrasi memasukkan nama lengkap, email, *username* dan *password*. | |  | |
|  | | 1. Sistem memvalidasi data yang dimasukkan. | |
| 1. Sistem menampilkan alert untuk *Field* *username* yang panjangnya kurang dari 8 karakter. | |
| 1. Sistem menampilkan alert jika *Field* *password* kurang dari 8 karakter atau lebih dari 16 karakter serta tidak berisi gabungan huruf besar, huruf kecil, angka, dan karakter. | |
| 1. Pengguna mengirimkan *form*ulir registrasi. | |  | |
|  | | 1. Sistem memeriksa email atau *username* yang sama serta memeriksa apakah ada *Field* yang tidak diisi. | |
| 1. Sistem menyimpan data pengguna baru ke dalam *database*. | |
| 1. Sistem menampilkan pesan sukses registrasi. | |
| *Post*-kondisi:   * Informasi pengguna divalidasi dan disimpan dalam *database*. * Akun pengguna dibuat dan siap untuk *login*. | | | |
| **Exception** | |  | |
| 1. *Field* nama lengkap tidak diisi : sistem akan memberikan alert bahwa *Field* nama lengkap harus diisi.  2. *Field* email tidak diisi : sistem akan memberikan alert bahwa *Field* email harus diisi.  3. *Field* *username* tidak diisi : Sistem akan memberian alert bahwa *Field* *username* harus diisi.  4. *Field* *Password* tidak diisi : sistem akan memberikan alert bahwa *Field* *password* harus diisi.  5. *Field* *username* kurang dari 8 karakter : sistem akan memberikan alert bahwa Panjang *Field* *username* minimal 8 karakter.  6. *Field* *Password* tidak memenuhi standar : Sistem akan menampilkan alert bahwa Panjang minimal *Field* *password* adalah 8 karakter dan Panjang maksimal 16 karakter. Serta harus terdiri dari gabungan huruf kecil, huruf besar, karakter, dan angka.  7. Mengisis *Username* atau email yang sudah didaftarkan : sistem akan menampilkan notifikasi akun gagal dibuat karena *username* atau email tersebut sudah terdaftar. | | | |

1. *Logout*

Tabel 4.3 Skenario Logout

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-03 | |
| Nama *Use Case* | | *Logout* | |
| Tujuan | | Pengguna ingin keluar dari akun website. | |
| Deskripsi | | Pengguna yang telah *login* (Admin, Pelanggan, Pemilik) menekan tombol *logout* untuk keluar dari sistem. Sistem kemudia mengakhiri sesi pengguna dan menampilkan halaman *login*. | |
| Nama aktor | | * Pelanggan * Admin * Pemilik | |
| Triger Event | | Pengguna yang telah *login* ingin keluar dari sistem. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan.  3. Pemilik : bertanggung jawab atas keberlangsungan usaha tersebut. | |
| Related *Use Case* | | *Login* : Setelah keluar dari sistem, pengguna bisa melakukan *login* untuk masuk ke dalam sistem Kembali. | |
| Skenario utama | | |  |
| Pre kondisi :  Pengguna telah melakukan *login* pada sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Admin, Pelanggan, Pemlik menekan tombol *logout*. | |  | |
|  | | * 1. Sistem Mengakhiri sesi pengguna. | |
| * 1. Sistem menampilkan halaman *login* dan notifikasi *logout* berhasil yang menandakan bahwa pengguna telah keluar dari sistem. | |
| *Post*-kondisi:   * Pengguna berhasil keluar dari sistem dan sesi pengguna berakhir. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan *logout* gagal. | | | |

1. Cari Menu

Tabel 4.4 Skenario Cari Menu

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-04 | |
| Nama *Use Case* | | Cari Menu | |
| Tujuan | | Pelanggan ingin mencari menu yang ada pada sistem. | |
| Deskripsi | | Pelanggan menggunakan halaman ini untuk mencari dan melihat menu yang tersedia berdasarkan kategori atau kata kunci. Sistem akan menerima input dari pelanggan dan menampilkan hasil pencarian. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan membuka halaman pencarian menu untuk melihat daftar menu melalui sistem. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : Pencarian hanya untuk pengguna yang masuk.  Lihat Menu : Setelah mencari, pelanggan bisa melihat deskripsi menu.  Chekout menu : Setelah menemukan menu yang diinginkan, pelanggan dapat langsung melakukan pemesanan. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah memasukkan data menu agar menu tersedia dalam sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Pelanggan membuka halaman pencarian menu. | |  | |
|  | | 1. Sistem menampilkan halaman pencarian menu. | |
| 1. Pelanggan memasukkan kata kunci atau memilih kategori. | |  | |
|  | | 1. Sistem memproses pencarian dengan mencocokan data menu dalam *database*. | |
| 1. Sistem menampilkan daftar menu yang sesuai dengan input pelanggan. | |
| *Post*-kondisi:   * Pelanggan mendapatkan daftar menu yang sesuai dengan pencarian. * Tidak ada hasil yang ditemukan, sistem menampilkan pesan bahwa menu tidak tersedia. | | | |
| **Exception** | |  | |
| 1. Tidak ada hasil yang ditemukan : sistem menampilkan pesan bahwa menu tidak tersedia. | | | |

1. Cari Menu (Detail Menu)

Tabel 4.5 Skenario Cari Menu (Detail Menu)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-05 | |
| Nama *Use Case* | | Detail Menu | |
| Tujuan | | Pelanggan dapat melihat detail menu yang tersedia pada sistem. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk melihat detail daftar menu. Sistem menampilkan informasi terkait menu secara lengkap seperti yang tersedia seperti nama menu, harga, deskripsi dan gambar. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin melihat detail menu melalui sistem. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Cari menu : pelanggan dapat mencari menu tertentu untuk melihat detail menu. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin sudah memasukan daftar menu yang tersedia dalam *database*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses Menu. | |  | |
|  | | 1. Menampilkan halaman daftar menu. | |
| 1. Menekan tombol detail. | |  | |
|  | | 1. Menampilkan detail menu. | |
| *Post*-kondisi:  Pelanggan berhasil menampilkan detail menu. | | | |
| **Exception** | |  | |
| 1. Detail menu gagal ditampilkan. | | | |

1. *Checkout* Menu

Tabel 4.6 Checkout Menu

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-06 | |
| Nama *Use* Case | | *Checkout* Menu | |
| Tujuan | | Pelanggan menyelesaikan transaksi pembelian dengan melakukan *checkout*. | |
| Deskripsi | | Halaman ini digunakan untuk pelanggan menambahkan menu kedalam keranjang dan pelanggan menekan tombol *checkout*. Sistem akan menampilkan ringkasan pesanan seperti menu yang dipilih, jumlah, harga, metode pembayaran. Sistem akan menyimpan pesanan dan memberi informasi terkait pembayaran. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan menekan tombol *checkout* setelah menambahkan menu kedalam keranjang. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pelanggan bisa menchekout menu.  Cari Menu : Sebelum *checkout* pelanggan mencari menu. | |
| Skenario utama | | |  |
| Pre kondisi :  Pelanggan telah memilih menu dalam keranjang. | | | |
| Aktor | | Reaksi sistem | |
| 1. Pelanggan membuka keranjang. | |  | |
|  | | 1. Sistem menampilkan ringkasan peasanan. | |
| 1. Pelangan menekan tombol “Konfirmasi Pesanan”. | |  | |
|  | | 1. Sistem menyimpan pesanan dan mengirim notifikasi ke admin. | |
| *Post*-kondisi:  Pesanan berhasil disimpan dalam sistem. | | | |
| **Exception** | |  | |
| 1. Menu gagal di*checkout*. | | | |

1. *Checkout* Menu (Memilih Menu)

Tabel 4.7 Checkout Menu (Memilih Menu)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-07 | |
| Nama *Use Case* | | Memilih Menu | |
| Tujuan | | Pelanggan memilih menu untuk melakukan transaksi pembelian. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk memilih menu. Sistem menampilkan terkait pilihan menu yang telah dipilih. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin memilih menu melalui sistem dan memasukkan ke keranjang. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pelanggan bisa memasukkan menu kedalam keranjang.  Cari Menu : Sebelum memilih menu pelanggan dapat mencari menu. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin sudah memasukan daftar menu yang tersedia dalam *database*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses menu. | |  | |
|  | | 1. Menampilkan halaman *checkout*. | |
| 1. Menekan tombol pilih menu atau “tambahkan keranjang’’. | |  | |
|  | | 1. Menampilkan pilihan yang telah dipilih di keranjang. | |
| *Post*-kondisi:  Pelanggan berhasil menambahkan menu dan menampilkan menu yang sudah dipilih. | | | |
| **Exception** | |  | |
| 1. Menu tidak dapat dipilih. | | | |

1. *Checkout* Menu (Detail Pesanan)

Tabel 4.8 Checkout Menu (Detail Pesanan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-08 | |
| Nama *Use Case* | | Detail Pesanan | |
| Tujuan | | Pelanggan dapat melihat detail menu yang tersedia pada sistem. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk melihat detail pesanan. Sistem menampilkan informasi terkait pesanan seperti nama lengkap, total harga, alamat dan keterangan pembayaran. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin melihat detail pesanan melalui sistem. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pelanggan bisa memesan menu dan melihat detail pesanan yang akan dichekout. | |
| Skenario utama | | |  |
| Pre kondisi :  Pelanggan sudah mengklik tombol *checkout*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses keranjang menu dan pilih menu yang akan di chekout. | |  | |
|  | | 1. Menampilkan halaman keranjang. | |
| 1. Menekan tombol *checkout*. | |  | |
|  | | 1. Menampilkan detail pesanan. | |
| *Post*-kondisi:  Pelanggan berhasil menampilkan detail pesanan. | | | |
| **Exception** | |  | |
| 1. Menu tidak dapat dichekout. | | | |

1. Pembayaran

Tabel 4.9 Pembayaran

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-09 | |
| Nama *Use Case* | | Pembayaran | |
| Tujuan | | Pelanggan menyelesaikan transaksi pembelian dengan melakukan pembayaran. | |
| Deskripsi | | Halaman ini digunakan untuk pelanggan melakukan pembayaran. Sistem akan menampilkan metode pembayaran. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan menekan tombol “Bayar Sekarang” setelah melakukan *checkout*. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pelanggan bisa melakukan transaksi pembayaran.  *Checkout* Pesanan : Sebelum pembayaran, pelanggan menyelesaikan proses *checkout* terlebih dahulu. | |
| Skenario utama | | |  |
| Pre kondisi :   * Pelanggan telah menyelesaikan proses *checkout*. * Admin sudah memasukan daftar metode pembayaran yang tersedia. | | | |
| Aktor | | Reaksi sistem | |
| 1. Menekan tombol “Bayar Sekarang”. | |  | |
|  | | 1. Menampilkan metode pembayaran yang tersedia. | |
| 1. Memilih metode pembayaran. | |  | |
|  | | 1. Menampilkan instruksi pembayaan. | |
| 1. Menginput nominal total pembayaran. | |  | |
|  | | 1. Memverifikasi transaksi. | |
| *Post*-kondisi:   * Status pesanan diperbarui. * Pelanggan menerima notifikassi pembayaran sukses. * Admin menerima informasi pesanan untuk diproses. | | | |
| **Exception** | |  | |
| 1. Membatalkan transaksi dan pesanan tetap dalam status “Menunggu Pembayaran”. 2. Menampilkan proses pembayaran gagal. | | | |

1. Pembayaran (Memilih Metode Pembayaran)

Tabel 4.10 Pembayaran (Memilih Metode Pembayaran)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-10 | |
| Nama *Use Case* | | Memilih Metode Pembayaran | |
| Tujuan | | Pelanggan menyelesaikan pembayaran dengan Memilih Metode Pembayaran. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk memilih mtode pembayaran. Sistem menampilkan informasi terkait metode pembayaran dan melakukan pembayaran. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan akan melakukan pembayaran dan memilih metode pembayaran terlebih dahulu. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pelanggan bisa memilih metode pembayaran.  *Checkout* Pesanan : Sebelum memilih metode pembayaran, pelanggan menyelesaikan proses *checkout* terlebih dahulu. | |
| Skenario utama | | |  |
| Pre kondisi :   * Pelanggan telah menyelesaikan proses *checkout*. * Admin sudah memasukan metode pembayaran yang tersedia. | | | |
| Aktor | | Reaksi sistem | |
| 1. Menekan tombol “Pilih Metode pembayaran”. | |  | |
|  | | 1. Menampilkan metode pembayaran yang tersedia. | |
| 1. Menyimpan pilihan pelanggan. | |
| 1. Menampilkan nomor rekening untuk pembayaran. | |
| *Post*-kondisi:   * Pelanggan berhasil menampilkan metode pembayaran. * Pelanggan berhasil memilih metode pembayaran yang tersedia. | | | |
| **Exception** | |  | |
| 1. Metode pembayaran sedang gangguan : menampilkan pesan error dan memilih metode pembayaran yang lain. | | | |

1. Memberikan *Review*

Tabel 4.11 Memberikan Review

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-11 | |
| Nama *Use Case* | | Memberikan *Review* | |
| Tujuan | | Pelanggan dapat memberikan *review* untuk mendukung Hanna *Catering*. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk me*review* dan merating pesanan yang sudah selesai dipesan. Sistem dapat menampilkan ulasan *review* dan rating serta dapat dilihat oleh admin dan pelanggan lain. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin memberikan *review*. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : Setelah melakukan *login* pelanggan bisa memberikan *review*.  *Checkout* Pesanan : Pelanggan dapat mengakses halaman *review* dari riwayat pesanan dan bisa meriview setelah melakukan pesanan. | |
| Skenario utama | | |  |
| Pre kondisi :   * Pelanggan telah menyelesaikan proses pemesanan. * Menampilkan halaman *review*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Membuka riwayat pesanan. | |  | |
|  | | 1. Menampilkan detail pesanan. | |
| 1. Menekan tombol “Beri *Review*”. | |  | |
|  | | * 1. Menampilkan *form*ulir *review*. | |
| 1. Memberikan rating. | |  | |
|  | | * 1. Menyimpan data *review* sementara. | |
| 1. Menekan “kirim *Review*”. | |  | |
|  | | 1. Menyimpan *review* secara permanen dan menampilkannya. | |
| *Post*-kondisi:   * *Review* tersimpan dan dapat dilihat oleh pelanggan lain. * Admin dapat menanggapi ulasan. | | | |
| **Exception** | |  | |
| 1. Terjadi kesalahan sistem saat mengirim *review* : Menampilkan pesan error dan meminta pelanggan mencoba lagi. | | | |

1. Memberikan *Review* (Tambah *Review*)

Tabel 4.12 Memberikan Review (Tambah Review)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-12 | |
| Nama *Use Case* | | Tambah *Review* | |
| Tujuan | | Pelanggan dapat menambahkan *review* untuk mendukung Hanna *Catering*. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk menambahkan *review* setelah memesan menu *Catering*. Sistem akan menampilkan hasil *review* pada halaman *review* untuk membantu pelanggan lain dalam mengambil Keputusan. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin memberikan *review* terkait pengalamannya memesan *Catering* pada Hanna *Catering*. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : Setelah melakukan *login* pelanggan bisa memberikan *review*.  *Checkout* Pesanan : Pelanggan dapat mengakses halaman *review* dari riwayat pesanan dan bisa meriview setelah melakukan pesanan. | |
| Skenario utama | | |  |
| Pre kondisi :   * Pelanggan sudah *login* akun di sistem Hana *Catering*. * Pelanggan telah menyelesaikan pemesanan dan menerima pesanan dari Hana *Catering*. * Sistem harus menyediakan halaman untuk menampilkan *review*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Membuka riwayat pesanan. | |  | |
|  | | 1. Menampilkan detail pesanan. | |
| 1. Menekan tombol “Beri *Review*”. | |  | |
|  | | * 1. Menampilkan *form*ulir *review*. | |
| 1. Mengetik ulasan lalu menekan “kirim *Review*”. | |  | |
|  | | 1. Menyimpan *review* secara permanen dan menampilkannya | |
| *Post*-kondisi:   * *Review* yang telah ditambahkan akan tersimpan di sistem. * *Review* akan ditampilkan pada halaman *review* untuk pelanggan lain. * Admin dapat menanggapi ulasan. | | | |
| **Exception** | |  | |
| 1. Terjadi kesalahan sistem saat tambah *review* : Menampilkan pesan error dan meminta pelanggan mencoba lagi. | | | |

1. Memberikan *Review* (Rate *Review*)

Tabel 4.13 Memberikan Review (Rate Review)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-13 | |
| Nama *Use Case* | | Rate *Review* | |
| Tujuan | | Pelanggan dapat memberikan Rating untuk mendukung Hanna *Catering*. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk rating setelah memesan makanan. Sistem akan menampilkan hasil rating pada halaman *review* untuk membantu pelanggan lain dalam mengambil Keputusan. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin memberikan rating terkait pengalamannya memesan *Catering* pada Hanna *Catering* | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : Setelah melakukan *login* pelanggan bisa memberikan *review*.  *Checkout* Pesanan : Pelanggan dapat mengakses halaman *review* dari riwayat pesanan dan bisa meriview setelah melakukan pesanan.  Tambah *Review* : Pelanggan dapat mengakses rating bersamaan dengan tambah *review*. | |
| Skenario utama | | |  |
| Pre kondisi :   * Pelanggan telah menyelesaikan pemesanan dan menerima pesanan dari Hana *Catering*. * Sistem sudah menyediakan halaman untuk menampilkan rate *review*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Membuka riwayat pesanan. | |  | |
|  | | 1. Menampilkan detail pesanan. | |
| 1. Menekan tombol “Beri Rating”. | |  | |
|  | | * 1. Menampilkan *form*ulir *review*. | |
| 1. Memberikan rating. | |  | |
|  | | * 1. Menyimpan data *review* sementara. | |
| 1. Menekan “kirim Rating”. | |  | |
|  | | 1. Menyimpan *review* secara permanen dan menampilkannya. | |
| *Post*-kondisi:   * *Review* yang telah ditambahkan akan tersimpan di sistem. * *Review* akan ditampilkan pada halaman *review* untuk pelanggan lain. * Sistem memperbarui jumlah rating pada *review* yang dinilai. | | | |
| **Exception** | |  | |
| 1. Terjadi kesalahan sistem saat tambah *review* : Menampilkan pesan error dan meminta pelanggan mencoba lagi. | | | |

1. *Chat* Admin

Tabel 4.14 Chat Admin

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-14 | |
| Nama *Use Case* | | *Chat* Admin | |
| Tujuan | | Membantu pelanggan ketika mengalami kesulitan saat proses pemesanan. | |
| Deskripsi | | Halaman ini digunakan oleh pelanggan untuk menghubungi admin melalui *chat* secara langsung. Sistem akan menampilkan menu *chat* untuk membantu pelanggan memberikan informasi atau bantuan. | |
| Nama aktor | | Pelanggan | |
| Triger Event | | Pelanggan ingin menanyakan terkait permasalahan atau kendala dan ingin berkomunikasi langsung melalui *chat* admin. | |
| Stakeholder | | 1. Pelanggan : Pengguna akhir yang menggunakan layanan pembelian.  2. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pelanggan bisa melakukan *chat* dengan admin. | |
| Skenario utama | | |  |
| Pre kondisi :  Sistem sudah menyediakan fitur *chat*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Masuk ke halaman "*Chat* Admin”. | |  | |
|  | | 1. Menampilkan *chat*. | |
| 1. Mengetik pesan dan mengirimkannya ke admin. | |  | |
|  | | * 1. Menyimpan dan meneruskan pesan ke admin. | |
| *Post*-kondisi:  Percakapan antara pelanggan dan admin tersimpan dalam sistem. | | | |
| **Exception** | |  | |
| 1. Menampilkan gagal pesan dikirim. 2. Admin sedang offline : Menampilkan pesan admin akan membalas dalam waktu tertentu. | | | |

1. Kelola Pesanan

Tabel 4.15 Kelola Pesanan

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-15 | |
| Nama *Use Case* | | Kelola Pesanan | |
| Tujuan | | Untuk mengelola informasi dari data pesanan. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk mengelola data pesanan. Sistem memungkinkan pengguna untuk melihat, menambah, mengedit, atau menghapus informasi pesanan. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin mengelola data pesanan melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* admin bisa mengakses menu Kelola pesanan. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses menu kelola data pesanan. | |  | |
|  | | 1. Menampilkan data pesanan. | |
| *Post*-kondisi:  Admin dapat melakukan kelola data pesanan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan tidak dapat mengakses Kelola pesanan | | | |

1. Kelola Pesanan (Lihat Pesanan)

Tabel 4.16 Kelola Pesanan (Lihat Pesanan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-16 | |
| Nama *Use Case* | | Lihat Pesanan | |
| Tujuan | | Untuk mengelola data pesanan. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk melihat data pesanan. Sistem menampilkan daftar pesanan dan menyediakan opsi untuk melihat detail pesanan yang dipilih. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin melihat data pesanan yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data pesanan : admin sudah mengakses Kelola data pesanan dan bisa melihat data. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola pesanan. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses menu kelola data pesanan. | |  | |
|  | | 1. Menampilkan halaman Kelola data pesanan. | |
| 1. Klik tombol detail. | |  | |
|  | | 1. Menampilkan detail data pesanan yang dipilih. | |
| *Post*-kondisi:  Admin berhasil melihat data pesanan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data gagal ditampilkan. | | | |

1. Kelola Pesanan (Update Status Pesanan)

Tabel 4.17 Kelola Pesanan (Update Status Pesanan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-17 | |
| Nama *Use Case* | | Update Status Pesanan | |
| Tujuan | | Untuk mengelola data pesanan terkait update status pesanan pelanggan. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk mengupdate data pesanan. Sistem menampilkan status pesanan dan menyediakan opsi untuk update status pesanan yang dipilih. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin mengupdate data pesanan yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data pesanan : admin sudah mengakses Kelola data pesanan dan bisa mengupdate status pesanan. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola pesanan. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses menu kelola data pesanan. | |  | |
|  | | 1. Menampilkan halaman Kelola data pesanan. | |
| 1. Klik tombol status pesanan. | |  | |
|  | | 1. Menampilkan update status data pesanan yang dipilih. | |
| 1. Pilih proses pesanan “Selesai, diproses, ditunda”. | |  | |
|  | | 1. Menampilkan proses pesanan. | |
| *Post*-kondisi:  Admin berhasil mengupdate data pesanan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data gagal diupdate. | | | |

1. Kelola Pesanan (Hapus Menu)

Tabel 4.18 Skenario Kelola Pesanan (Hapus Pesanan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-18 | |
| Nama *Use Case* | | Hapus Pesanan | |
| Tujuan | | Admin dapat hapus pesanan pelanggan yang tersedia pada sistem. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk menghapus data pesanan pelanggan. Sistem menyediakan *form* untuk menghapus data pesanan dan menyimpan perubahan dalam *database*. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin menghapus data pesanan yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data pesanan: admin sudah mengakses Kelola data menu dan bisa menghapus data. | |
| Skenario utama | | |  |
| Pre kondisi :   * Pelanggan sudah melakukan pemesanan catering dan masuk dalam *database*. * Admin telah melakukan *login* pada sistem dan mengakses Kelola menu. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data pesanan pelanggan. | |  | |
|  | | 1. Menampilkan halaman Kelola data pesanan. | |
| 1. Menekan tombol hapus. | |  | |
|  | | 1. Menampilkan informasi apakah data ingin dihapus. | |
| 1. Melakukan hapus data pesanan. | |  | |
|  | | 1. Menyimpan data ke *database* dan menampilkan notifikasi data berhasil dihapus. | |
| *Post*-kondisi:  Admin berhasil menghapus data pesanan dalam sistem. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data gagal dihapus | | | |

1. Kelola Menu

Tabel 4.19 Kelola Menu

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-19 | |
| Nama *Use Case* | | Kelola Menu | |
| Tujuan | | Untuk mengelola informasi dari data menu *Catering.* | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk mengelola data menu caterig. Sistem memungkinkan pengguna untuk melihat, menambah, mengedit, atau menghapus informasi menu *Catering*. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin mengelola data menu melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* admin bisa mengakses Kelola menu. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola menu. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data menu *Catering*. | |  | |
|  | | 1. Menampilkan data menu *Catering* | |
| *Post*-kondisi:  Admin dapat melakukan kelola data menu *Catering*. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan tidak dapat mengakses Kelola menu | | | |

1. Kelola Menu (Tambah Menu)

Tabel 4.20 Kelola Menu (Tambah Menu)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-20 | |
| Nama *Use Case* | | Tambah Menu | |
| Tujuan | | Untuk mengelola informasi dari data menu *Catering* terkait menambahkan data menu *Catering*. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk menambahkan data menu *Catering*. Sistem menyediakan *form* untuk pengisian data menu baru dan menyimpanya ke *database*. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin menambahkan data menu melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data menu : admin sudah mengakses Kelola data menu dan bisa menambahkan data. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola menu. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data menu *Catering*. | |  | |
|  | | 1. Menampilkan halaman Kelola data menu. | |
| 1. Klik tombol tambah. | |  | |
|  | | 1. Menampilkan *form* untuk tambah data. | |
| 1. Mengisi *form* data menu. | |  | |
|  | | 1. Melakukan validasi data yang diisi. | |
| 1. Klik tombol simpan. | |  | |
|  | | 1. Menyimpan data ke *database* dan menamplkan notifikasi data berhasil disimpan. | |
| *Post*-kondisi:  Admin berhasil menambahkan data menu *Catering*. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data menu gagal ditambahkan. 2. *Field* nama menu tidak diisi : Sistem akan memberikan alert bahwa *Field* nama menu harus diisi. 3. *Field* Deskripsi tidak diisi : Sistem akan memberikan alert bahwa *Field* deskripsi harus diisi. 4. *Field* Harga tidak diisi : Sistem akan memberikan alert bahwa *Field* harga harus diisi. | | | |

1. Kelola Menu (Edit Menu)

Tabel 4.21 Kelola Menu (Edit Menu)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-21 | |
| Nama *Use Case* | | Edit Menu | |
| Tujuan | | Untuk mengelola informasi dari data menu *Catering* terkait edit data menu *Catering*. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk mengedit data menu *Catering*. Sistem menyediakan *form* untuk mengedit data menu dan menyimpan perubahan ke dalam *database*. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin mengedit data menu yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data menu : admin sudah mengakses Kelola data menu dan bisa mengedit data. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola menu. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data menu *Catering*. | |  | |
|  | | 1. Menampilkan halaman Kelola data menu. | |
| 1. Klik tombol edit. | |  | |
|  | | 1. Menampilkan *form* untuk edit data. | |
| 1. Melakukan edit data menu. | |  | |
|  | | 1. Melakukan validasi data yang diisi. | |
| 1. Klik tombol simpan. | |  | |
|  | | 1. Menyimpan data ke *database* dan menampilkan notifikasi data berhasil disimpan. | |
| *Post*-kondisi:  Admin berhasil mengedit data menu *Catering* dalam sistem. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data menu gagal diedit. | | | |

1. Kelola Menu (Detail Menu)

Tabel 4.22 Skenario Kelola Menu (Detail Menu)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-22 | |
| Nama *Use Case* | | Detail Menu | |
| Tujuan | | Admin dapat melihat detail menu yang tersedia pada sistem. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk melihat detail daftar menu. Sistem menampilkan informasi terkait menu secara lengkap seperti yang tersedia seperti nama menu, harga, deskripsi dan gambar. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin melihat detail menu melalui sistem. | |
| Stakeholder | | 1. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data menu : admin sudah mengakses Kelola data menu dan bisa melihat detail data. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin sudah memasukan daftar menu yang tersedia dalam *database*. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data menu *Catering*. | |  | |
|  | | 1. Menampilkan halaman Kelola data menu. | |
| 1. Menekan tombol detail | |  | |
|  | | 1. Menampilkan detail menu | |
| *Post*-kondisi:  Admin berhasil menampilkan detail menu. | | | |
| **Exception** | |  | |
| 1. Detail menu gagal ditampilkan. | | | |

1. Kelola Menu (Hapus Menu)

Tabel 4.23 Skenario Kelola Menu (Hapus Menu)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-23 | |
| Nama *Use Case* | | Hapus Menu | |
| Tujuan | | Admin dapat hapus menu yang tersedia pada sistem. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk menghapus data menu *Catering*. Sistem menyediakan *form* untuk menghapus data menu dan menyimpan perubahan dalam *database*. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin menghapus data menu yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data menu : admin sudah mengakses Kelola data menu dan bisa menghapus data. | |
| Skenario utama | | |  |
| Pre kondisi :   * Admin sudah memasukan daftar menu yang tersedia dalam *database*. * Admin telah melakukan *login* pada sistem dan mengakses Kelola menu. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data menu *Catering*. | |  | |
|  | | 1. Menampilkan halaman Kelola data menu. | |
| 1. Menekan tombol hapus. | |  | |
|  | | 1. Menampilkan informasi apakah data ingin dihapus. | |
| 1. Melakukan hapus data menu. | |  | |
|  | | 1. Menyimoan data ke *database* dan menampilkan notifikasi data berhasil dihapus. | |
| *Post*-kondisi:  Admin berhasil menghapus data menu dalam sistem. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data gagal dihapus | | | |

1. Kelola Pembayaran

Tabel 4.24 Skenario Kelola Pembayaran

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-24 | |
| Nama *Use Case* | | Kelola Pembayaran | |
| Tujuan | | Untuk mengelola informasi dari data pembayaran. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk mengelola data pembayaran. Sistem memungkinkan pengguna untuk melihat, menginput dan mengedit informasi pembayaran. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin mengelola data pembayaran melalui sistem. | |
| Stakeholder | | 1. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* admin bisa mengakses Kelola pembayaran. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola data pembayaran. | |  | |
|  | | 1. Menampilkan data pembayaran. | |
| *Post*-kondisi:  Admin dapat melakukan kelola data pembayaran. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan tidak dapat mengakses Kelola pembayaran. | | | |

1. Kelola Pembayaran (Edit Pembayaran)

Tabel 4.25 Kelola Menu (Edit Pembayaran)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-25 | |
| Nama *Use Case* | | Edit Pembayaran | |
| Tujuan | | Untuk mengelola informasi dari data pembayaran terkait edit data pembayaran. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk mengedit data pembayaran. Sistem menyediakan *form* untuk mengedit data pembayaran seperti update pelunasan dan menyimpan perubahan ke dalam *database*. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin mengedit data pembayaran yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data pembayaran : admin sudah mengakses Kelola data pembayaran dan bisa mengedit data. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola pembayaran. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses Kelola data pembayaran. | |  | |
|  | | 1. Menampilkan halaman Kelola data pembayaran. | |
| 1. Klik tombol edit. | |  | |
|  | | 1. Menampilkan *form* untuk edit data. | |
| 1. Melakukan edit data pembayaran. | |  | |
|  | | 1. Melakukan validasi data yang diisi. | |
| 1. Klik tombol simpan. | |  | |
|  | | 1. Menyimpan data ke *database* dan menampilkan notifikasi data berhasil disimpan. | |
| *Post*-kondisi:  Admin berhasil mengedit data pembayaran dalam sistem. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data gagal diedit atau diupdate. | | | |

1. Kelola Pembayaran (Lihat Pembayaran)

Tabel 4.26 Kelola Pesanan (Lihat Pesanan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-26 | |
| Nama *Use Case* | | Lihat Pembayaran | |
| Tujuan | | Untuk mengelola data pembayaran. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk melihat data pembayaran. Sistem menampilkan daftar pembayaran pelanggan dan menyediakan opsi untuk melihat detail pembayaran yang dipilih. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin melihat data pembayaran yang sudah ada melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | Kelola data pesanan : admin sudah mengakses Kelola data pembayaran dan bisa melihat data. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem dan mengakses Kelola pembayaran. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses menu kelola data pembayaran. | |  | |
|  | | 1. Menampilkan halaman Kelola data pembayaran. | |
| 1. Klik tombol detail. | |  | |
|  | | 1. Menampilkan detail data pembayaran yang dipilih. | |
| *Post*-kondisi:  Admin berhasil melihat data pesanan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan data gagal ditampilkan. | | | |

1. Menanggapi *Chat* Pelanggan

Tabel 4.27 Menangapi Chat Pelanggan

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-27 | |
| Nama *Use Case* | | Menanggapi *Chat* Pelanggan | |
| Tujuan | | Membantu pelanggan ketika mengalami kesulitan saat proses pemesanan. | |
| Deskripsi | | Halaman ini digunakan oleh admin untuk menanggapi *chat* pelanggan. Sistem memungkinkan admin untuk melihat dan membalas *chat* pelanggan. | |
| Nama aktor | | Admin | |
| Triger Event | | Admin ingin menanggapi *chat* melalui sistem. | |
| Stakeholder | | Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* admin bisa mengakses *chat* pelanggan. | |
| Skenario utama | | |  |
| Pre kondisi :  Admin telah melakukan *login* pada sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses *chat* pelanggan. | |  | |
|  | | 1. Menampilkan *chat* pelanggan. | |
| 1. Membalas pesan dan mengirimkannya ke pelanggan. | |  | |
|  | | * 1. Menyimpan dan meneruskan pesan ke pelanggan. | |
| *Post*-kondisi:  Admin dapat menanggapi *chat* pelanggan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan tidak dapat mengakses *chat* pelanggan. | | | |

1. Kelola Laporan

Tabel 4.28 Kelola Laporan

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-28 | |
| Nama *Use Case* | | Kelola Laporan | |
| Tujuan | | Untuk mengelola informasi dari data laporan | |
| Deskripsi | | Halaman ini digunakan oleh admin atau pemilik untuk mengelola laporan. Sistem memungkinkan pengguna untuk melihat laporan. | |
| Nama aktor | | Admin dan Pemilik | |
| Triger Event | | Admin dan pemilik ingin mengelola data laporan melalui sistem. | |
| Stakeholder | | * 1. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan.   2. Pemilik : bertanggung jawab atas keberlangsungan usaha tersebut. | |
| Related *Use Case* | | *Login* : setelah melakukan *login* pengguna bisa mengakses Kelola laporan. | |
| Skenario utama | | |  |
| Pre kondisi :  Pengguna telah melakukan *login* pada sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses kelola laporan. | |  | |
|  | | 1. Menampilkan laporan. | |
| *Post*-kondisi:  Pengguna dapat melakukan kelola laporan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan tidak dapat mengakses Kelola laporan. | | | |

1. Kelola Laporan (Lihat Laporan)

Tabel 4.29 Kelola Laporan(Lihat Laporan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-29 | |
| Nama *Use Case* | | Kelola Laporan | |
| Tujuan | | Untuk melihat informasi dari data laporan. | |
| Deskripsi | | Halaman ini digunakan oleh admin atau pemilik untuk melihat laporan. Sistem menampilkan laporan dan menyediakan opsi untuk melihat detail laporan. | |
| Nama aktor | | Admin dan Pemilik | |
| Triger Event | | Admin atau pemilik ingin melihat laporan yang sudah ada melalui sistem. | |
| Stakeholder | | 1. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. 2. Pemilik : bertanggung jawab atas keberlangsungan usaha tersebut. | |
| Related *Use Case* | | Kelola laporan : admin sudah mengakses Kelola laporan dan bisa melihat laporan. | |
| Skenario utama | | |  |
| Pre kondisi :  Pengguna telah melakukan *login* pada sistem dan mengakses Kelola laporan. | | | |
| Aktor | | Reaksi sistem | |
| 1. Mengakses Kelola laporan. | |  | |
|  | | 1. Menampilkan halaman Kelola laporan. | |
| 1. Klik tombol detail | |  | |
|  | | 1. Menampilkan detail laporan yang dipilih. | |
| *Post*-kondisi:  Pengguna berhasil melihat lapoan. | | | |
| **Exception** | |  | |
| 1. Menampilkan pesan laporan gagal ditampilkan. | | | |

1. Kelola Laporan (Print Laporan)

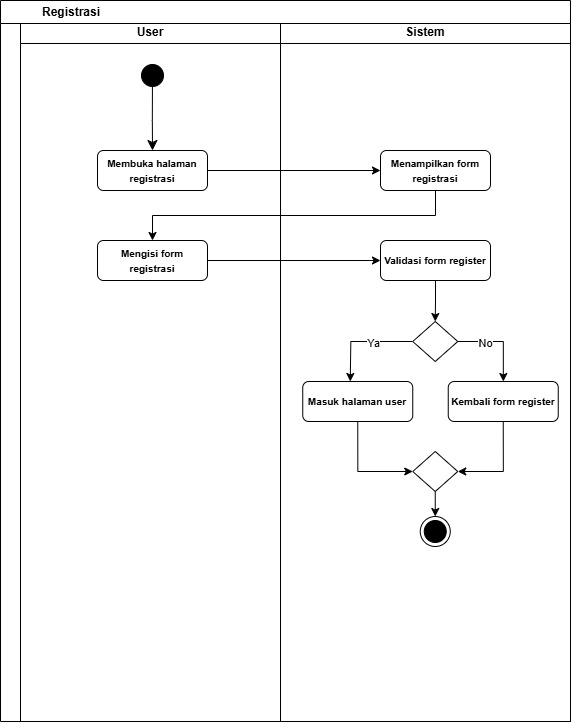
Tabel 4.30 Kelola Laporan(Print Laporan)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Identifikasi | | |
| No. *Use Case* | | UC-28 | |
| Nama *Use Case* | | Print Laporan | |
| Tujuan | | Laporan dapat diprint sesuai kebutuhan. | |
| Deskripsi | | Pemilik atau admin memilih laporan yang akan dicetak melalui sistem. Setelah laporan dipilih, sistem akan menyiapkan file dalam *form*at cetak dan mengirimkannya ke printer. | |
| Nama aktor | | Admin dan Pemilik | |
| Triger Event | | Admin atau pemilik ingin mendapatkan salinan fisik laporan dalam bentuk cetakan untuk keperluan analisis atau dokumentasi. | |
| Stakeholder | | 1. Admin : bertanggung jawab untuk mengelola sistem dalam suatu layanan. 2. Pemilik : bertanggung jawab atas keberlangsungan usaha tersebut. | |
| Related *Use Case* | | Kelola laporan : admin sudah mengakses Kelola laporan dan bisa print laporan. | |
| Skenario utama | | |  |
| Pre kondisi :   * Laporan tersedia di dalam sistem. * Halaman pengelolaan laporan sudah diakses. * Printer terhubung dengan sistem. | | | |
| Aktor | | Reaksi sistem | |
| 1. Pemilik/Admin membuka halaman laporan. | |  | |
|  | | 1. Sistem menampilkan daftar laporan yang tersedia. | |
| 1. Pemilik/Admin memilih laporan yang akan dicetak. | |  | |
|  | | 1. Sistem memuat detail laporan yang dipilih. | |
| 1. Sistem menyiapkan laporan dalam *form*at cetak. | |
| 1. Pemilik/Admin menekan tombol "*Print*". | |  | |
|  | | 1. Sistem mengirimkan data laporan ke printer. | |
| 1. Sistem memberikan notifikasi bahwa proses berhasil. | |
| *Post*-kondisi:   * Jika berhasil, laporan dicetak dengan *form*at yang benar. * Jika gagal, sistem memberikan notifikasi kesalahan. | | | |
| **Exception** | |  | |
| 1. Laporan tidak tersedia. 2. Printer tidak terhubung. 3. Kertas printer habis. 4. Kesalahan *form*at laporan. 5. Menampilkan pesan. 6. Laporan gagal diprint. | | | |

## **Activity Diagram**

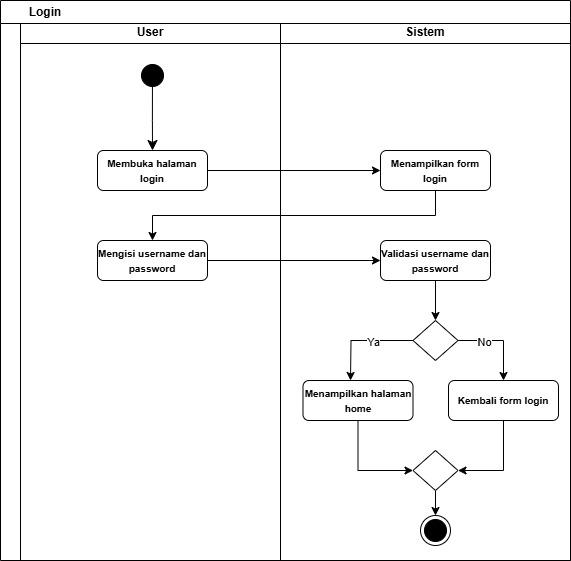
Berikut adalah activity diagram dari sistem informasi manajemen pemesana beserta penerapan CRM berbasis website untuk meningkatkan kepuasan pelanggan pada hanna catering :

* 1. Activity Diagram Registrasi



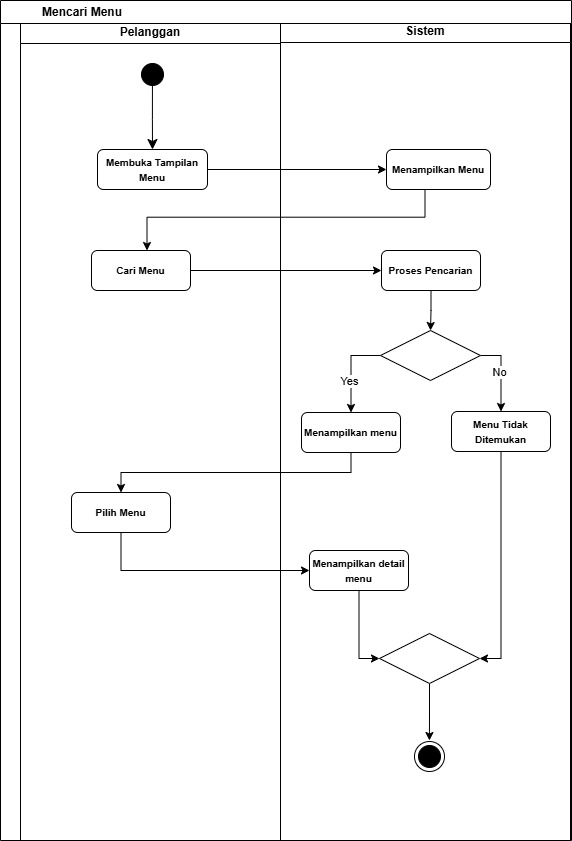
Gambar 1. Activity Diagram Registrasi

* 1. Activity Diagram Login



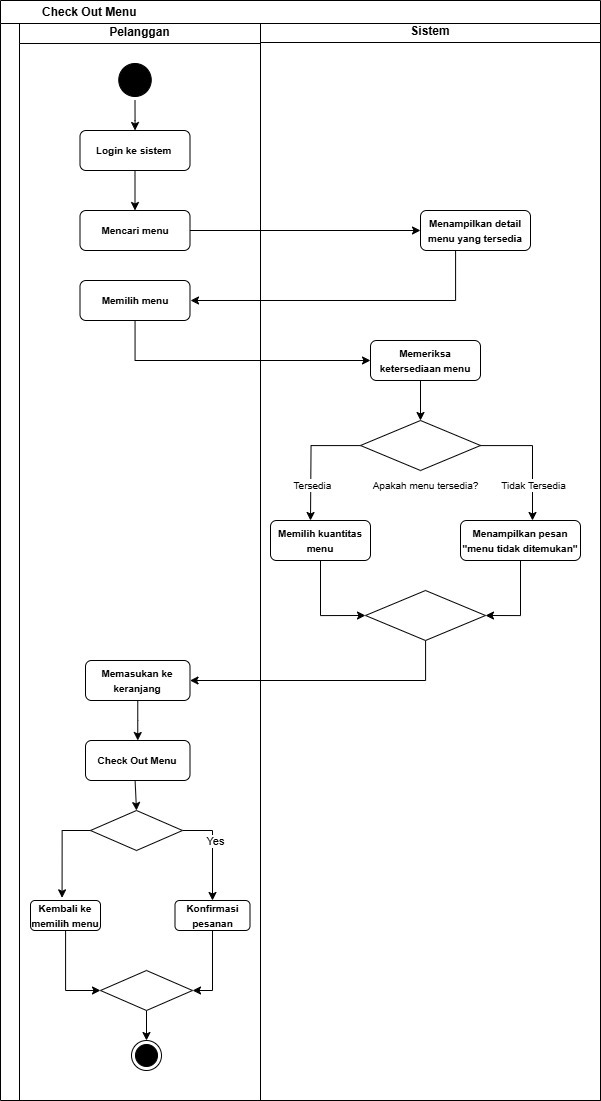
Gambar 2. Activity Diagram Login

* 1. Activity Diagram Mencari Menu



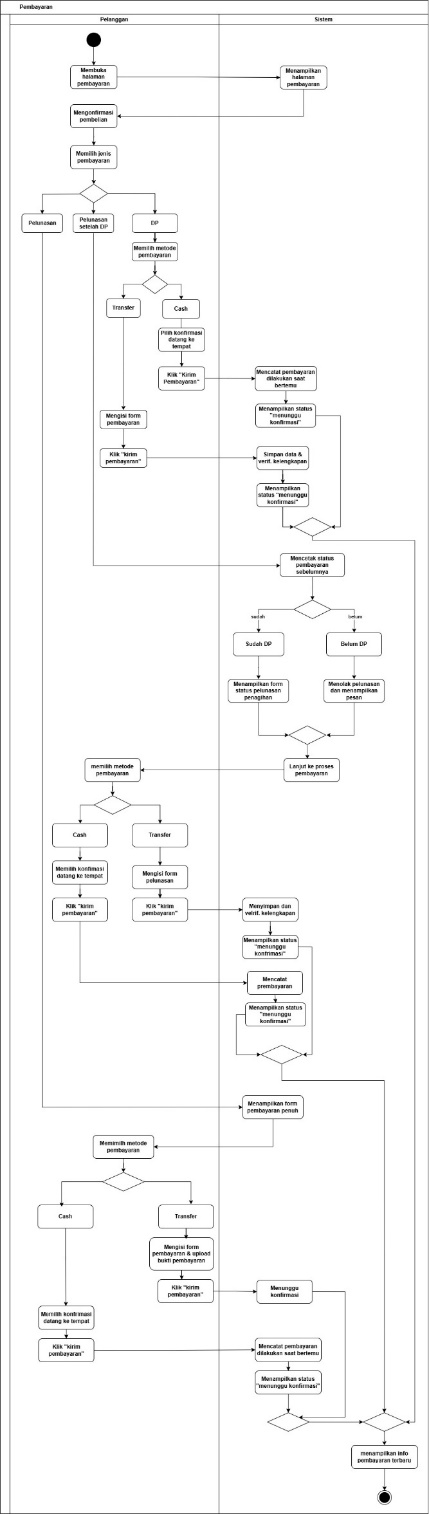
Gambar 3. Activity Diagram Mencari Menu

* 1. Activity Diagram Chekout Menu

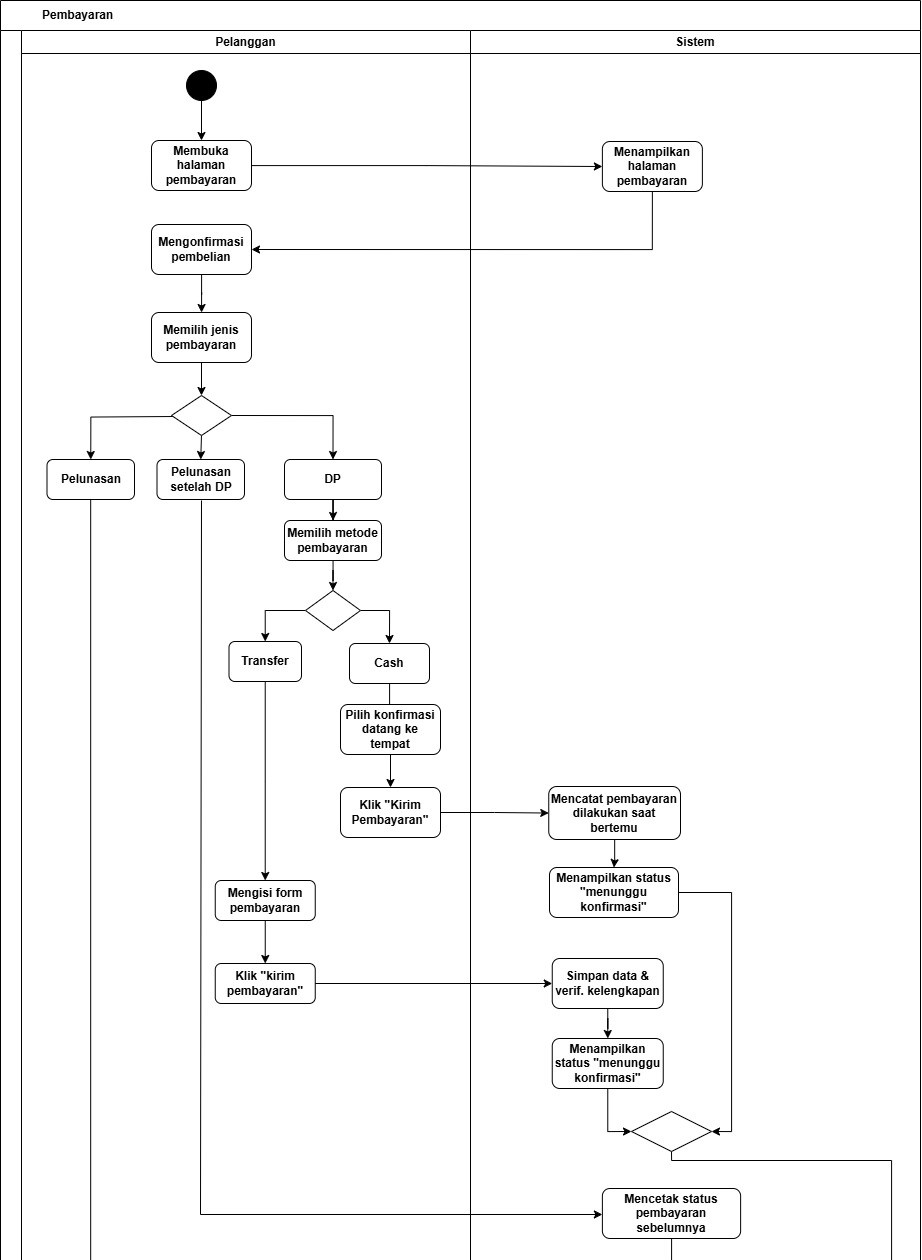


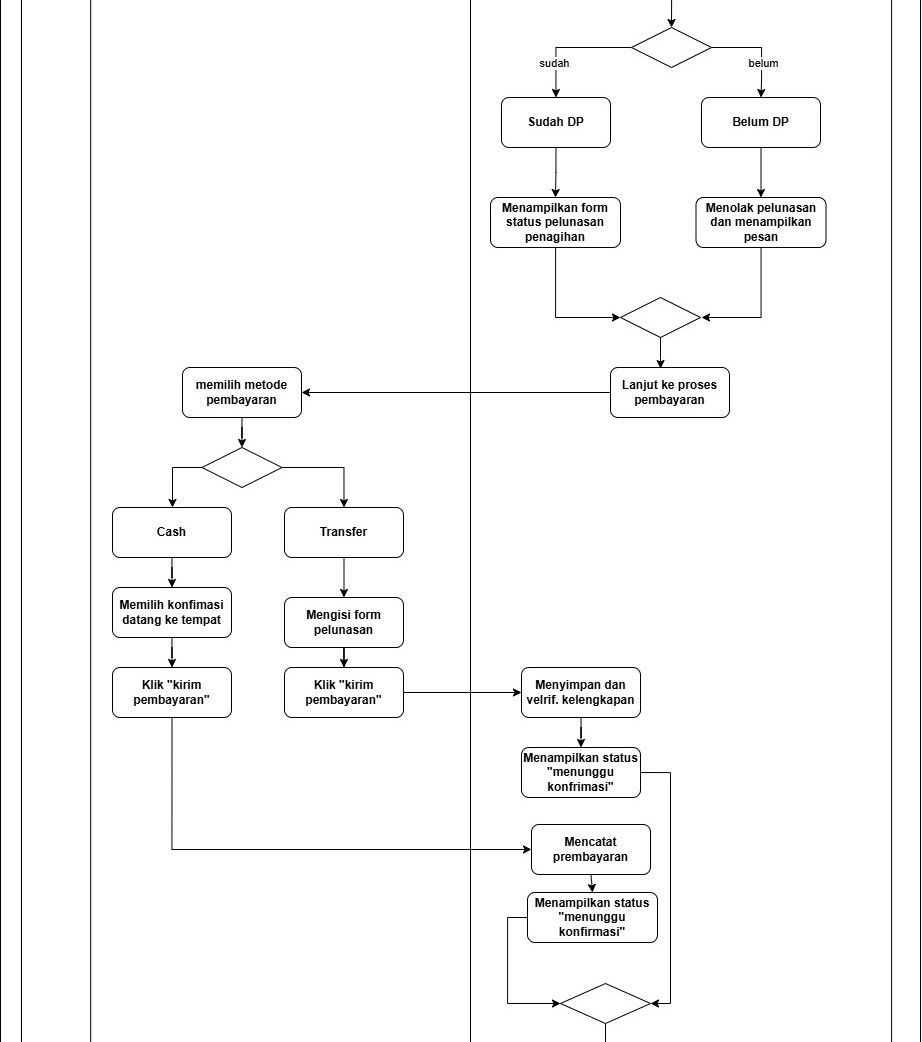
Gambar 4. Activity Diagram Chekout Menu

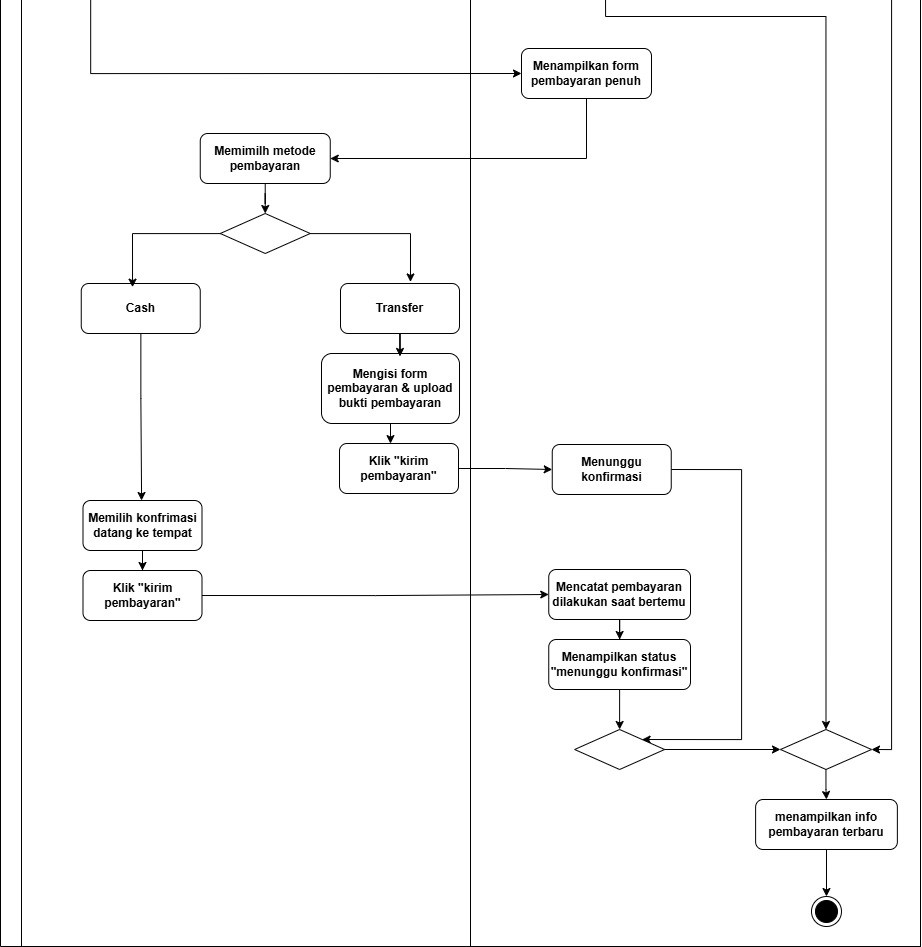
* 1. Activity Diagram Pembayaran

****

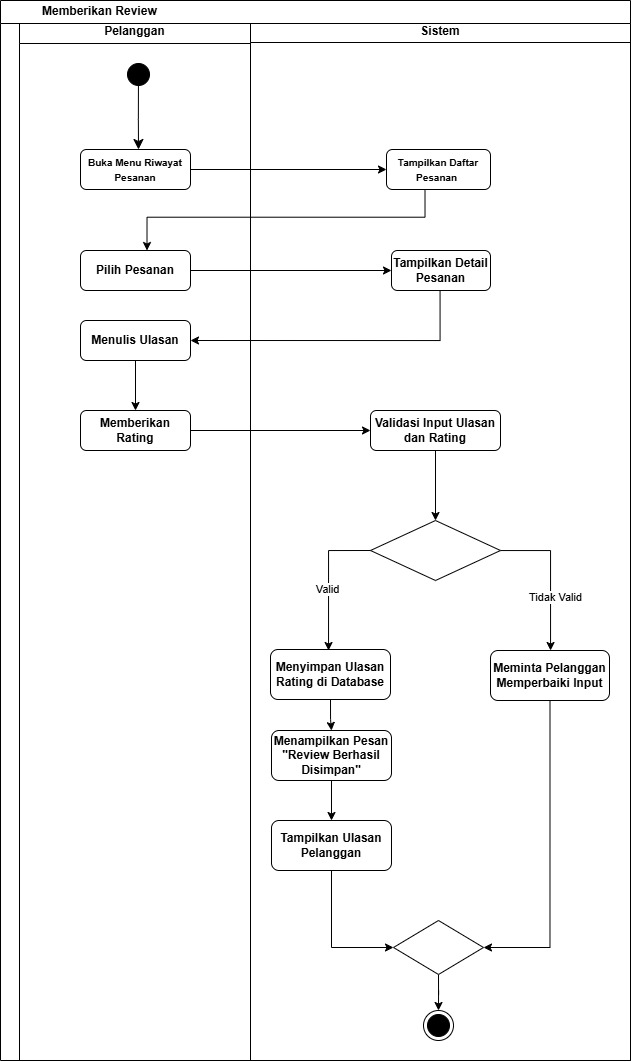
Gambar 5. Activity Diagram Pembayaran

****

****

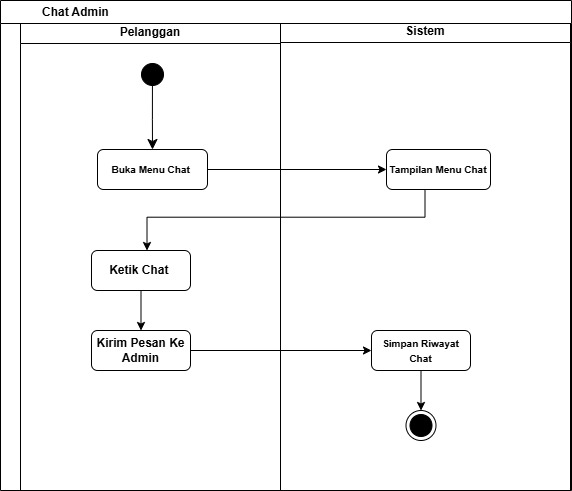
****

* 1. Activity Diagram Memberikan Review

****

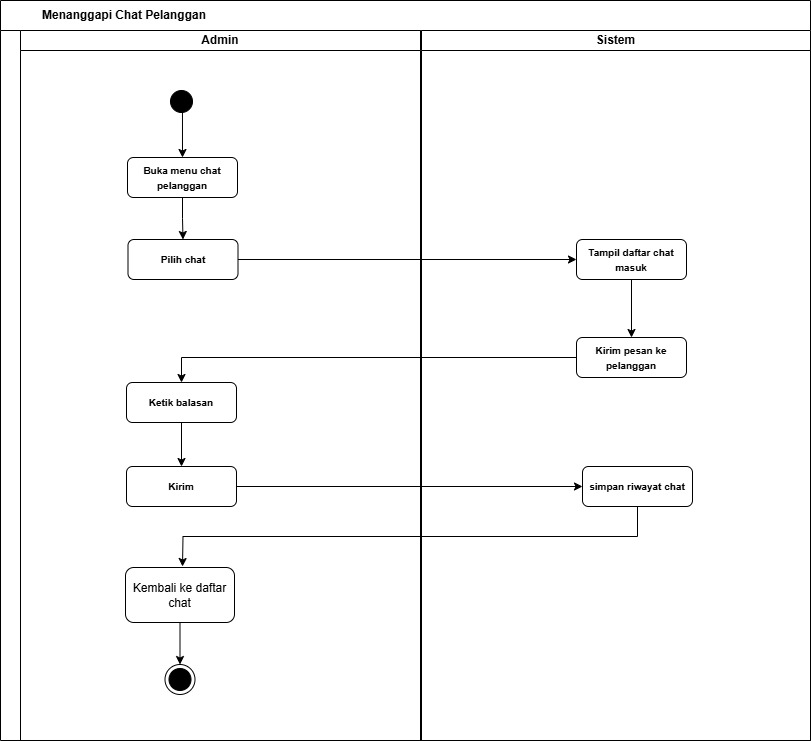
Gambar 6. Activity Diagram Memberikan Review

* 1. Activity Diagram Chat Admin



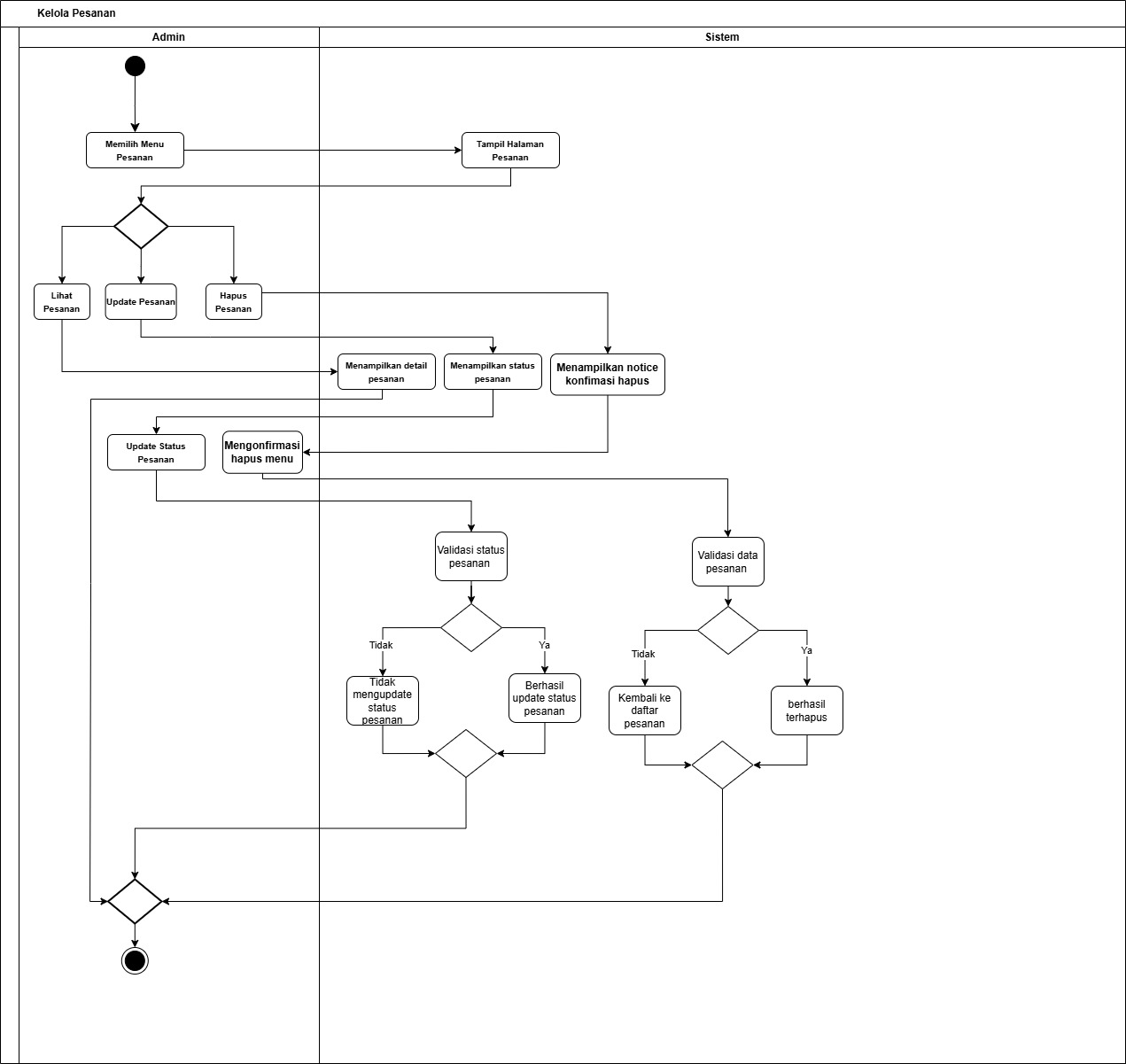
Gambar 7. Activity Diagram Chat Admin

* 1. Activity Diagram Menanggapi Chat Pelanggan



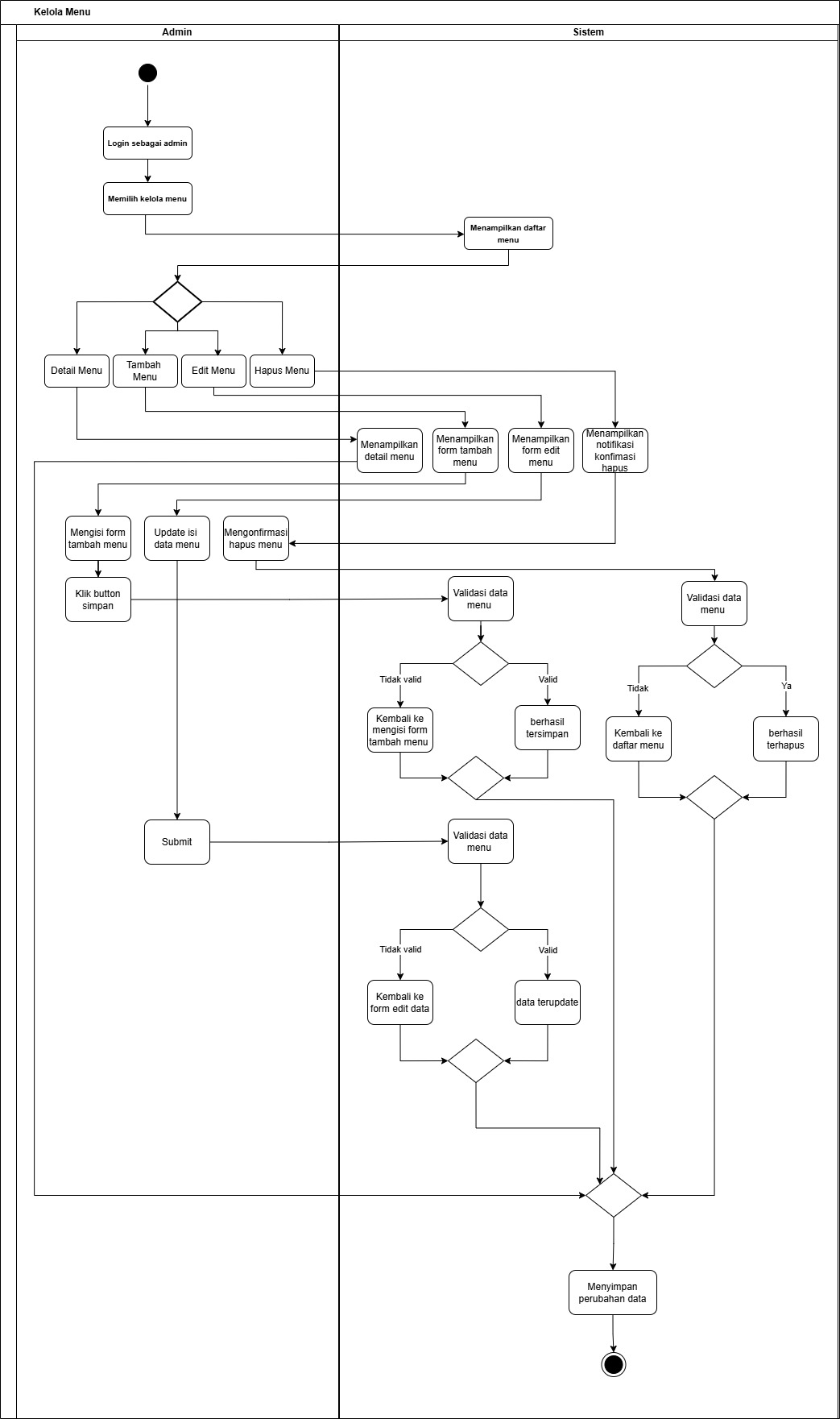
Gambar 8. Activity Diagram Menanggapi Chat Pelanggan

* 1. Activity Diagram Kelola Pesanan

****

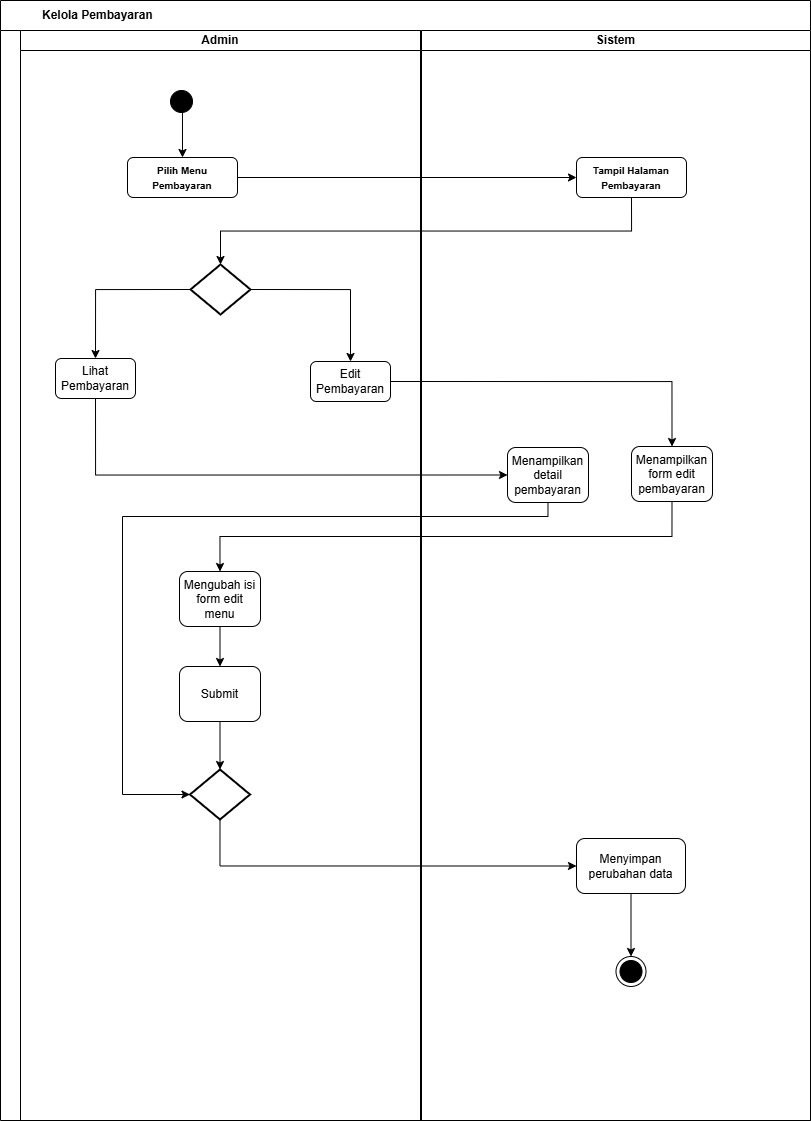
Gambar 9. Activity Diagram Kelola Pesanan

* 1. Activity Diagram Kelola Menu

****

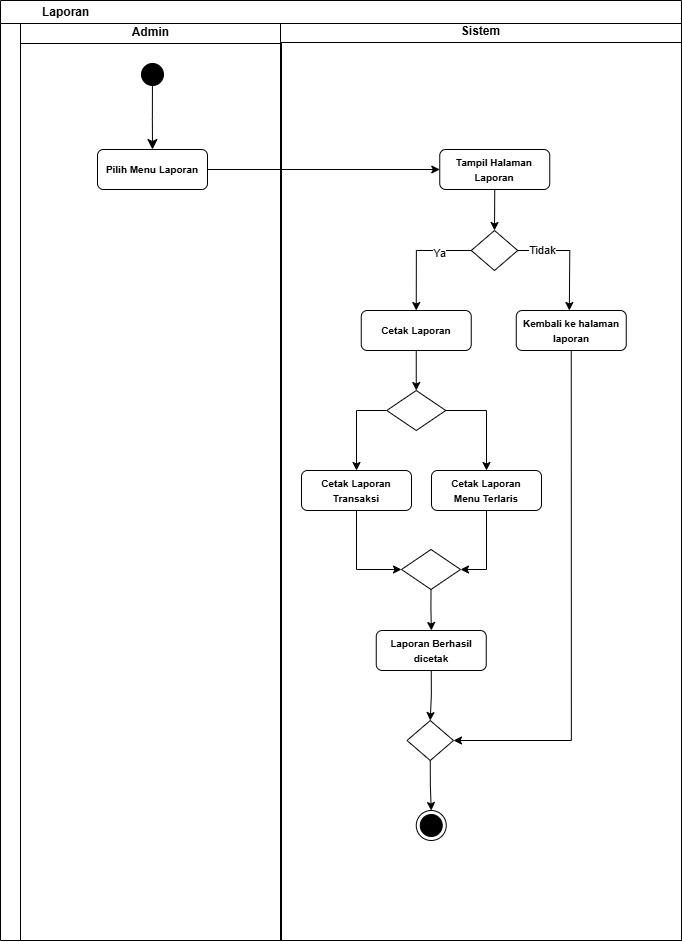
Gambar 10. Activity Diagram Kelola Menu

* 1. Activity Diagram Kelola Pembayaran

****

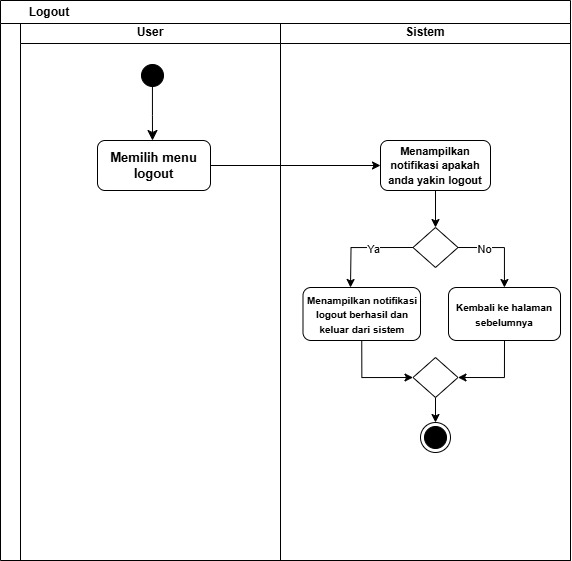
Gambar 11. Activity Diagram Kelola Pembayaran

* 1. Activity Diagram Kelola Laporan

****

Gambar 12. Activity Diagram Kelola Laporan

* 1. Diagram Activity Logout



Gambar 13. Activity Diagram Logout