**RANCANG BANGUN *E-COMMERCE* BERBASIS WEBSITE UNTUK MEMPERLUAS PASAR PRODUK MENGGUNAKAN FRAMEWORK CODEIGNITER**

**(STUDI KASUS DI TOKO 3 PUTRI GORDEN)**

**SKRIPSI**

**Karya Tulis sebagai Syarat untuk Memperoleh**

**Gelar Sarjana Komputer dari Fakultas Teknologi Informasi**

**Universitas Bale Bandung**

Disusun oleh:

**TIARA AWALUL LESTARI**

**NPM.302200014**



**PROGAM STRATA 1**

**PROGRAM STUDI SISTEM INFORMASI**

**FAKULTAS TEKNOLOGI INFORMASI**

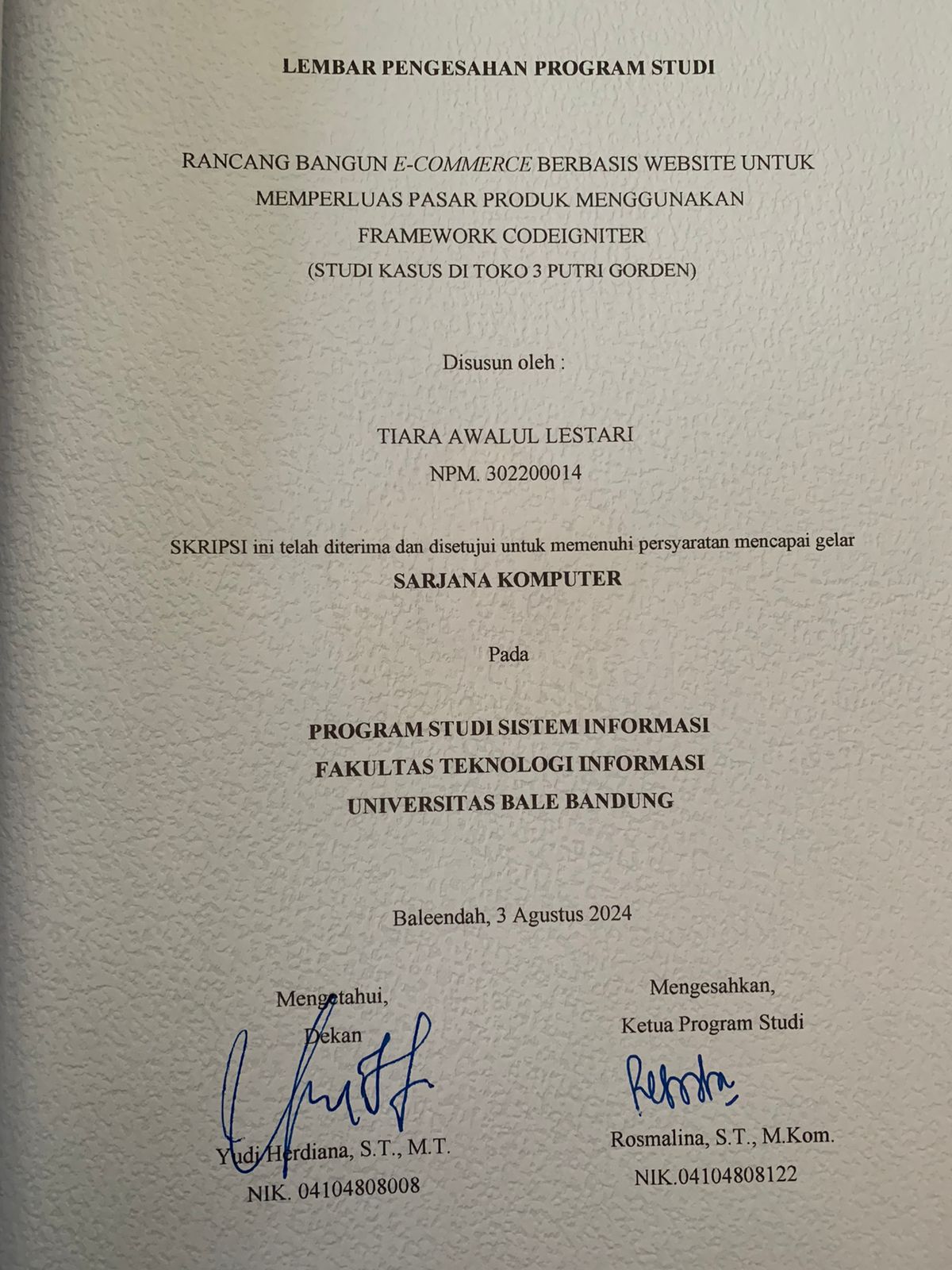
**UNIVERSITAS BALE BANDUNG**

**BANDUNG**

**2024**

# 

****



# 

# ABSTRAK

*E-Commerce* adalah salah satu bagian yang paling terpenting dari internet, Saat ini dengan semakin pesatnya perkembangan teknologi dan internet di indonesia, telah memiliki dampak yang besar terhadap perubahan bisnis. Yaitu mulai dari cara beriklan, cara jual beli, dan sebagainya. Dengan *e-commerce* telah banyak merubah dalam proses jual-beli. Jika dalam suatu jual-beli penjual dan pembeli bertemu, namun jika dengan *e-commerce* mereka tidak perlu bertemu dalam kegiatan promosinya. Toko 3 Putri Gorden merupan toko yang bergerak dibidang gorden dan aksesoris rumah lainnya, Beralamattkan di Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan.

Toko ini mengalami kendala dalam menjangkau konsumen yang lebih luas, Sistem penjualan masih mengandalkan media interaksi dengan konsumen secara langsung dan konsumen yang akan membeli harus datang ke toko secara langsung. Berdasarkan permasalahan diatas maka dibutuhkan sistem yang bisa membantu Toko 3 Putri Gorden dalam mengoptimalkan kegiatan penjualan agar lebih luas dan memudahkan konsumen dalam melakukan pembelian. Metode yang digunakan untuk merancang sistem ini adalah *waterfall* menggunakan tahapan Analisa, Desain, Pengkodean, Pengujian, Iplementasi. Perancangannya menggunakan *Unified Modeling Language* (UML) dengan jenis perancangan *usecase diagram, activity diagram,* dan *class diagram*. Pada tahap Desain menggunakan Aplikasi Figma. Pada tahap pengkodean memakai Framework Codeigniter. Hasil penelitian ini adalah Sistem Informasi *E-Commerce* Berbasis Website Untuk Memperluas Pasar Produk Menggunakan Framework Codigniter.

Dengan adanya website ini diharapkan dapat mengoptimalkan kegiatan penjualan toko sebab pada sistem ini bisa menyajikan informasi tentang produk apa saja yang ada di Toko 3 Putri Gorden. Serta sistem ini juga dapa melakukan proses pembelian sehingga memudahkan konsumen agar tidak perlu datang langsung ke toko.

**Kata Kunci** : Codigniter, E-Commerce, Penjualan, Pembelian.

# ABSTRACT

*E-Commerce is one of the most crucial parts of the internet. With the rapid development of technology and the internet in Indonesia, it has had a significant impact on business changes, including advertising methods, buying and selling processes, and more. E-commerce has greatly altered the buying and selling process. Unlike traditional commerce where sellers and buyers meet, e-commerce eliminates the need for physical interaction during promotional activities. Toko 3 Putri Gorden is a store specializing in curtains and other home accessories, located at Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan.*

*However, the store faces challenges in reaching a wider audience. Its sales system relies on direct interaction with customers, who must visit the store in person to make purchases. To address these challenges, a system is needed to help Toko 3 Putri Gorden optimize its sales activities and reach a broader audience, making it easier for customers to make purchases. The methodology used to design this system is the waterfall model, with stages including Analysis, Design, Coding, Testing, and Implementation. The design utilizes the Unified Modeling Language (UML), incorporating use case diagrams, activity diagrams, and class diagrams. The design stage employs the Figma application. For coding, the Codeigniter Framework is utilized. The outcome of this research is an E-Commerce Information System based on a website to expand product market using the Codeigniter Framework.*

*With this website, it is hoped that the store's sales activities can be optimized. The system provides information about the products available at Toko 3 Putri Gorden and facilitates the purchasing process, eliminating the need for customers to visit the store physically.*

***Keywords****: Codeigniter, E-Commerce, Sales, Purchase.*

# KATA PENGANTAR

Segala puji bagi Allah SWT yang telah memberikan berkat dan Rahmat- Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik dan tepat pada waktunya. Skripsi ini merupakan salah satu syarat untuk menyelesaikan program studi Sistem Informasi jenjang Strata-1 Universitas Bale Bandung. Dalam skripsi ini penulis membahas mengenai “Rancang Bangun E-commerce Berbasis Website Untuk Memperluas Pasar Produk Menggunakan Framework Codiegniter (Studi Kasus Di Toko 3 Putri Gorden)”.

Dalam penyusunan laporan skripsi ini, penulis dibantu oleh berbagai pihak, berkat bantuan dan bimbingannya penulis dapat mengumpulkan data, menyusun, dan pada akhirnya dapat menyelasaikan laporan skripsi ini. Maka dari itu, penulis ingin mengucapkan terima kasih yang sebesar besarnya kepada:

1. Allah SWT yang telah memberikan kemudahan serta kelancaran sehingga penulis dapat menjalani proses demi proses pelaksanakaan Tugas Akhir ini.
2. Kepada cinta pertama dan pintu syurgaku, Ayah Ujang Yayat Hendayana dan Mamah Rosi Handayani, Terima kasih atas do’a dan dukungan yang tak pernah putus, Memberikan cinta kasih sayang dan pengorbanan yang mengiringi setiap langkah untuk menyelesaikan pendidikan ini, Terima kasih telah berjuang untuk penulis, membesarkan dan mendidik sampai mendapatkan gelar sarjana, Semoga Allah SWT senantiasa menjaga kalian sampai melihat penulis berhasil dengan keputusan sendiri. Hiduplah lebih lama.
3. Bapak Yudi Herdiana, S.T., M.T. selaku Dekan Fakultas Teknologi Informasi.
4. Ibu Rosmalina, S.T., M.Kom. selaku Ketua Prodi Sistem Informasi Fakultas Teknologi Informasi dan dosen penguji 1
5. Bapak Denny Rusdianto, S.T., M.Kom. selaku dosen pembimbing utama.
6. Bapak Sukiman, S.Tr.Kom., S.Pd., M.Kom. selaku dosen pembmbing pendamping.
7. Bapak Sutiyono, S.T., M.Kom. selaku dosen penguji 2.
8. Seluruh dosen dan staff pengajar program studi Sistem Informasi Fakultas Teknologi Informasi Universitas Bale Bandung.
9. Bapak Ujang Yayat selaku pemilik Toko 3 Putri Gorden.
10. Adik perempuanku Anggita Rasya Naysila Putria dan Bilqis Siti Dzakira, Terima kasih sudah ikut serta dalam proses penulis menempuh pendidikan selama ini, Terima kasih atas semangat, do’a dan cinta yang selalu diberikan kepada penulis, Tumbuhlah jadi versi yang paling hebat.
11. Teruntuk sahabat seperjuangan, Kelas SI 2020 (Satu7an), Terima kasih sudah menemani penulis selama masa perkuliahan, Terima kasih sudah menjadi teman, Sahabat dan saudara yang selalu menguatkan penulis, Membersamai dalam perjuangan dan tidak pernah mengeluh ketika direpotkan, Kulambangkan kalian dengan angka 0, yang tanpa ujung kurindukan.
12. Kepada para *bestie* yaitu Fazriani Maulani, Tuhfah Taj Nabilah, Lutfia Kasifatul Milah, Eygin Reygina, dan Aulia Hanifa Ardita Terima kasih atas segala dukungan, Motivasi, Pengalaman, Waktu dan ilmu yang dijalani bersama selama perkuliahan, Ucapan syukur kepada Allah SWT karena sudah mempertemukan sahabat seperti kalian, *See you on top guys!*
13. Kepada Devia Fitri Nopiani yang selalu ada disaat penulis sedang berjuang dalam hal apapun, Terima kasih telah menjadi tempat terbaik untuk mengutarakan keluh & dan kesah, menemani perjuangan, memberikan do’a dan dukungaan penuh dalam proses penyelesaian pendidikan ini.
14. Kepada seluruh keluarga besar H.Usman dan Alm Sobandi yang selalu mendo’akan penulis.
15. Last but not least, untuk diri saya sendiri Tiara Awalul Lestari. I wanna thank me for doing all this hard work. I wanna thank me for never quit, Terima kasih sudah bertahan, keluar dari zona nyaman dan berjuang sampai saat ini atas harapan dan impian yang harus diwujudkan. Selamat bergelar S.Kom raaa.
16. Dan untuk semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah turut membantu penulis dalam menyelesaikan laporan ini.

Dalam penulisan skripsi ini, penulis telah berusaha semaksimal mungkin untuk menghasilkan yang terbaik, penulis juga menyadari bahwa masih banyak kekurangan dalam skripsi ini dan segala kritik dan saran yang membangun akan penyusun terima dengan baik. Akhir kata semoga proposal ini bisa diterima dan bermanfaat bagi kita semua.

Bandung, 3 Agustus 2024

TIARA AWALUL LESTARI

NPM. 302200014

# DAFTAR ISI

[**ABSTRAK** i](#_Toc174903154)

[**ABSTRACT** ii](#_Toc174903155)

[**KATA PENGANTAR** iii](#_Toc174903156)

[**DAFTAR ISI** vi](#_Toc174903157)

[**DAFTAR GAMBAR** viii](#_Toc174903158)

[**DAFTAR TABEL** x](#_Toc174903159)

[**BAB I PENDAHULUAN** 1](#_Toc174903160)

[I.1 Latar Belakang 1](#_Toc174903161)

[I.2 Rumusan Masalah 3](#_Toc174903162)

[I.3 Batasan Masalah 3](#_Toc174903163)

[I.4 Tujuan Penelitian 3](#_Toc174903164)

[I.5 Metodologi Penelitian 4](#_Toc174903165)

[1.5.1 Metode Pengumpulan Data 4](#_Toc174903166)

[1.5.2 Metode Pengembaangan Sistem 4](#_Toc174903167)

[1.5.3 Sistematika Penelitian 5](#_Toc174903168)

[**BAB II TINJAUAN PUSTAKA** 7](#_Toc174903169)

[2.1 Landasan Teori 7](#_Toc174903170)

[2.2 Dasar Teori 10](#_Toc174903171)

[**BAB III METODOLOGI PENELITIAN** 29](#_Toc174903172)

[3.1 Kerangka Pikir 29](#_Toc174903174)

[3.2.1 Pengumpulan Data 30](#_Toc174903175)

[3.2.2 Analisis 30](#_Toc174903176)

[3.2.3 Desain 31](#_Toc174903177)

[3.2.4 Pengkodean 34](#_Toc174903178)

[3.2.5 Pengujian 34](#_Toc174903179)

[**BAB IV ANALISIS DAN PERANCANGAN** 35](#_Toc174903180)

[4.1 Analisis 35](#_Toc174903182)

[4.1.1 Analisis Tata Kelola Perusahaan 35](#_Toc174903183)

[4.1.2 Analisis Sistem 37](#_Toc174903184)

[4.1.3 Analisis Masalah 37](#_Toc174903185)

[4.1.4 Analisis kebutuhan sistem 40](#_Toc174903186)

[4.1.5 Analisis penunjang keputusan 41](#_Toc174903187)

[4.1.6 Analisis data penunjang keputusan 42](#_Toc174903188)

[4.1.7 Analisis kebutuhan / Alat 44](#_Toc174903189)

[4.1.8 Analisis Biaya 45](#_Toc174903190)

[4.2 Perancangan 46](#_Toc174903191)

[4.2.1 Pemodelan UML (Unified Modelling Language) 46](#_Toc174903192)

[**BAB V IMPLEMENTASI DAN PENGUJIAN 99**](#_Toc174903193)

[5.1 Hasil Tampilan 99](#_Toc174903195)

[5.2 Hasil Pengujian 106](#_Toc174903196)

[**BAB VI KESIMPULAN DAN SARAN 110**](#_Toc174903197)

[6.1 Kesimpulan 110](#_Toc174903199)

[6.2 Saran 110](#_Toc174903200)

[**DAFTAR PUSTAKA 112**](#_Toc174903201)

[**LAMPIRAN 115**](#_Toc174903202)

[**BERITA ACARA HASIL WAWANCARA 115**](#_Toc174903203)

[**RIWAYAT HIDUP 119**](#_Toc174903204)

# DAFTAR GAMBAR

[Gambar 2. 1 Logo PHP 22](#_Toc173505313)

[Gambar 2. 2 Metode Waterfall 24](#_Toc173505314)

[Gambar 2. 3 Logo Figma 25](#_Toc173505315)

[Gambar 2. 4 Logo Visual Studio Code 26](#_Toc173505316)

[Gambar 2. 5 Logo HTML 27](#_Toc173505317)

[Gambar 3. 1 Kerangka Pikir 29](#_Toc173504481)

[Gambar 4. 1 Struktur Organisasi 35](#_Toc173504221)

[Gambar 4. 2 Use Case Diagram 46](#_Toc173504222)

[Gambar 4. 3 Activity Diagram Login Admin 49](#_Toc173504223)

[Gambar 4. 4 Activity Diagram Logout 50](#_Toc173504224)

[Gambar 4. 5 Activity Diagram Pesanan 51](#_Toc173504225)

[Gambar 4. 6 Activity Diagram Produk 52](#_Toc173504226)

[Gambar 4. 7 Activity Diagram User 54](#_Toc173504227)

[Gambar 4. 8 Activity Diagram Login Customer 56](#_Toc173504228)

[Gambar 4. 9 Activity Diagram Logout Customer 57](#_Toc173504229)

[Gambar 4. 10 Activity Diagram Checkout Barang 58](#_Toc173504230)

[Gambar 4. 11 Activity Diagram Riwayat Pesanan 60](#_Toc173504231)

[Gambar 4. 12 Activity Diagram Profil 62](#_Toc173504232)

[Gambar 4. 13 Class Diagram 63](#_Toc173504233)

[Gambar 4. 14 User Interface Login Customer 68](#_Toc173504234)

[Gambar 4. 15 User Interface Register Customer 69](#_Toc173504235)

[Gambar 4. 16 User Interface Halaman Utama 70](#_Toc173504236)

[Gambar 4. 17 User Interface Produk 71](#_Toc173504237)

[Gambar 4. 18 User Interface Pesanan 71](#_Toc173504238)

[Gambar 4. 19 User Interface Pembayaran 72](#_Toc173504239)

[Gambar 4. 20 User Interface Profil 73](#_Toc173504240)

[Gambar 4. 21 User Interface Contact 74](#_Toc173504241)

[Gambar 4. 22 User Interface Login Admin 75](#_Toc173504242)

[Gambar 4. 23 User Interface Register Admin 75](#_Toc173504243)

[Gambar 4. 24 User Interface Home Admin 76](#_Toc173504244)

[Gambar 4. 25 User Interface User 76](#_Toc173504245)

[Gambar 4. 26 User Interface Produk Admin 77](#_Toc173504246)

[Gambar 4. 27 User Interface Pesanan Admin 77](#_Toc173504247)

[Gambar 5. 1 Tampilan Login Admin 99](#_Toc173500262)

[Gambar 5. 2 Tampilan Register Admin 99](#_Toc173500263)

[Gambar 5. 3 Tampilan Dashboard Admin 100](#_Toc173500264)

[Gambar 5. 4 Tampilan User Admin 100](#_Toc173500265)

[Gambar 5. 5 Tampilan Produk Admin 101](#_Toc173500266)

[Gambar 5. 6 Tampilan Pesanan Admin 101](#_Toc173500267)

[Gambar 5. 7 Tampilan Login Customer 102](#_Toc173500268)

[Gambar 5. 8 Tampilan Registrasi Customer 102](#_Toc173500269)

[Gambar 5. 9 Tampilan Beranda Customer 103](#_Toc173500270)

[Gambar 5. 10 Tampilan Beranda Customer 103](#_Toc173500271)

[Gambar 5. 11 Tampilan Produk 104](#_Toc173500272)

[Gambar 5. 12 Tampilan Pembayaran 104](#_Toc173500273)

[Gambar 5. 13 Tampilan Pesanan / Riwayat Pesanan 105](#_Toc173500274)

[Gambar 5. 14 Tampilan Profil 105](#_Toc173500275)

[Gambar 5. 15 Tampilan Kontak 106](#_Toc173500276)

[Gambar 5. 16 Tampilan Kontak 106](#_Toc173500277)

# DAFTAR TABEL

[Tabel 2. 1 Acuan Penelitian 7](#_Toc173507885)

[Tabel 2. 2 Simbol Use Case Diagram 17](#_Toc173507886)

[Tabel 2. 3 Simbol Activity Diagram 19](#_Toc173507887)

[Tabel 2. 4 Simbol Class Diagram 20](#_Toc173507888)

[Tabel 3. 1 Perangkat Keras 31](#_Toc173507512)

[Tabel 3. 2 Perangkat Lunak 31](#_Toc173507513)

[Tabel 4. 1 Analisis Metode PIECES 37](#_Toc173507258)

[Tabel 4. 2 Analisis Biaya 45](#_Toc173507259)

[Tabel 4. 3 Deskripsi Aktor 47](#_Toc173507260)

[Tabel 4. 4 Deskripsi Use Case 48](#_Toc173507261)

[Tabel 4. 5 Activation 64](#_Toc173507262)

[Tabel 4. 6 Groups 64](#_Toc173507263)

[Tabel 4. 7 Groups User 64](#_Toc173507264)

[Tabel 4. 8 Groups 64](#_Toc173507265)

[Tabel 4. 9 Login 65](#_Toc173507266)

[Tabel 4. 10 Permissions 65](#_Toc173507267)

[Tabel 4. 11 Reset 65](#_Toc173507268)

[Tabel 4. 12 Tokens 65](#_Toc173507269)

[Tabel 4. 13 User 66](#_Toc173507270)

[Tabel 4. 14 Kategori 66](#_Toc173507271)

[Tabel 4. 15 Order 66](#_Toc173507272)

[Tabel 4. 16 Produk 67](#_Toc173507273)

[Tabel 4. 17 User 67](#_Toc173507274)

[Tabel 5. 1 Rencana Pengujian 107](#_Toc173505814)

# BAB I PENDAHULUAN

## Latar Belakang

Transformasi digital penting bagi semua perusahaan industri dan sektor permerintahan yang sangat bergantung pada sistem, TI, strategi, dan sumber daya manusia. Transformasi digital secara ekstensif dianggap sebagai pendorong perubahan dalam semua konteks, terutama dalam konteks bisnis, dan mempengaruhi semua aspek kehidupan manusia berdasarkan pemanfaatan teknologi dan digitalisasi. (Tulungen, Saerang, and Maramis 2022)

Penjualan secara online mulai merambah dunia seiring dengan berkembangnya internet dan masuknya era globalisasi dan perdagangan bebas. Tingkat pendapatan masyarakat yang makin meningkat juga membawa dampak signifikan dengan tingkat konsumerisme masyarakat. Makin majunya teknologi komputer pada suatu wilayah sangat berpengaruh terhadap perilaku dan budaya para penduduknya, Inilah Salah satu hal yang positif seiring dengan kemajuan IT adalah cara masyarakat melakukan jual beli yang semula menggunakan cara tradisional, mulai bergeser menjadi jual beli online. Termasuk dalam produknya. Sebelum internet berkembang, perusahaan umumnya memasarkan dengan memasang iklan baik di radio, Tv, Koran, pamflet dan spanduk serta media lain. Tetapi dengan majunya perdagangan dengan menggunakan internet atau sering kita sebut E-Commerce, maka mulai berubahlah paradigma perusahaan dalam memasarkan dan menjual produknya. Mereka mulai memanfaatkan website untuk memasarkan dan menjual produknya. (Fathur Rohman 2019)

Toko 3 Putri Gorden merupakan toko yang bergerak dibidang Gorden dan aksesoris rumah lain nya, Beralamatkan di Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan. Selama ini sistem penjualan masih mengandalkan media interaksi dengan konsumen secara langsung dan konsumen yang akan membeli harus datang ke toko secara langsung. Hal ini tentunya menyulitkan perusahaan mempromosikan dan memperluas produk dagangannya karena hanya terbatas pada lokasi tertentu.

Berdasarkan observasi yang penulis lakukan, dapat ditemukan bahwa sistem informasi berbasis website sangat dibutuhkan, sehingga dapat diolah perusahaan untuk memasarkan barang sebagaimana kebutuhan produk yang diperlukan oleh konsumen. Integrasi dan koordinasi merupakan elemen penting untuk melakukan upaya sinkronisasi proses penjualan perusahaan sehingga bila semuanya dilakukan dengan baik maka efisiensi biaya akan diperoleh dan menguntungkan bagi perusahaan.

Beberapa penelitian telah dilakukan, oleh penelitian terdahulu seperti penelitian yang dilakukan oleh Sri Peni Mugi Handayani, Bambang Eka Purnama., (2013) di Toko Distro JavaTrend yang membangun sistem informasi *e-commerce.* Selain itu, penelitian yang dilakukan oleh Iis Meilinda.,(2022) yang membangun sistem informasi *e-commerce* menggunakan metode *waterfall.* Lalu, penelitian yang dilakukan oleh Sri Haryanti, Tri Irianto.,(2011) yang membangun sistem informasi *e-commerce* berbasis website.

Oleh karena itu, pembuatan sistem informasi *e-commerce* sangat bermanfaat untuk memperluas pasar produk, terutama bagi bisnis yang sebelumnya masih menjalankan penjualan secara manual. Dengan adanya platform e-commerce, Toko 3 Putri Gorden dapat menjangkau lebih banyak pelanggan potensial secara geografis tanpa batas, meningkatkan efisiensi operasional melalui otomatisasi proses bisnis, menyediakan pengalaman belanja yang lebih nyaman bagi pelanggan, serta memanfaatkan data dan analitik untuk strategi pemasaran yang lebih efektif. Semua ini pada akhirnya dapat berkontribusi pada peningkatan penjualan dan pertumbuhan bisnis secara keseluruhan.

Berdasarkan latar belakang tersebut maka penulis mempertimbangkan untuk membuat sistem informasi E-commerce berbasis website dengan judul "RANCANG BANGUN *E-COMMERCE* BERBASIS WEBSITE UNTUK MEMPERLUAS PASAR PRODUK MENGGUNAKAN FRAMEWORK CODEIGNITER (Studi Kasus Di Toko 3 Putri Gorden)" dimana dengan adanya sistem informasi *e-commerce* ini nantinya akan memudahkan toko untuk memperluas penjualan dan mempermudah konsumen membeli tanpa harus datang langsung ke toko.

## Rumusan Masalah

Berdasarkan uraian pada latar belakang, maka dapat dirumuskan bahwa masalah penelitian yaitu :

1. Bagaimana merancang sistem informasi *e-commerce*?
2. Bagaimana membangun sistem informasi e-commerce untuk membantu Toko 3 Putri Gorden agar bisa memperluas produknya
3. Bagaimana pembangunan *e-commerce* untuk menjangkau dan menarik pelanggan potensial?
4. Bagaimana membangun *e-commerce* untuk membantu konsumen dalam mempermudah melakukan pembelian?

## Batasan Masalah

Adapun batasan masalah dalam pelaksanaan penelitian dan pembuatan sistem informasi yakni:

1. Penelitian dilakukan di Toko 3 Putri Gorden di Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan.
2. Website ini juga dapat melakukan proses pembelian produk
3. Metode yang digunakan dalam penelitian ini adalah *Waterfall.* Pada metode ini, penelitian hanya dilakukan sampai pengujian saja, tidak sampai implementasi dikarenakan hanya pihak owner yang berhak untuk mengimplementasikan di perusahaan nya atau tidak.

## Tujuan Penelitian

Berdasarkan rumusan masalah sebelumnya, maka tujuan penelitian yang ingin dicapai adalah:

Terbangunnya sistem informasi *e-commerce* untuk menjangkau dan menarik pelanggan potensial

Terbangunnya sistem informasi e-commerce untuk membantu Toko 3 Putri Gorden agar bisa memperluas produknya

Terbangunnya sistem informasi *e-commerce* agar membantun konsumen dalam mempermudah melakukan pembelian

## Metodologi Penelitian

Metodologi penelitian merupakan sebuah cara untuk mengetahui hasil dari sebuah permasalahan, dimana permasalahannya disebut juga dengan permasalahan penelitian.

### 1.5.1 Metode Pengumpulan Data

Dalam pelaksanaan pengerjaan skripsi, penulis menggunakan metode kualitatif untuk mendapatkan data yang akurat yang diperlukan dalam pembuatan sistem informasi ini, penyusunan proposal, dan penyusunan laporan, yaitu sebagai berikut:

* Observasi, yaitu dilakukan dengan cara mengamati langsung sistem informasi yang berjalan pada Toko 3 Putri Gorden untuk mendapatkan informasi yang dibutuhkan dalam membuat sistem informasi *e-commerce.*
* *Interview*, yaitu melakukan Tanya jawab bersama bapak ujang yayat selaku pemilik Toko 3 putri gorden mengenai sistem yang berjalan di toko tersebut.
* Studi Pustaka, yaitu dilakukan dengan mengumpulkan data dari beberapa buku, jurnal, dan sumber lainnya yang bersangkutan dengan masalah yang sedang dibahas.

### 1.5.2 Metode Pengembaangan Sistem

Dalam penelitian ini penulis menggunakan metode *Waterfall* untuk metode pegembangaan sistem. Metode ini memungkinkan untuk departementalisasi dan *control*. *Waterfall* adalah metode pengembangan sistem klasik, dan paling banyak digunakan para pengembang sistem, adapun tahapan-tahapan dalam metode *waterfall* yaitu: Analisa, Desain, Pengkodean dan Pengujian.

### 1.5.3 Sistematika Penelitian

Dalam menyusun laporan skripsi ini diatur dan disusun dalam enam bab, yang masing-masing terdiri dari beberapa sub bab. Adapun urutannya adalah sebagai berikut:

**BAB I : PENDAHULUAN**

Bagian ini berisi mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian dan sistematika penulisan.

**BAB II : TINJAUAN PUSTAKA**

Bagian ini berisi mengenai landasan teori yang bersumber dari jurnal-jurnal yang sesuai dengan objek penelitian dan dasar teori yang membangun penyusunan skripsi ini.

**BAB III : METODOLOGI PENELITIAN**

Dalam bab ini penulis mengemukakan metode penelitian yang dilakuan dalam analisis, waktu dan tempat penelitian, alat dan bahan, juga perancangan sistem.

**BAB IV : ANALISIS DAN PERANCANGAN**

Bagian ini membahas mengenai tata kelola perusahaan, analisis sistem, analisis kebutuhan sistem, analisis sistem penunjang keputusan, analisis data penunjang keputusan, analisis kebutuhan atau alat, analisis biaya dan perancangan mengenai sistem informasi yang akan dibuat berupa *use case* diagram*, activity* diagram*, sequence* diagram*, class* diagramdan *user interface.*

**BAB V : IMPLEMENTASI DAN PENGUJIAN**

Bagian ini membahas mengenai implementasi dan pengujian sebuah aplikasi yang telah dibuat oleh penyusun serta pengambilan dokumentasi atau foto mengenai hasil implementasi yang telah di buat.

**BAB VI : KESIMPULAN DAN SARAN**

Bagian ini membahas mengenai kesimpulan dan saran dari hasil penelitian yang telah dilakukan.

# BAB II TINJAUAN PUSTAKA

## Landasan Teori

Pada penelitian ini, penyusun mempelajari penelitian terdahulu sebagai acuan dalam penelitian Perancangan dan Pembangunan *E-Commerce* Berbasis Website Untuk Memperluas Pasar Produk Di Toko 3 Putri Gorden sebagai berikut:

Tabel 2. 1 Acuan Penelitian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Judul Penelitian** | **Masalah** | **Metode** | **Solusi** |
| 1 | Judul: Pembuatan Website E-commerce Pada Distro Java Trend | Penjualan dan pembelian masih manual dengan cara datang ke toko atau melalui media situs pertemanan facebook serta BBM. | *Waterfall* | Membangun website yang dapat membantu konsumen dalam melakukan proses pembelian, pembayaran dan pengiriman barang. |
| Penulis : Sri Peni Mugi Handayani, Bambang Eka Purnama. |
| Tahun : 2013 |
| Jurnal : Seminar Riset Unggulan Nasional Informati dan Bisnis |
| 2 | Judul : Rancang Bangun Sistem E-Commerce Menggunakan Metode Marketing Mix Untuk Memperluas Pangsa Pasar Pada Toko Dewi Lampung Selatan | Calon Pembeli haruslah datang langsung ke toko hanya untuk menanyakan nama barang, harga barang dan spesifikasi barang yang dijual di Toko Dewi Pasuruan. Toko Dewi pun hanya bisa memberikan informasi tentang promo/diskon hanya pada saat calon pembeli datang ke toko dengan memberikan brosur promo. | *Waterfall* | Membangun E-commerce berbasis web untuk mendukung peningkatan konsep 4P pada Toko Ibu Dewi |
| Penulis : Iis Meilinda |
| Tahun : 2022 |
| Jurnal: Jurnal Informatika dan Rekayasa Perangkat Lunak (JATIKA) |
| 3 | Judul : Rancang Bangun Sistem Informasi E-Commerce Untuk Usaha Fashion Studi Kasus Omah Mode Kudus | Penjualan fashion di toko ini masih secara offline atau masih konvensional sehingga untuk meningkatkan dan memperluas penjualan perlu dibangunnya sebuah Ecommerce | *Waterfall* | Membangun sistem informasi e-commerce agar penjualan fashion ditoko ini meningkat serta lebih luas. |
| Penulis : Sri Haryanti, Tri Irianto |
| Tahun : 2011 |
| Jurnal : Journal Speed – Sentra Penelitian Engineering dan Edukasi |

* + 1. Sri Peni Mugi Hnadayani, Bambang Eka Purnama, dalam penelitian ini menyatakan permasalahan dalam Penjualan dan pembelian masih manual dengan cara datang ke toko atau melalui media situs pertemanan facebook serta BBM. Metode yang digunakan dalam pengembangan sistem yaitu dengan model *waterfall.* Penelitian ini menghasilkan perancangan dan pembuatan Sistem informasi website penjulan secara online di Distro JavaTrend, Membangun website yang dapat membantu konsumen dalam melakukan proses pembelian , pembayaran dan pengiriman barang. Sistem Informasi ini diharapkan dapata membantu toko Java Trend sebagai media promosi produk dan dalam transaksi penjualan produk dan para calon pembeli dapat melihat update produk yang tersedia setiap saat melalui website Java Trend. Semoga dengan adanya website Java Trend para pembeli dapat melihat dan memesan produk tanpa harus datang langsung ke toko.
    2. Iis Meilinda, dalam penelitian ini menyatakan permasalahan calon Pembeli haruslah datang langsung ke toko hanya untuk menanyakan nama barang, harga barang dan spesifikasi barang yang dijual di Toko Dewi Pasuruan. Toko Dewi pun hanya bisa memberikan informasi tentang promo/diskon hanya pada saat calon pembeli datang ke toko dengan memberikan brosur promo. Metode yang digunakan dalam pengembangan sistem yaitu dengan model *waterfall.* Penelitian ini menghasilkan Sistem Informasi E-commerce yang dapat diterapkan sebagai bauran pemasaran atau marketing mix di Toko Dewi. Dimana E-commerce tersebut mampu untuk mempercepat pelayanan jarak jauh.
    3. Sri Haryanti, Tri Irianto dalam penelitian ini menyatakan permasalahan penjualan fashion di toko ini masih secara offline atau masih konvensional sehingga untuk meningkatkan dan memperluas penjualan perlu dibangunnya sebuah Ecommerce. Metode yang digunakan dalam pengembangan sistem yaitu dengan model *Waterfall*. Penelitian ini menghasilkan sistem informasi menghasilkan sebuah toko online factory outlet dimana didalamnya menyediakan informasi produk dan menangani pembelian secara online, menghasilkan simulator toko online factory outlet, dan menghasilkan reporting untuk setiap transaksi penjualan. di harapkan memudahkan dalam hal menawarkan dan menjual produk fashion “Omah Mode” secara online kepada konsumen. Selain itu website E-Commerce juga dapat memperluas serta meningkatkan penjualan karena dengan online ruang lingkup perdagangan juga akan semakin luas, Bagi konsumen transaksi jual beli secara online lebih cepat, mudah dan aman karena dengan pembayaran lewat internet banking kita tidak perlu pergi kemana-mana untuk mendapatkan barang yang akan kita beli lewat internet.

## Dasar Teori

Pada pembuatan sistem informasi ini, penyusun mempunyai bahan dan alat untuk pembuatan sistem informasi, maka penyusun mempelajari teori-teori yang ada sebagai acuan dalam implementasi sistem informasi yang akan penulis buat yaitu sebagai berikut

* + - 1. **Basis data**

Teori tentang basis data diperoleh di mata kuliah SIF 311 yaitu sistem basis data. Basis data memiliki peran penting dalam perancangan website karena memberikan pemahaman yang mendalam tentang pengelolaan data, struktur database, dan teknologi terkait.

* + - 1. **Interaksi manusia dan komputer**

Teori tentang konsep manusia dan komputer diperoleh di mata kuliah TIF308 yaitu interaksi manusia dan komputer. Dengan memadukan pemahaman tentang pengguna dan desain antarmuka, mata kuliah Interaksi manusia dan komputer mempersiapkan mahasiswa untuk merancang website yang tidak hanya estetis tetapi juga memperhatikan pengalaman pengguna yang optimal.

* + - 1. **Pemrograman Berbasis Web**

Teori tentang Pemrograman web diperoleh di mata kuliah SIF311 yaitu pemrograman berbasis web. Mata kuliah pemrograman berbasis website memiliki peran kunci dalam mempersiapkan mahasiswa untuk merancang, mengembangkan, dan memelihara website, Dengan menguasai keterampilan pemrograman berbasis website, mahasiswa siap untuk terlibat dalam pengembangan website dari awal hingga akhir, serta untuk membuat website yang interaktif, responsif, dan dapat diandalkan.

1. **Sistem**

Menurut Erni Widarti, S.Kom., M.Kom dalam buku nya yang berjudul “BUKU AJAR PENGANTAR SISTEM INFORMASI” Istilah “sistem” merujuk pada suatu entitas yang terdiri dari elemen atau komponen yang saling berhubungan dan berinteraksi secara kontinu atau berkesinambungan dengan lingkungan sekitarnya. Hal ini bertujuan untuk mencapai suatu tujuan tertentu yang terikat dan terpadu yang berlangsung terus-menerus. Setiap elemen atau komponen yang terdapat dalam sebuah sistem memiliki fungsi dan cara kerja masing-masing tetapi tetap beroperasi secara terpadu dalam satu kesatuan fungsi atau kerja. Tugas dan fungsi antara setiap elemen atau komponen tidak akan saling bertentangan atau berlawanan, karena semuanya saling bergantung dan saling membutuhkan satu sama lain untuk mencapai tujuan tertentu. (Agustina 2024)

“Sistem adalah jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan-kegiatan pokok perusahaan” (Mulyadi)

Berdasarkan definisi diatas, maka dapat disimpulkan sistem adalah suatu kumpulan dari bagian-bagian ataupun jaringan yang saling berhubungan satu dengan yang lainnya dan bekerja sama untuk mencapai satu tujuan.

1. **Informasi**

Informasi dapat diartikan sebagai data yang telah diolah menjadi bentuk yang memiliki arti dan fungsi sehingga dapat dipahami oleh manusia, sedangkan data merupakan fakta-fakta mentah yang belum diolah agar dapat dimengerti oleh manusia Setiap orang pasti tak lepas dari informasi dalam kehidupan sehari hari mereka, baik dalam hal menyampaikan informasi telah menjadi bagian penting dari kehidupan manusia. Ada berbagai teori informasi yang diungkapkan oleh para ahli yang berusaha menjelaskan makna “informasi” dalam kalimat yang bisa dipahami oleh orang banyak dalam pengertian yang hampir seragam. (Laudon dan Laudon 2018)

Informasi melalui media online seperti jurnal, ebook, surat kabar online, media social (facebook, instagram, twitter) dan sebagainya yang dapat memberikan data dan informasi bermanfaat guna menjawab persoalan bagi penggunanya sedangkan menurut sudut pandang dunia perpustakaan, informasi adalah suatu rekaman fenomena yang diamati, atau bisa juga berupa Keputusan yang dibuat seseorang. (Gusnita et al., 2021)

Informasi merupakan hasil dari pengolahan data menjadi bentuk yang lebih berguna bagi yang menerimanya dan dapat digunakan sebagai alat bantu untuk pengambilan suatu keputusan.(Hapzi 2019)

Berdasarkan definisi diatas, maka dapat disimpulkan definisi informasi tersebut maka ini artinya mempunyai peranan penting dalam pengembangan kebudayaan, ilmu pengetahuan sepanjang masa dan informasi dapat ditemukan dalam berbagai media baik cetak maupun media non cetak.

1. **Sistem Informasi**

Sistem Informasi adalah sistem yang digunakan untuk menyimpan sekaligus menganalisis data-data yang sudah diinput serta menghasilkan suatu format laporan yang merepresentasikan data-data yang telah diinput. Sistem informasi merupakan gabungan antara bahasa pemrograman yang didukung dengan sistem basis data. (Kurniawati 2017)

Pengertian sistem informasi adalah sistem informasi yang diciptakanoleh para pengolahan data anak, data kegiatan, data sponsor, data pemeriksaan kesehatan, pemeriksaan sosio emosi, data perpindahan anak, data pejabat, data tutor, bantuan pelayanandan data inventarisasi barang analisis dan manajer guna melaksanakan tugas khusus tertentuyang sangat esensial bagi berfungsinya organisasi. (Nopriandi 2018)

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak tertentu dengan laporan-laporan yang diperlukan. (Amri, Kurnia Cahya Lestari dan Arni Muarifah 2014)

Sistem Informasi adalah kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data, memproses dan menyimpan serta mendistribusikan informasi. (SIHOTANG 2019)

Berdasarkan beberapa pengertian di atas maka dapat disimpulkan bahwa sistem informasi adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi tersebut untuk mendukung operasi, manajemen, dan pengambilan keputusan dalam suatu organisasi. Sistem informasi mencakup berbagai komponen, termasuk perangkat keras (hardware), perangkat lunak (software), data, prosedur, dan manusia.

1. ***E-commerce***

Menurut Albert Lodewyk Sentosa Siahaan dalam bukunya yang berjudul “*E-COMMERCE*” Internet saat ini sangat mempengaruhi perilaku bisnis dan menjadi penyebab perubahan sesungguhnya dalam industri. Memanfaatkan internet dalam menjalankan bisnis biasa dikenal dengan sebutan *E-Commerce (Electronic Commerce)*. Dikutip dari Laudon dan Traver, yang mengatakan bahwa *E-Commerce* merupakan proses dimana terjadinya jual dan beli produk ataupun jasa, yang dilakukan secara elektronik oleh konsumen dan dari perusahaan ke perusahaan dengan menggunakan perantara komputer untuk transaksi bisnis. *E-Commerce* bermanfaat mengurangi beberapa biaya jika dibandingkan dengan toko *offline.* Waktu yang dihabiskan untuk bertransaksi juga bisa lebih singkat. Bahkan *E-Commerce* juga dapat meningkatkan hubungan penjual dan pembeli. (Richter, Carlos, and Beber n.d.)

*E-Commerce (Electronic Commerce)* adalah proses transaksi jual beli yang menggunakan alat elektronik seperti telepon dan internet. *E-commerce* lebih dari sekedar membeli dan menjual produk secara online. *E-commerce* meliputi seluruh proses dari pengembangan, pemasaran, penjualan, pengiriman, pelayanan, dan pembayaran para pelanggan, dengan dukungan dari jaringan para mitra bisnis di seluruh dunia. Sistem *e-commerce* sangat bergantung pada sumber daya internet dan banyak teknologi informasi lainnya untuk mendukung setiap proses ini. (Yadewani and Wijaya 2017)

*E-commerce* adalah singkatan dari *Electronic Commerce* yang artinya sistem pemasaran secara atau dengan media elektronik. *E-commerce* ini mencakup distribusi, penjualan, pembelian, marketing dan service dari sebuah produk yang dilakukan dalam sebuah sistem elektronika seperti internet atau bentuk jaringan komputer yang lain. *E-commerce* bukan sebuah jasa atau sebuah barang, tetapi merupakan perpaduan antara jasa dan barang*.E-commerce* dan kegiatan yang terkait melalui internet dapat menjadi penggerak untuk memperbaiki ekonomi domestik melalui liberalisasi jasa domestik dan mempercepat integrasi dengan kegiatan produksi global. (Rehatalanit 2016)

Secara garis besar *e-commerce* adalah istilah yang digunakan untuk menggambarkan penjualan barang dan jasa melalui Internet. Dalam pengertian yang paling umum, hanya menciptakan situs Web yang mengiklankan dan mempromosikan produk anda dapat dianggap “*e-commerce*.” dalam beberapa tahun terakhir, bagaimanapun *e-commerce* telah menjadi jauh lebih canggih. Bisnis *e-commerce* sekarang menawarkan toko online dimana pelanggan dapat mengakses ribuan produk, pemesanan, pilih metode pengiriman yang diinginkan dan pembayaran dengan menggunakan atm, mobile-banking, atau kartu kredit mereka. (Rehatalanit 2016)

*E-commerce* merupakan perluasan makna dari elektronik b*usiness (e-business)* yang mencakup lebih luas mengenai jual beli barang dan jasa. Ruang lingkup *E-business* mencakup tentang pelayanan pelanggan, kolaborasi antara beberapa relasi bisnis dan penggunaan transaksi berbasis elektronik dalam sebuah perusahaan bisnis. (Yusuf et al. 2022)

Berdasarkan beberapa pengertian di atas maka dapat disimpulkan bahwa *e-commerce* dapat membantu penjual dalam mempromosikan barangnya secara online Karena dengan adanya *e-commerce* memberikan banyak kemudahan bagi calon konsumen untuk membeli barang yang diinginkannya dan memanfaatkan semua fasilitas yang ada. Dan setiap orang bisa memilih barang memberikan banyak kemudahan transaksi, produk yang beraneka ragam ataupun belanja di toko yang menjual kebutuhan konsumen. Kapan saja calon pelanggan bisa memanfaatkan *e-commerce* tersebut. *E-commerce* juga memberikan cara – cara baru dalam melaksanakan bisnis, sehingga siapa saja bisa memanfaatkan media tersebut untuk menjadi wirausaha yang mampu menciptakan dan memperluas bisnis secara lebih cepat dan lebih mudah dibanding dulu. *E-commerce* lebih dari sekedar membeli dan menjual produk secara online.

1. **Penjualan**

Penjualan adalah sebuah sistem yang melibatkan sumber daya di dalam suatu organisasi, prosedur, data, maupun sarana pendukung untuk mengoperasikan sistem penjualan, sehingga menghasilkan informasi yang berguna bagi pihak manajemen di dalam pengambilan suatu keputusan yang diinginkan. (Anggraini et al. 2020)

Penjualan adalah suatu transfer hak atas benda-benda. Dari penjelasan tersebut dalam memindahkan atau mentransfer barang dan jasa diperlukan orang-orang yang bekerja di bidang penjualan seperti pelaksanaan dagang, agen, wakil pelayanan dan wakil pemasaran.(Rachmawati 2011)

Berdasarkan pengertian di atas dapat diambil kesimpulan bahwa penjualan adalah kegiatan bidang pemasaran dalam menjual barang produksinya kepada konsumen yang dilaksanakan perusahaan dalam menyampaikan barang produksinya agar dapat dinikmati oleh konsumen. Selain itu dapat mengajak orang lain agar bersedia membeli barang atau jasa yang ditawarkan oleh produsen atau konsumen.

1. **Pembelian**

Penerimaan siswa baru merupakan gerbang awal yang harus dilalui peserta didik dan sekolah dalam penyaringan objek-objek pendidikan. Peristiwa penting bagi suatu sekolah, karena peristiwa ini merupakan titik awal yang menentukan kelancaran tugas suatu sekolah. Kesalahan dalam penerimaan siswa baru dapat menentukan sukses tidaknya usaha pendidikan di sekolah yang bersangkutan(Sarwindah 2018)

Pembelian adalah transaksi belanja untuk barang masuk atau pengeluaran uang yang kita lakukan untuk mendapatkan produk yang akan dijual, transaksi ini terjadi pada supplier yang produknya dibeli (Nurhayati, Josi, and Hutagalung 2018)

Berdasarkan pengertian diatas Penulis menyimpulkan bahwa pembelian adalah suatu kegiatan yang mengeluarkan sejumlah uang untuk mendapatkan barang ataupun produk yang diinginkan sesuai dengan kebutuhan.

1. **Perancangan Sistem**

***Unified Modeling Language* (UML)**

UML digunakan untuk menggambarkan perancangan awal dari sistem yang akan dibangun. UML memiliki banyak jenis pemodelan, tetapi hanya beberapa yang digunakan saja yang akan dibahas.

*Unified Modeling Language* (UML) adalah sebuah bahasa pemodelan perangkat lunak yang telah distandarisasi sebagai media penulisan cetak biru (*blueprints*) perangkat lunak (*Pressman*). UML bisa saja digunakan untuk *visualisasi, spesifikasi, konstruksi* dan *dokumentasi* beberapa bagian-bagian dari *system* yang ada dalam perangkat lunak. Dalam kata lain, seperti halnya seorang arsitek dalam membuat dokumen cetak biru yang digunakan oleh perusahaan konstruksi untuk membangun sebuah bangunan, arsitek perangkat lunak membuat diagram-diagram UML untuk membantu programmer/ developer membangun perangkat lunak. Untuk selanjutnya, semakin kita mengetahui beberapa kosakata yang. (Abdillah 2021)

***Use Case Diagram***

*Use Case Diagram* adalah sesuatu atau proses merepresentasikan hal-hal yang dapat dilakukan oleh aktor dalam menyelesaikan sebuah pekerjaan (Heriyanto 2018)

Menurut Shalahuddin dalam jurnal (Heriyanto 2018) mengungkapkan : “*Diagram use case* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.” Berikut adalah simbol-simbol yang ada pada diagram *use case*:

Tabel 2. 2 Simbol Use Case Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | *Actor* | Menspesifikasikan himpunan peran yang pengguna mainkan ketik berinteraksi dengan *use case*. |
| 2 |  | *Use Case* | Deskripsi dari urutan aksi- aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu Actor |
| 3 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 4 |  | *Extend* | Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan |
| 5 |  | *Generalization* | Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada diatasnya objek induk. |
| 6 |  | *Include* | Menspesifikasikan bahwa *use case* sumber secara eksplisit. |

***Activity Diagram***

“Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas dapat dilakukan oleh sistem”. (Heriyanto 2018)

Tabel 2. 3 Simbol Activity Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Ketrangan** |
| 1 |  | Status Awal | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja |
| 2 |  | Aktivitas | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu Actor |
| 3 |  | Percabangan | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| 4 |  | Penggabungan | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan  menjadi satu |
| 5 |  | Status Akhir | Status akhir yang dilakukan system, sebuah diagram aktivitas memiliki sebuah status akhir |
| 6 |  | *Swimlane* | *Swimlane* memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

***Class Diagram***

“*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas”. (Heriyanto 2018)

Tabel 2. 4 Simbol Class Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | *Actor* | Mewakili peran orang, sistem yang lain atau alat Ketika berkomunikasi dengan *use case.* |
| 2 |  | *Use Case* | Abstraksi dan interaksi antara system actor dengan *use case* |
| 3 |  | *Association* | Abstraksi dari penghubung antara aktor dengan *use case.* |
| 4 |  | *Generalisasi* | Menunjukan spesialisasi actor untuk dapat berpartisipasi dengan *use case.* |
| 5 | *<<include>>* | *Include* | Menunjukan bahwa suatu *use case* seluruhnya merupakan fungsionalitas dari *use case* lainnya. |
| 6 | *<<extend>>* | *Extend* | Menunjukan bahwa suatu *use case* merupakan tambahan fungsionalitas dari *use case* lainnya jika suatu kondisi terpenuhi |

1. **MYSQL**

MySQL adalah sebuah DBMS (*Database Management System*) menggunakan perintah SQL (*Structured Query Language*) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL dibagi menjadi dua lisensi, pertama adalah *Free Software*dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah *Shareware* dimana perangkat lunak berpemilik memiliki batasan dalam penggunaannya.

SQL (*Structured Query Language*) adalah sebuah bahasa scripting yang dipergunakan untuk mengolah database. Database besar seperti Mysql, Postgre SQL dan SQL Server sudah menggunakan SQL untuk mengolah database nya. SQL yang dipergunakan software database tersebut adalah sama kecuali sedikit perbedaan dibeberapa tempat. (Ramadhan and Mukhaiyar 2020)

1. **PHP**

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman berbasis *server–side* yang dapat melakukan parsing *script* php menjadi *script* web sehingga dari sisi client menghasilkan suatu tampilan yang menarik. PHP adalah PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman web berbasis *server* (*server slide*) yang mampu memparsing kode PHP dari kode dengan ekstensi PHP sehingga menghasilkan tampilan website yang dinamis di sisi *client* (Yusuf Amir, Aisyiyah, and Devi 2022)*.*



Gambar 2. 1 Logo PHP

1. **Framework Codeigniter**

CodeIgniter adalah kerangka kerja untuk pengembangan aplikasi berbasis web dengan menggunakan bahasa PHP. Dengan menggunakan framework, kita tidak perlu membuat program dari awal, tetapi kita sudah diberikan library fungsi-fungsi yang sudah diorganisasi untuk dapat membuat suatu program dengan cepat. CodeIgniter juga menggunakan mekanisme perancangan dengan konsep Model, View, Controller (MVC) yang merupakan konsep untuk mengatur aplikasi menjadi 3 bagian.Codeigniter memiliki tujuan untuk memberikan alat bantu yang dibutuhkan seperti helpers and libraries untuk mengimplementasi tugas yang biasa dilakukan. Dengan demikian, pengembangan proyek menjadi lebih mudah dan cepatdanpengembang tidak perlu menulis lagi dari awal. (Krisna, Muhammad, and Ambadar 2022)

Codeigniter adalah sebuah framework yang mengadopsi aturan MVC (Model, View, Controller) berbasis PHP. Codeigniter mempercepat pengerjaan aplikasi tanpa perlu menulis semua kode dari awal. Codeigniter sangat ringan digunakan, dan juga memiliki dokumentasi yang lengkap dengan sertakan contoh penggunaannya. (Christian and Alfath 2021)

Codeigniter Adalah kerangka kerja PHP: Hypertext Preprocessor yang dapat membantu mempercepat developer dalam pengembangan aplikasi website (web) berbasis PHP daripada menulis semua kode program dari awal Beberapa keuntungan menggunakan Codeigniter adalah gratis, ditulis menggunakan PHP 4, ukuran kecil, menggunakan Konsep Model View Controller (MVC), Uniform Resource Locator (URL) yang sederhana. (Kurniadi, Riki, and Nurkamilah 2022)

1. **Metode SDLC Waterfall**

Metode air terjun atau yang sering disebut metode waterfall seing dinamakan siklus hidup klasik (classic life cycle), nama model ini sebenarnya adalah “Linear Sequential Model” dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (planning), permodelan (modelling), konstruksi (contruction), serta penyerahan sistem ke para pengguna (deployment), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan.(Abdul Wahid 2020)

(SDLC) adalah proses mengembangkan atau mengubah suatu sistem dengan menggunakan model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem sebelumnya. Sementara itu, model waterfall sering juga disebut sekuensial linier (Sequential linier) atau alur hidup klasik. Model waterfall menyediakan pendekatan alur hidup sistem secara sekuensial atau terurut dimulai dari:

1. Analisa Kebutuhan

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user, pada tahapan ini perlu didokumentasikan.

1. Desain

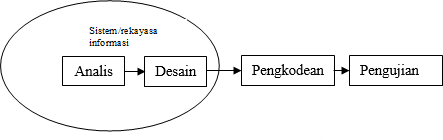
Proses multilangkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.

1. Pengkodean

Desain harus ditransletkan ke dalam perangkat lunak, hasil dari tahapan ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahapan sebelum nya.

1. Pengujian

Pengujian fokus pada perangkat lunak, secara segi logik dan fungsional untuk memastikan bahwa semua bagian sudah diuji, hal ini dilakukan untuk meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.



Gambar 2. 2 Metode Waterfall

1. **XAMPP**

XAMPP adalah perangkat lunak ( *free software*) bebas, yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri beberapa program antara lain : *Apache HTTP Server*, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis. Untuk mendapatkanya XAMPP anda dapat mendownload langsung dari web resminya. Dan berikut beberapa definisi program lainnya yang terdapat dalam XAMPP.

XAMPP ini merupakan project non-profit yang dikembangkan oleh Apache Friends yang didirikan Kai 'Oswald' Seidler dan Kay Vogelgesang pada tahun 2002, project mereka ini bertujuan mempromosikan penggunaan Apache web server**.**

1. **Figma**

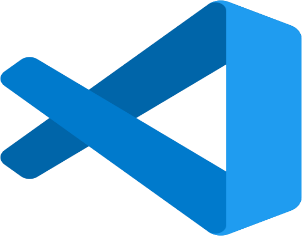
Figma adalah salah satu design tool yang biasanya digunakan untuk membuat tampilan aplikasi mobile, desktop, website dan lain-lain. Figma bisa digunakan di sistem operasi windows, linux ataupun mac dengan terhubung ke internet. Umumnya Figma banyak digunakan oleh seseorang yang bekerja dibidang UI/UX, web design dan bidang lainnya yang sejenis. Selain mempunyai kelengkapan fitur layaknya Adobe XD, Figma memiliki keunggulan yaitu untuk pekerjaan yang sama dapat dikerjakan oleh lebih dari satu orang secara bersama-sama walaupun ditempat yang berbeda. Hal tersebut bisa dikatakan kerja kelompok dan karena kemampuan aplikasi figma tersebut lah yang membuat aplikasi ini menjadi pilihan banyak UI/UX designer untuk membuat prototype website atau aplikasi dengan waktu yang cepat dan efektif. (Muhyidin, Sulhan, and Sevtiana 2020)



Gambar 2. 3 Logo Figma

1. **Visual Studio Code**

Visual Studio Code merupakan software kode editor lintas platform yang ringan namun kuat yang dapat digunakan oleh siapa saja untuk membangun aplikasi Web. Muncul dengan builtin dukungan untuk JavaScript, naskah dan Node.js dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C# , Python, dan PHP. Visual Studio Code menggunakan open source NET untuk memberikan dukungan ASP. Antarmuka yang mudah untuk bekerja karena didasarkan pada gaya explorer umum, dengan panel di sebelah kiri yang 11 menunjukkan semua file dan folder dan di sebelah kanan yang menunjukkan isi dari file yang telah dibuka. Visual studio code memiliki fungsi yang baik, dengan intellisense dan autocomplete bekerja dengan baik untuk JSON, CSS, HTML, Node.js dan lainnya



Gambar 2. 4 Logo Visual Studio Code

1. **Website**

Website disebut juga site, situs, situs web atau portal. Merupakan kumpulan halaman web yang berhubungan antara satu dengan lainnya, halaman pertama sebuah website adalah home page, sedangkan halaman demi halamannya secara mandiri disebut web page, dengan kata lain website adalah situs yang dapat diakses dan dilihat oleh para pengguna internet diseluruh dunia. Website adalah situs yang dapat diakses dan dilihat oleh para pengguna Internet. Pengguna Internet semakin hari semakin bertambah banyak, sehingga hal ini adalah potensi pasar yang berkembang terus. (Abas 2021)

Website adalah kumpulan dari halaman-halaman situs, yang terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam World Wide Web ( WWW ) di dalam internet, website juga dapat diartikan sebagai sebuah halaman yang berisi data, baik data text, gambar, suara dan lainya yang dapat diakses secara online. (Josi 2017)

Website merupakan salah satu jenis layanan/fasilitas yang disediakan oleh internet yang paling banyak digunakan disamping layanan-layanan yang lainnya. Kelebihan website dibandingkan dengan fasilitas internet yang lainnya adalah karena website mampu menyajikan informasi berupa teks, gambar, suara ataupun video yang interaktif.(Soejono, Setyanto, and Sofyan 2018)

1. **HTML**

HyperText Markup Language (HTML) HyperText Markup Language (HTML) adalah Bahasa yang digunakan untuk menggambarkan struktur halaman Web. HTML dapat digunakan untuk :

a. Publikasi dokumen online dengan judul, teks, tabel, daftar, foto, dll.

b. Mengambil informasi online melalui tautan hypertext, dengan sekali klik sebuah tombol.

c. Merancang formulir untuk melakukan transaksi dengan layanan jarak jauh, untuk digunakan dalam mencari informasi, melakukan pemesanan, pembelian produk, dll.

d. Menyertakan spread sheet, klip video, klip suara, dan aplikasi lainnya secara langsung di dalam dokumen. Dengan HTML, penulis mendeskripsikan struktur halaman dengan menggunakan markup atau penanda. (Saputra and Astuti 2018)



Gambar 2. 5 Logo HTML

1. **Black Box Testing**

Metode Black Box adalah metode yang ketika melakukan pengujian tidak perlu melihat dari kode pemrograman, tidak seperti White Box Testing yang mengutamakan kode pemrograman yang digunakan beserta dengan kelas dan fungsinya. Black Box Testing sendiri memiliki beberapa teknik dalam melakukan pengujian salah satu tekniknya adalah Equivalence Partitions. Perancangan kasus uji (test case) pada teknik Equivalence Partitions didasarkan pada evaluasi kelas yang menggambarkan kondisi ekuivalen masukan dari kumpulan kadaan yang valid dan tidak valid (Trengginaz et al. 2020)

1. **Pieces**

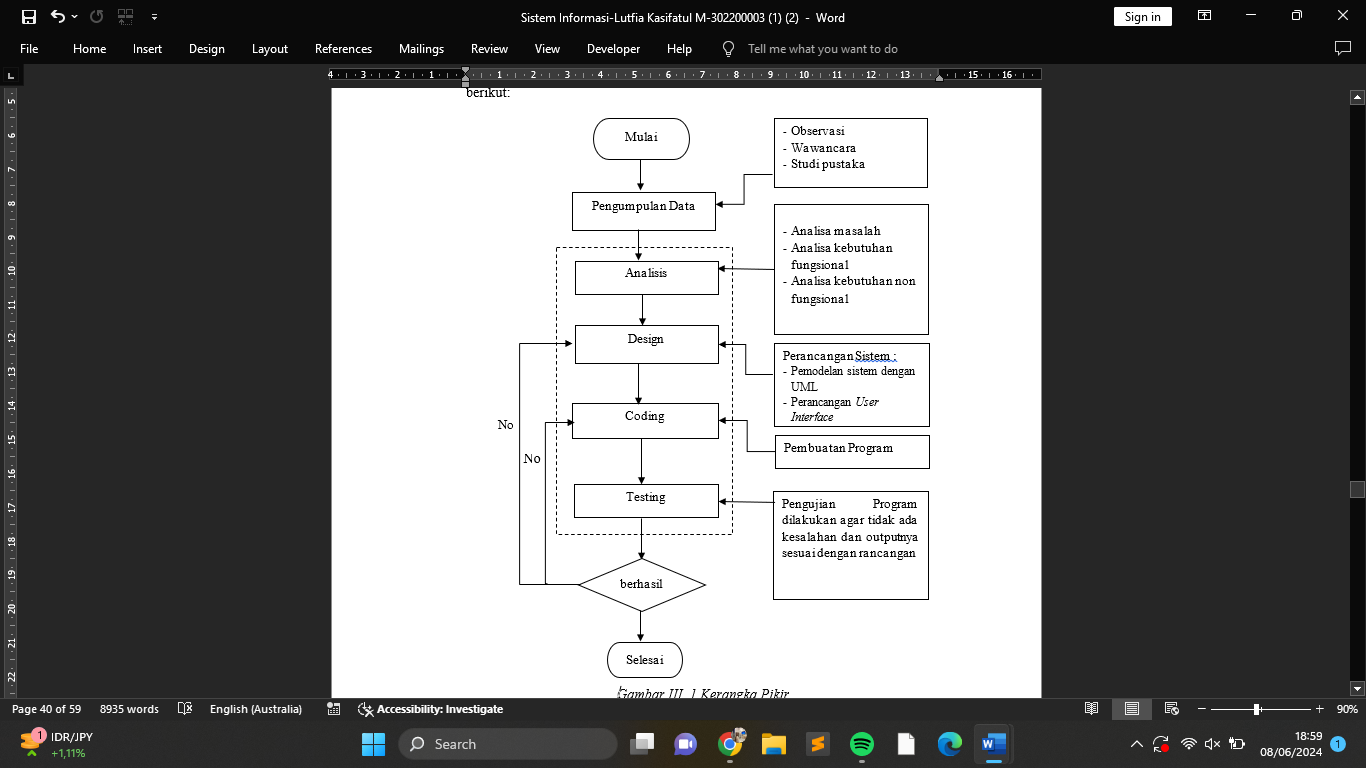
Metode PIECES adalah metode analisis sebagai dasar untuk memperoleh pokok-pokok permasalahan yang lebih spesifik. Dalam menganalisis sebuah sistem, biasanya akan dilakukan terhadap beberapa aspek antara lain adalah kinerja (performance), informasi (information), ekonomi (economics), keamanan aplikasi (control), efisiensi (efficiency) dan pelayanan pelanggan (service). (Flora, Tarigan, and Maksum 2022)

# BAB III

# METODOLOGI PENELITIAN

## Kerangka Pikir

Kerangka Pikir ini merupakan langkah-langkah yang akan dilakukan untuk menyelesaikan masalah dalam penelitian ini, berdasarkan metode *waterfall,* sebagai berikut:



Gambar 3. 1 Kerangka Pikir

1. **Deskripsi**

Berikut adalah tahapan-tahapan yang ada pada penelitian ini:

### Pengumpulan Data

Tahap pengumpulan data adalah salah satu tahap awal yang penting dalam proses pengembangan perangkat lunak. Tahap ini bertujuan untuk mengumpulkan informasi dan persyaratan yang diperlukan untuk memahami kebutuhan pengguna dan tujuan proyek. Informasi ini dapat diperoleh melalui pengumpulan data sebagai berikut:

* Wawancara, yaitu dilakukan dengan proses tanya jawab dengan Bapak Ujang Yayat Hendayana selaku Pemilik Toko 3 Putri Gorden. Pemantauan mengenai pengelolaan data, sistem dan mengumpulkan informasi, mengidentifikasi masalah yang terjadi serta data yang dibutuhkan dalam penelitian ini.
* Observasi, yaitu dilakukan dengan cara mengamati langsung proses yang berjalan di Toko 3 Putri Gorden yang beralamat di Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan., untuk mendapatkan informasi yang dibutuhkan dalam membuat sistem informasi *e-commerce* berbasis website*,* mengidentifikasi masalah yang terjadi dan supaya penulis dapat mengetahui sistem yang sedang berjalan di Toko 3 Putri Gorden.
* Studi pustaka, yaitu dilakukan dengan mengumpulkan data dari membaca beberapa buku, jurnal, dan sumber lainnya yang berhubungan dengan sistem informasi *e-commerce* berbasis web.

### Analisis

Tahap ini penulis melakukan analisis terhadap sistem yang berjalan serta melakukan analisis kebutuhan :

* + - 1. Analisis kebutuhan fungsional

Analisis kebutuhan fungsional merupakan analisis kebutuhan sistem dari segi fungsionalitas pada saat sistem dirancang. Beberapa kebutuhan fungsionalitas yaitu:

* + - 1. Memiliki form login yang diisi username dan password yang dimiliki oleh admin dan user.
      2. Memiliki halaman utama yang terdiri dari menu produk dan menu pemesanan.
      3. Menu pemesanan merupakan menu untuk mendata pemesanan. Pada menu pemesanan terdapat form yang harus diisi oleh konsumen berdasarkan keterangan dari pesanan tersebut.
      4. Menu produk merupakan menu yang menampilkan produk.
      5. Analisis kebutuhan non fungsional

1. Analisis perangkat keras (*Hardware*)

Tabel 3. 1 Perangkat Keras

|  |  |
| --- | --- |
| Processor | Intel® Coleron®Quad core processor N4120 |
| RAM | 4 GB |
| Monitor | 14”inch |
| Printer | Inkjet Printer |

1. Analisis perangkat lunak (*Software*)

Tabel 3. 2 Perangkat Lunak

|  |  |
| --- | --- |
| Sistem Operasi | Windows 10 |
| Database | Mysql |
| Aplikasi Pembuatan | Xampp, VSCode, Codeigniter, Figma |

### Desain

Tahap desain pada perancangan sistem ini bertujuan untuk memberikan gambaran secara umum tentang perancangan sistem informasi *e-commerce* berbasis web. Adapun rancangan yang dilakukan dalam tahapan ini antara lain sebagai berikut:

1. Perancangan Model

Pada perancangan model ini dilakukan untuk mengetahui alur yang akan dibuat untuk merancang aplikasi toko online. Perancangan model ini akan dibuat dengan menggunakan *Unified Modelling Language* (UML). Dan diagram yang akan digunakan yaitu:

* 1. *Use Case* Diagram digunakan untuk keterhubungan *actor* dan *use case* dalam aplikasi yang akan dibuat. *Actor* dan *use case* yang terlibat dalam penelitian ini adalah:
     1. *Actor : admin, customer*
     2. *Use case :*

a) Admin : dapat mengelola data produk dan pesanan

b) *Customer* : melakukan pemesanan dan pembelian

* 1. *Activity* Diagram digunakan untuk memodelkan alur kerja dari sistem dan aktivitas dari actor dalam aplikasi. Adapun diagram alur yang akan dibuat yaitu:

Admin : dapat melakukan login, menambah data produk, menghapus data produk, dan mengedit data produk

Customer : dapat melakukan pemesanan dan pembelian produk

* 1. *Class* Diagram

Dengan alur database yang saling berhubungan pada sistem yang akan dikembangkan yaitu digunakan untuk menggambarkan atribut, operasi, constraint pada sistem yang akan dibuat. Class diagram yang akan dibuat adalah sebagai berikut:

a. Class diagram admin

b. Class diagram user

c. Class diagram produk

d. Class diagram pemesanan

1. Database

Dalam perancangan database, data yang dibutuhkan sebelumnya telah didapatkan pada proses pengumpulan data yang dibutuhkan sebelumnya pada proses pengumpulan data dan sudah di analisis. Perancangan database menggunakan class diagram. beberapa data yang digunakan adalah:

* + Data User
  + Data user

1. Perancangan *User Interface*

Perancangan antarmuka merupakan perancangan terakhir dalam tahap desain pengembangan sistem. Mendesain user interface atau tampilan program secara keseluruhan yang terdiri dari menu dan sub menu. Perancangan user interface dengan menggunakan *Figma*. Adapun perancangan antarmuka yang akan dibuat sebagai berikut:

* 1. Tampilan Halaman *Login*
  2. Tampilan Halaman Admin
     + Tampilan halaman utama
     + Tampilan produk
     + Tampilan Data Pesanan
     + Tampilan detail pesanan
  3. Tampilan halaman user
     + Tampilan produk
     + Tampilan form pemesanan

### Pengkodean

Pada tahapan Peengkodean berisikan pembuatan program dimana penulis mulai mengaplikasikan website yang berada pada tahap desain kedalam bentuk program agar dapat dipahami oleh mesin (komputer) menggunakan bahasa pemrograman PHP dengan framework yang digunakan yaitu Codeigniter, HTML dan CSS. Sementara untuk data basenya penulis menggunakan MYSQL dan untuk kode editornya penulis menggunakan VSCode.

### Pengujian

Program yang telah dibuat akan dilakukan tahap pengujian program secara keseluruhan dengan tujuan untuk mengetahui tingkat keberhasilan website sesuai dengan fungsinya. Jika terdapat kesalahan, maka akan dilakukan perbaikan. Dalam pengujian ini menggunakan metode black box testing yang dilakukan untuk mengetahui kesesuaian input dan output dengan apa yang di harapkan.

# BAB IV

# ANALISIS DAN PERANCANGAN

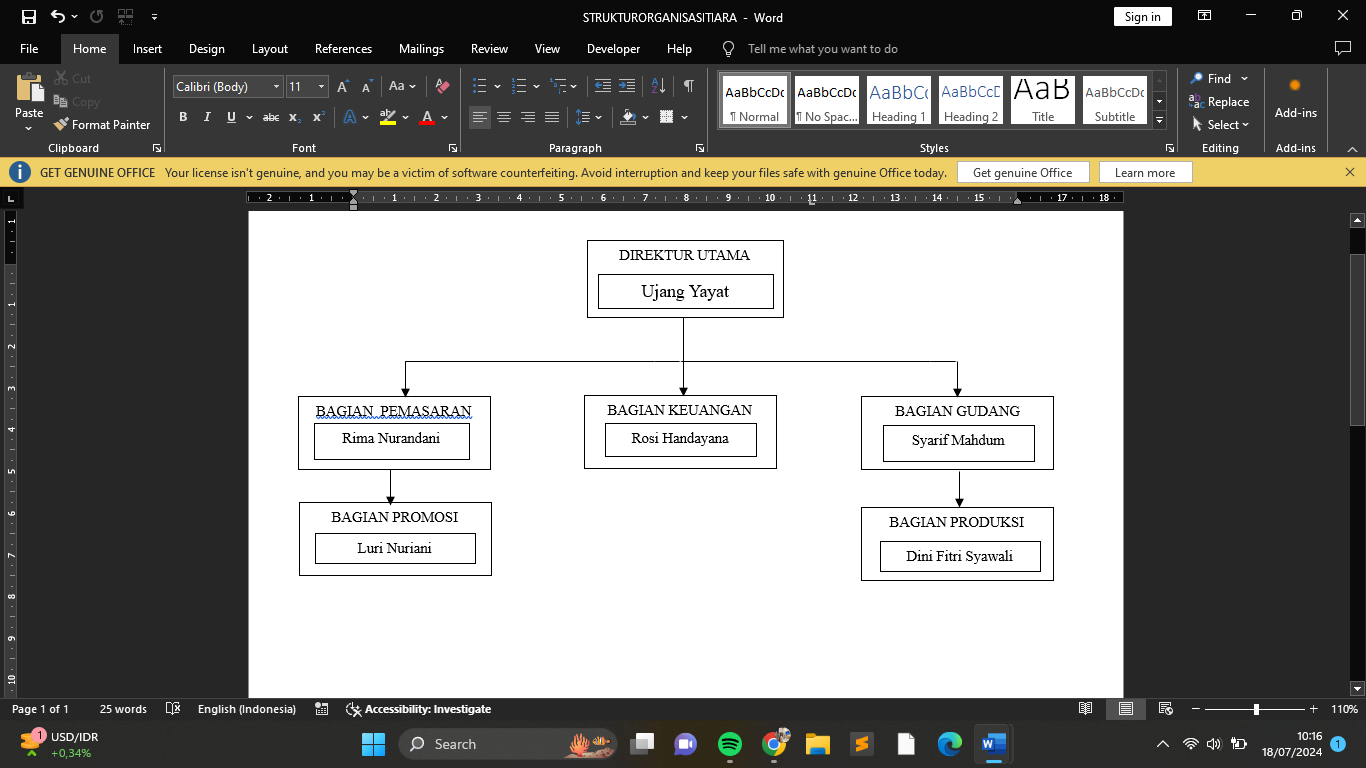
## Analisis

Pada tahap ini, analisis yang dilakukan dalam proses penelitian ini meliputi beberapa tahapan yaitu Analisis masalah, analisis software, analisis pengguna, user interface, fitur-fitur, analisis data serta analisis biaya.

### Analisis Tata Kelola Perusahaan

Toko 3 Putri Gorden merupakan toko yang bergerak dibidang Gorden dan aksesoris rumah lain nya, Beralamatkan di Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan.

Adapun struktur organisasi Toko 3 Putri Gorden adalah sebagai berikut.



Gambar 4. 1 Struktur Organisasi

Tanggung jawab dari pengelola Toko 3 Putri Gorden:

* Direktur Utama
* Memimpin dan mengawasi keseluruhan operasional toko.
* Mengambil keputusan strategis untuk perkembangan dan keberlanjutan bisnis.
* Mengawasi kinerja pegawai dan memastikan bahwa semua departemen bekerja sesuai dengan target.
* Bagian Promosi
* Mengembangkan strategi pemasaran untuk meningkatkan penjualan.
* Mengelola kampanye iklan dan promosi.
* Melakukan analisis pasar untuk memahami tren dan kebutuhan pelanggan.
* Bagian Penjualan
* Melakukan penjualan produk kepada pelanggan.
* Memberikan pelayanan yang baik kepada pelanggan dan menangani keluhan.
* Memastikan stok produk selalu tersedia dan terorganisir dengan baik di toko.
* Bagian Keuangan
* Mengelola keuangan toko, termasuk perencanaan anggaran, pembukuan, dan laporan keuangan.
* Melakukan analisis keuangan untuk mendukung keputusan bisnis.
* Mengawasi pengeluaran dan memastikan efisiensi biaya operasional.
* Bagian Gudang
* Mengelola persediaan barang di gudang.
* Memastikan barang yang masuk dan keluar dari gudang tercatat dengan baik.
* Mengatur penyimpanan barang agar mudah ditemukan dan aman dari kerusakan.
* Melakukan inventarisasi berkala untuk memastikan akurasi data stok.
* Bagian Produksi
* Merencanakan dan mengawasi proses produksi gorden dan aksesoris rumah lainnya.
* Memastikan produk yang dihasilkan memenuhi standar kualitas yang ditetapkan.
* Melakukan pemeliharaan dan perbaikan peralatan produksi.
* Bekerja sama dengan manajer gudang untuk memastikan kelancaran alur bahan baku dan produk jadi.

### Analisis Sistem

Toko 3 Putri Gorden, yang beralamat di Jl. Raya Urip Sumoharjo, Kerti Jayan, Pasar Banyu Urip – Kota Pekalongan, merupakan toko yang menjual berbagai jenis gorden dan aksesoris rumah lainnya. Saat ini, sistem penjualan toko ini masih sangat tradisional, di mana penjualan dilakukan secara langsung di toko, sehingga konsumen harus datang langsung ke toko untuk melihat dan membeli produk. Pemasaran yang dilakukan pun masih mengandalkan metode tradisional seperti iklan di media lokal, pamflet, dan dari mulut ke mulut. Interaksi dengan konsumen sepenuhnya terjadi secara langsung di toko, tanpa ada platform online untuk interaksi atau transaksi**.**

### Analisis Masalah

Berdasarkan hasil observasi dan wawancara yang telah dilakukan di Toko 3 Putri Gorden, dilihat dari segi pemasaran yang kurang meluas dan konsumen sulit untuk mendapatkan informasi seputar produk yang dijual serta pemesanan produk harus datang secara langsung ke toko. Hal itu tentu saja masih banyak kendala yang dihadapi terkait dengan sistem lama tersebut, dimana konsumen diharuskan berkunjung ke Toko 3 Putri Gorden membuat tidak efisien dengan mengeluarkan waktu, usaha dan tenaga. Analisis masalah akan diuraikan dengan menggunakan metode PIECES.

Tabel 4. 1 Analisis Metode PIECES

|  |  |  |
| --- | --- | --- |
| **PIECES** | **Hasil Analisis Terhadap Sistem Lama** | **Hasil yang Diharapkan pada Sistem Baru** |
| *Performance* (Kinerja) | * + Kinerja sistem penjualan masih manual dan terbatas pada transaksi tatap muka di toko, menyebabkan pelayanan yang lambat dan tidak efisien   + Proses transaksi yang memakan waktu, antrean pelanggan yang panjang, dan penanganan yang lambat pada jam sibuk. | * + Dengan sistem e-commerce berbasis website, kinerja penjualan akan meningkat melalui proses transaksi yang otomatis dan lebih cepat.   + Pelayanan yang lebih efisien dan cepat, dengan pengurangan waktu transaksi dan peningkatan kepuasan pelanggan. |
| *Information* (Informasi) | * + Informasi produk dan promosi hanya tersedia secara fisik di toko dan melalui media lokal, sehingga sulit diakses oleh calon pelanggan di luar daerah.   + Keterbatasan penyebaran informasi produk dan promosi, serta kurangnya visibilitas informasi yang update dan akurat bagi konsumen. | * + Informasi produk dan promosi dapat diakses secara online, memberikan visibilitas yang lebih luas dan update secara real-time.   + Penyebaran informasi yang lebih luas dan efekif, serta informasi produk yang selalu update dan mudah diakses oleh konsumen. |
| *Economic* (Ekonomi) | * + Pengeluaran biaya operasional yang tinggi karena kebutuhan akan staf tambahan untuk pelayanan langsung di toko.   + Efisiensi biaya rendah dan margin keuntungan yang lebih kecil karena biaya operasional yang besar. | * + Pengurangan biaya operasional dengan berkurangnya kebutuhan staf untuk pelayanan langsung dan peningkatan efisiensi.   + Efisiensi biaya yang lebih tinggi dan peningkatan margin keuntungan melalui pengurangan biaya operasional |
| *Control* (Kontrol) | * + Kontrol terhadap stok barang dan transaksi penjualan dilakukan secara manual, meningkatkan risiko kesalahan pencatatan dan kekeliruan inventaris.   + Tingginya kemungkinan kesalahan data dan kurangnya transparansi dalam pengelolaan stok dan transaksi. | * + - Sistem otomatis untuk pengelolaan stok dan transaksi penjualan, mengurangi risiko kesalahan dan meningkatkan akurasi data.     - Kontrol yang lebih baik dan transparan terhadap stok dan transaksi, serta pengurangan risiko kesalahan data. |
| *Efficiency* (Efisiensi) | * + Efisiensi waktu dan tenaga sangat rendah karena semua proses penjualan dan interaksi dengan konsumen dilakukan secara langsung.   + Proses penjualan yang memakan waktu lama dan sumber daya manusia yang tidak dimanfaatkan secara optimal. | * + Proses penjualan dan pelayanan yang otomatis, mengurangi waktu dan tenaga yang dibutuhkan.   + Peningkatan efisiensi waktu dan sumber daya, dengan pemanfaatan teknologi untuk mendukung operasional toko. |
| *Service* (Pelayanan) | * + Layanan terbatas pada jam operasional toko dan mengharuskan konsumen datang langsung untuk mendapatkan layanan.   + Keterbatasan dalam memberikan layanan 24/7 dan kesulitan dalam menjangkau pelanggan di luar daerah. | * + Layanan penjualan dan customer service yang dapat diakses secara online 24/7, memberikan kemudahan bagi konsumen.   + Peningkatan kualitas layanan dengan akses yang lebih mudah dan fleksibilitas waktu, menjangkau konsumen di luar daerah tanpa batasan waktu. |

### Analisis kebutuhan sistem

Dengan semakin pesatnya perkembangan teknologi dan internet di Indonesia, adopsi e-commerce akan memberikan banyak manfaat dan keunggulan kompetitif bagi Toko 3 Putri Gorden. Salah satu kebutuhan utama yang harus dipenuhi oleh sistem baru ini adalah kemampuan untuk menampilkan katalog produk secara lengkap dan menarik, termasuk deskripsi, harga, dan ketersediaan stok, sehingga konsumen dapat dengan mudah melihat dan memilih produk yang mereka inginkan tanpa harus datang ke toko secara fisik.

Selain itu, sistem ini harus mendukung fitur transaksi online yang aman dan efisien, memungkinkan konsumen untuk melakukan pembelian dengan mudah melalui berbagai metode pembayaran. Ini akan menghilangkan batasan geografis dan waktu, serta meningkatkan kenyamanan dan pengalaman berbelanja bagi konsumen.

Untuk mendukung pemasaran dan promosi, sistem e-commerce ini harus memiliki fitur untuk mengelola kampanye pemasaran digital, seperti penawaran khusus, diskon, dan program loyalitas pelanggan. Ini akan membantu Toko 3 Putri Gorden dalam menarik lebih banyak pelanggan dan meningkatkan penjualan. Selain itu, sistem ini juga harus mampu mengumpulkan dan menganalisis data pelanggan, perilaku pembelian, dan tren pasar, sehingga pimpinan toko dapat membuat keputusan bisnis yang lebih informatif dan strategis.

Dalam merancang sistem ini, metode waterfall digunakan dengan tahapan analisa, desain, pengkodean, pengujian, dan implementasi. Penggunaan Unified Modeling Language (UML) seperti usecase diagram, activity diagram, dan class diagram membantu dalam mendefinisikan kebutuhan sistem dan alur kerja secara jelas. Aplikasi Figma digunakan untuk tahap desain, memastikan tampilan antarmuka yang user-friendly. Framework Codeigniter dipilih untuk pengkodean, karena kemampuannya dalam mengembangkan aplikasi web yang cepat, aman, dan efisien.

Hasil akhir dari penelitian dan perancangan ini adalah sebuah sistem informasi e-commerce berbasis website yang tidak hanya memperluas pasar produk Toko 3 Putri Gorden tetapi juga mengoptimalkan kegiatan penjualan. Sistem ini memungkinkan konsumen untuk melihat katalog produk, melakukan pembelian secara online, dan menerima berbagai informasi terkait promosi dan penawaran khusus, tanpa harus mengunjungi toko secara fisik. Dengan demikian, implementasi sistem ini diharapkan dapat mengatasi kendala yang dihadapi, meningkatkan jangkauan pasar, dan memudahkan proses pembelian bagi konsumen, serta memberikan nilai tambah yang signifikan bagi Toko 3 Putri Gorden.

### Analisis penunjang keputusan

Dalam mendukung keputusan sistem e-commerce berbasis website di Toko 3 Putri Gorden, beberapa analisis telah dilakukan untuk memastikan bahwa langkah ini akan memberikan manfaat yang signifikan bagi toko. Pertama, sistem e-commerce akan membuat produk Toko 3 Putri Gorden lebih mudah diakses oleh konsumen, tidak hanya di Pekalongan, tetapi juga di seluruh Indonesia. Konsumen tidak lagi harus datang langsung ke toko untuk melihat dan membeli produk, sehingga jangkauan pasar menjadi lebih luas.

Sistem ini juga akan membantu dalam mengelola stok barang secara lebih efisien. Dengan data inventaris yang diperbarui secara real-time, toko dapat menghindari masalah kehabisan atau kelebihan stok, yang sering kali menjadi tantangan dalam penjualan langsung. Selain itu, dengan adanya platform online, toko dapat dengan mudah menjalankan promosi dan diskon, menarik lebih banyak pelanggan tanpa perlu biaya iklan yang besar.

Penjualan diperkirakan akan meningkat karena lebih banyak pelanggan yang dapat dijangkau. Selain itu, kemudahan berbelanja online akan meningkatkan kepuasan pelanggan, yang pada gilirannya dapat meningkatkan loyalitas pelanggan dan frekuensi pembelian. Operasional toko juga akan menjadi lebih efisien karena banyak proses yang dapat diotomatisasi, mengurangi kebutuhan akan tenaga kerja dan biaya operasional lainnya.

### Analisis data penunjang keputusan

Dalam mengembangkan sistem informasi penunjang keputusan, berbagai jenis data perlu diproses untuk menyediakan informasi yang relevan dan bermanfaat bagi para pimpinan. Berikut adalah analisis data pendukung keputusan yang nantinya akan diproses dalam sistem, termasuk lingkup data yang menjadi input dalam sistem yang akan dibangun.

Data Penjualan

* Produk Terlaris: Mengetahui produk-produk yang paling laris membantu toko dalam mengelola stok dan menentukan produk mana yang perlu dipromosikan lebih lanjut.
* Waktu Penjualan: Analisis waktu penjualan dapat mengidentifikasi periode puncak penjualan, sehingga toko dapat mengatur kampanye promosi dan diskon pada waktu yang tepat.

1. Data Pelanggan

* Riwayat Pembelian: Catatan transaksi sebelumnya membantu memahami perilaku pembelian pelanggan dan mengidentifikasi pelanggan setia.
* Umpan Balik Pelanggan: Ulasan dan feedback pelanggan memberikan wawasan tentang kualitas produk dan layanan, serta area yang perlu diperbaiki

1. Data Pemasaran

* Efektivitas Kampanye: Data tentang kampanye pemasaran yang telah dijalankan, seperti jumlah barang yang sudah terjual, konversi, dan penjualan yang dihasilkan, membantu menilai keberhasilan strategi pemasaran.
* Sumber Lalu Lintas: Informasi mengenai sumber lalu lintas (media sosial, mesin pencari, iklan berbayar) membantu menentukan saluran pemasaran yang paling efektif.

1. Data Keuangan

* Pendapatan: Informasi mengenai jumlah uang yang dihasilkan dari penjualan produk setiap bulan. Data ini membantu toko melihat apakah penjualan meningkat atau menurun.
* Pengeluaran: Data tentang biaya-biaya yang dikeluarkan setiap bulan, seperti biaya sewa toko, gaji karyawan, biaya listrik, dan biaya bahan baku. Ini penting untuk mengelola anggaran dan memastikan toko tidak mengeluarkan lebih dari yang diterima.
* Keuntungan: Selisih antara pendapatan dan pengeluaran. Data ini menunjukkan apakah toko menghasilkan keuntungan atau rugi. Dengan memahami ini, toko dapat menyesuaikan strategi bisnisnya untuk meningkatkan keuntungan.

1. Data Pasar

* Tren Industri: Informasi tentang tren terbaru dalam industri gorden dan aksesoris rumah, seperti desain yang sedang populer, bahan baru, dan teknologi yang digunakan. Data ini membantu toko untuk tetap up-to-date dengan perkembangan pasar.
* Permintaan Pasar: Informasi tentang tingkat permintaan untuk produk-produk tertentu. Data ini membantu toko untuk memahami kebutuhan dan preferensi konsumen, serta menyesuaikan stok dan penawaran produk.

1. Data Kompetitif

* Strategi Kompetitor: Informasi mengenai strategi pemasaran, harga, dan inovasi produk dari pesaing membantu toko dalam merumuskan strategi yang kompetitif.
* Posisi Kompetitor: Data mengenai posisi Toko 3 Putri Gorden di pasar dibandingkan dengan kompetitor utama membantu dalam mengevaluasi kekuatan dan kelemahan toko.

1. Data oprasional

* Efisiensi Proses: Informasi mengenai efisiensi proses operasional di toko, termasuk waktu pemrosesan pesanan dan pengiriman, membantu dalam meningkatkan efisiensi dan mengurangi biaya.
* Biaya Operasional: Data mengenai biaya operasional sehari-hari seperti gaji karyawan, biaya listrik, dan biaya sewa membantu dalam manajemen biaya dan pengendalian anggaran

Sistem informasi yang akan dibangun akan mengintegrasikan semua jenis data ini dari berbagai sumber, melakukan pemrosesan dan analisis yang mendalam, dan menyajikan informasi yang relevan dan akurat untuk mendukung pengambilan keputusan yang lebih baik dan strategis oleh para pimpinan. Dengan memiliki akses ke data yang komprehensif dan terstruktur, pimpinan dapat membuat keputusan yang lebih tepat waktu, berdasarkan data, dan efektif dalam mengarahkan perusahaan menuju tujuan strategisnya.

### Analisis kebutuhan / Alat

Analisis kebutuhan /alat berfungsi untuk menentukan perangkat apa saja yang dibutuhkan dalam pembangunan sistem informasi e-commerce tersebut. Dengan analisis kebutuhan diharapkan sistem yang akan dibangun dapat diuraikan secara utuh menjadi komponen-komponen dasar dengan tujuan identifikasi dan analisis ini juga dilakukan untuk menjamin bahwa sistem yang akan dibangun sesuai dengan kebutuhan dari objek penelitian. Perancangan sistem baru diharapkan dapat mengubah dan mempercepat serta mengoptimalkan waktu dengan baik. Analisis kebutuhan menjadi 2 bagian yaitu:

Kebutuhan Perangkat keras

|  |  |
| --- | --- |
| * Laptop | : Acer Aspire A314-32 |
| * *Processor* | : Intel(R) Celeron(R) N4120 CPU @ 1.10GHz 1.10 GHz |
| * *Memory* | : 4 GB |
| * *Storage* | : 237 GB |
| * Monitor * Printer | : 14 Inch  : Inkjet Printer |

Kebutuhan Perangkat lunak

|  |  |
| --- | --- |
| * Sistem Operasi | : Windows 11 |
| * Database | : XAMPP |
| * Framework | : Codiegniter |
| * Aplikasi Pembuatan | : Visual Studio Code |
| * Perancangan UI | : Figma Mockup |
| * Perancangan UML | : StarUML |

### Analisis Biaya

Proses pembuatan sistem informasi monitoring ini memerlukan tenaga serta biaya yang harus dikeluarkan. Adapun biaya yang diperlukan untuk membangun sistem informasi reservasi ini yaitu sebagai berikut:

Tabel 4. 2 Analisis Biaya

|  |  |  |
| --- | --- | --- |
| **No.** | **Jenis Kebutuhan** | **Biaya** |
| 1 | ATK | Rp. 500.000,- |
| 2 | Laptop | Rp. 5.000,000.- |
| 2 | Wifi | Rp. 350.000,- |
| 3 | Analisis | Rp. 500.000,- |
| 4 | *Programming* | Rp. 1.000.000,- |
| 5 | Transportasi | Rp. 250.000,- |
| **Total** | | **Rp. 7.600.000,-** |

## Perancangan

Perancangan sistem informasi e-commerce berbasis website ini akan dibangun menggunakan *Unified Modelling Language* (UML) sebagai Bahasa pemodelan pembangunan website ini dilakukan dengan menggunakan *tools* utama yaitu sebagai berikut.

### Pemodelan UML (Unified Modelling Language)

Diagram UML yang dilakukan dalam perancangan sistem informasi *e-commerce* berbasis website ada 4 diagram, yaitu *use case* diagram, *activity* diagram, *class* diagram, dan *sequence* diagram.

* + - 1. *Use Case* Diagram

Berikut merupakan *use case* dari pembaharuan sistem informasi *e-commerce* berbasis website :



Gambar 4. 2 Use Case Diagram

U*se case* diagram di atas menunjukkan aktor yang terlibat dalam sistem informasi *e-commerce* berbasis website. Terdapat 2 aktor yang masing-masing memiliki hak akses sebagai berikut :

Deskripsi Aktor

Tabel 4. 3 Deskripsi Aktor

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Aksi** |
| 1 | Admin | * Admin bertanggung jawab untuk menambahkan, mengubah, dan menghapus produk dari sistem. Mereka memastikan bahwa informasi produk selalu up-to-date, termasuk deskripsi, harga, dan gambar produk. * Admin mengelola pesanan yang masuk, termasuk melihat detail pesanan, memproses pesanan, dan mengkonfirmasi pengiriman. Mereka memastikan bahwa setiap pesanan diproses dengan tepat dan efisien. * Admin mengelola konten di halaman website, seperti halaman beranda, tentang kami, diskon dan kontak. Mereka memastikan bahwa informasi yang disajikan kepada konsumen selalu relevan dan akurat. |
| 2 | Customer | * Customer dapat membuat akun baru dan login ke sistem untuk melakukan transaksi. Mereka mengelola informasi akun mereka, termasuk detail pribadi dan preferensi pembayaran. * ustomer dapat mencari dan melihat produk yang tersedia di toko. * Customer dapat menghubungi layanan pelanggan melalui formulir kontak jika mereka memiliki pertanyaan atau masalah. |

* + - * 1. Deskripsi *Usecase*

Berikut adalah deskripsi use case pada gambar use case Diagram *e-commerce* berbasis website :

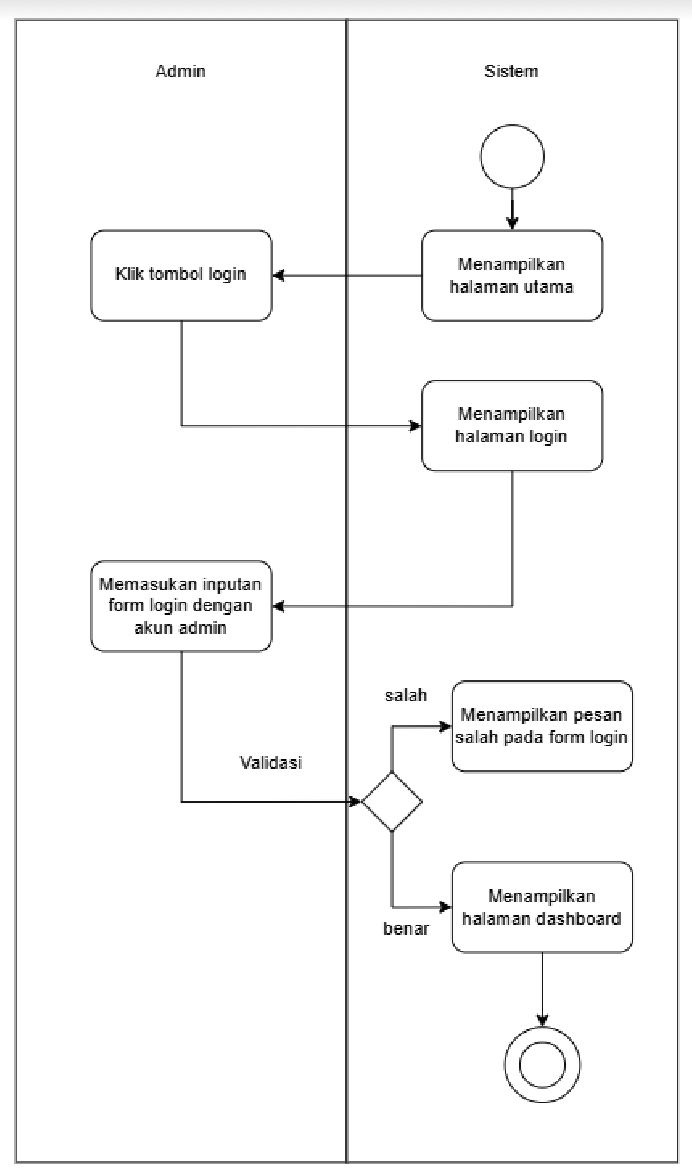
Tabel 4. 4 Deskripsi Use Case

|  |  |  |  |
| --- | --- | --- | --- |
| No | Aktor | *Use Case* | Deskripsi |
| 1. | Admin | Kelola Produk | Admin dapat menambahkan, mengubah, dan menghapus produk dari sistem. |
|  |  | Kelola Pesanan | Admin mengelola pesanan yang masuk, termasuk melihat detail pesanan, memproses pesanan, dan mengkonfirmasi pengiriman. |
|  |  | Kelola Pengguna | Admin dapat menambahkan, mengubah, dan menghapus akun pengguna lain. |
|  |  | Kelola Konten | Admin mengelola konten di halaman website, seperti halaman beranda, tentang kami, diskon dan kontak. |
| 2. | Customer | Registrasi dan Login | Customer dapat membuat akun baru dan login ke sistem untuk melakukan transaksi. |
|  |  | Penjelajahan Produk | Customer dapat mencari dan melihat produk yang tersedia di toko. |
|  |  | Proses Pembelian | Customer dapat melakukan checkout. |

* + - 1. *Activity* Diagram

Activity Diagram menggambarkan urutan aktivitas proses pada sebuah sistem. Berikut adalah activity diagram pada website *e-commerce*

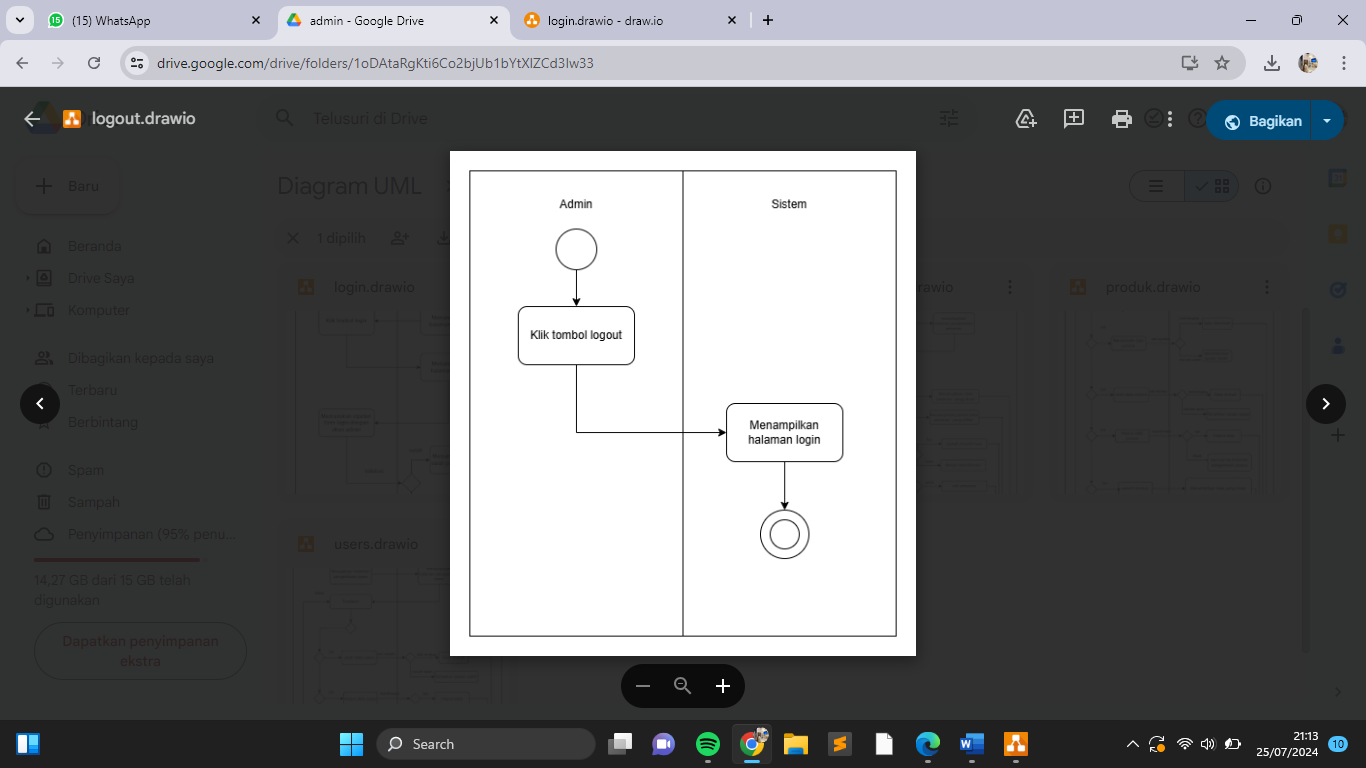
1. *Activity* diagram *Login* Admin



Gambar 4. 3 Activity Diagram Login Admin

Pada gambar *activity* diagram *login* admin menggambarkan proses *login* admin pada sistem, yaitu dimulai ketika admin mengklik tombol "Login" pada halaman utama atau halaman login sistem. Setelah mengklik tombol tersebut, admin diarahkan ke halaman form login, di mana mereka diminta untuk memasukkan username dan password yang telah terdaftar sebagai akun admin. Admin mengisi kedua kolom tersebut dengan informasi yang benar dan kemudian mengklik tombol "Submit" atau "Login" untuk mengirimkan informasi tersebut ke sistem. Setelah form login dikirimkan, sistem memproses dan memvalidasi kredensial yang diberikan oleh admin. Validasi ini mencakup pengecekan apakah username dan password yang dimasukkan sesuai dengan data yang tersimpan dalam sistem. Jika validasi berhasil dan kredensial cocok, sistem akan memberikan akses kepada admin dan menampilkan halaman dashboard, yang merupakan halaman utama bagi admin untuk mengelola berbagai aspek sistem. Namun, jika validasi gagal karena username atau password yang dimasukkan tidak cocok dengan data, sistem akan menampilkan pesan kesalahan pada form login, Admin kemudian dapat mencoba lagi dengan memasukkan informasi login yang benar.

1. Activity diagram Logout



Gambar 4. 4 Activity Diagram Logout

Pada gambar *actvitiy* diagram *logout* admin menggambarkan proses *logout* admin pada sistem, yaitu dimulai ketika admin mengklik tombol "Logout" yang terletak di bagian atas atau menu dashboard. Saat admin mengklik tombol tersebut, sistem secara otomatis memproses permintaan logout. Setelah sesi diakhiri, sistem secara otomatis mengarahkan admin kembali ke halaman login.

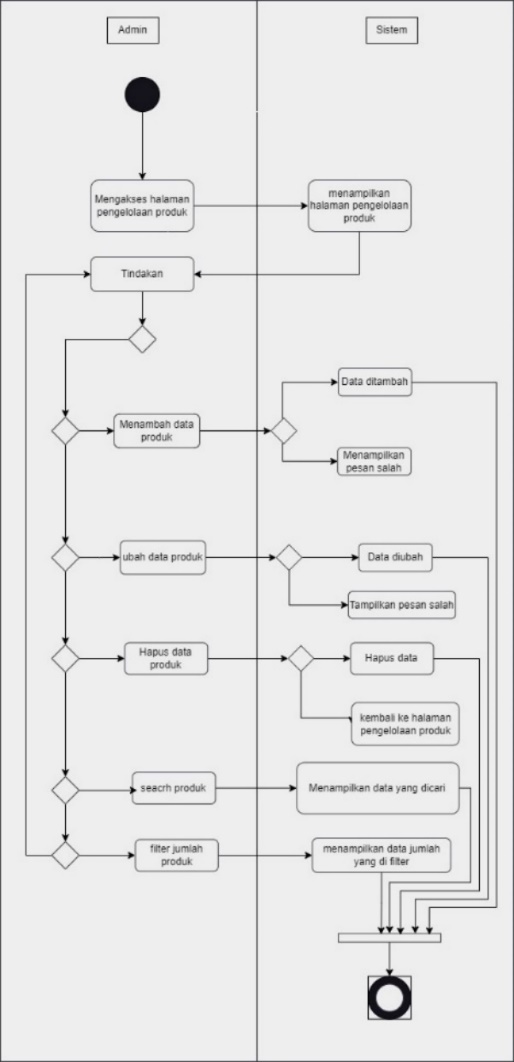
1. Activity diagram Pesanan



Gambar 4. 5 Activity Diagram Pesanan

Pada gambar *activity* diagram *user* ini menggambarkan alur kerja sistem pengelolaan pesanan. Sistem dimulai dengan admin mengakses halaman pengelolaan pesanan. Sistem kemudian menampilkan halaman pengelolaan pesanan kepada admin. Admin dapat melakukan tindakan seperti mencari pesanan, memfilter jumlah pesanan, atau mengkonfirmasi pesanan. Jika admin mencari pesanan, sistem akan menampilkan data pesanan yang dicari. Jika admin memfilter jumlah pesanan, sistem akan menampilkan jumlah data pesanan yang difilter. Jika admin mengkonfirmasi pesanan, sistem akan menandai pesanan tersebut sebagai sudah dikonfirmasi. Jika admin mengedit pesanan, sistem akan menampilkan formulir edit pesanan. Jika admin memilih untuk tidak mengkonfirmasi pesanan, sistem akan menandai pesanan tersebut sebagai belum dikonfirmasi. Jika admin memasukkan data yang salah, sistem akan menampilkan pesan kesalahan. Setelah semua tindakan selesai, sistem akan kembali ke halaman pengelolaan pesanan.

1. Activity diagram Produk



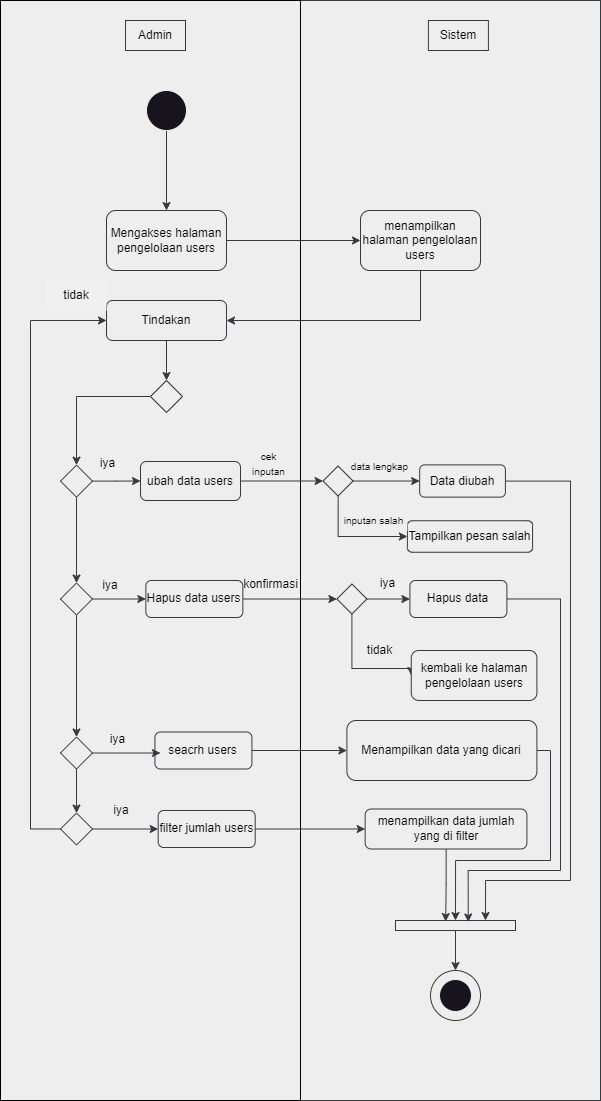
Gambar 4. 6 Activity Diagram Produk

Pada gambar *activity* diagram produk menggambarkan ketika Admin ingin mengelola produk dalam sistem , pertama-tama Admin harus mengakses halaman pengelolaan produk. Admin masuk ke sistem menggunakan akun yang telah terdaftar, Setelah Admin memilih opsi untuk mengelola produk, sistem menampilkan halaman pengelolaan produk. Halaman ini mencakup daftar produk yang saat ini tersedia di sistem, serta tombol dan form untuk menambah, mengedit, dan menghapus produk. Admin kemudian dapat melakukan berbagai tindakan pengelolaan produk sesuai kebutuhan.

Jika Admin ingin menambahkan produk baru, Admin mengklik tombol "Tambah Produk Baru". Sistem kemudian menampilkan form untuk memasukkan detail produk baru, seperti nama produk, deskripsi, harga, dan gambar. Admin mengisi form tersebut dengan informasi yang diperlukan dan mengklik tombol "Simpan". Sistem memproses data yang dimasukkan, menyimpan produk baru, dan menampilkan pesan konfirmasi bahwa produk telah berhasil ditambahkan. Untuk mengedit data produk yang sudah ada, Admin memilih produk dari daftar yang ingin diubah dan mengklik tombol "Edit". Sistem kemudian menampilkan form dengan informasi produk yang sudah ada. Admin mengubah informasi yang diperlukan dan mengklik tombol "Simpan". Sistem memproses perubahan tersebut, memperbarui data produk di database, dan menampilkan pesan konfirmasi bahwa produk telah berhasil diperbarui. Jika Admin perlu menghapus produk, Admin memilih produk dari daftar dan mengklik tombol "Hapus". Sistem akan meminta konfirmasi untuk memastikan bahwa Admin benar-benar ingin menghapus produk tersebut. Setelah Admin mengonfirmasi penghapusan, sistem menghapus produk dari database dan menampilkan pesan konfirmasi bahwa produk telah berhasil dihapus.

Setelah semua tindakan pengelolaan produk selesai, sistem akan menampilkan data yang dicari atau jumlah data yang difilter sesuai permintaan Admin.

1. Activity diagram User



Gambar 4. 7 Activity Diagram User

Pada gambar *activity* diagram *user* menggambarkan ketika Admin mengakses halaman pengelolaan user dalam sistem, sistem akan terlebih dahulu menampilkan halaman pengelolaan user yang mencakup daftar pengguna yang saat ini terdaftar dalam sistem, serta berbagai tombol dan form untuk mengelola data user. Dari sini, Admin memiliki beberapa pilihan tindakan yang dapat dilakukan, seperti mengubah data user, menghapus user, melakukan pencarian user, atau melakukan filter jumlah user.

Jika Admin memutuskan untuk tidak melakukan tindakan apa pun, Admin dapat kembali ke halaman awal dengan memilih opsi yang sesuai. Namun, jika Admin memilih untuk melakukan perubahan data user, mereka akan memilih user dari daftar dan mengklik tombol "Edit". Sistem kemudian menampilkan form dengan informasi user yang sudah ada. Admin mengubah informasi yang diperlukan dan mengklik tombol "Simpan". Jika inputan yang diberikan oleh Admin lengkap dan sesuai, sistem akan memproses perubahan tersebut dan memperbarui data user. Sebaliknya, jika inputan tidak lengkap atau salah, sistem akan menampilkan pesan kesalahan yang relevan untuk membantu Admin memperbaiki inputan tersebut. Selain mengubah data user, Admin juga memiliki opsi untuk menghapus user dari sistem. Admin memilih user dari daftar dan mengklik tombol "Hapus". Sistem akan meminta konfirmasi untuk memastikan bahwa Admin benar-benar ingin menghapus user tersebut. Jika Admin mengonfirmasi penghapusan, sistem akan menghapus data user dari database dan menampilkan pesan konfirmasi bahwa user telah berhasil dihapus.

Jika Admin membatalkan konfirmasi, sistem akan kembali ke halaman pengelolaan user tanpa melakukan perubahan apa pun. Admin juga bisa melakukan pencarian user dengan memasukkan nama atau kriteria tertentu ke dalam form pencarian. Jika user yang dicari ditemukan, sistem akan menampilkan data user yang sesuai dengan kriteria pencarian. Hal ini memudahkan Admin untuk menemukan dan mengelola data user dengan lebih efisien. Selain itu, Admin dapat melakukan filter jumlah user berdasarkan kriteria tertentu, seperti status aktif atau peran dalam sistem. Admin memilih opsi filter yang diinginkan, dan sistem kemudian menampilkan daftar user yang sesuai dengan filter yang telah diterapkan. Sistem menampilkan hasil filter ini untuk memberikan gambaran yang lebih jelas dan terfokus tentang data user yang sedang dikelola. Dengan alur kerja ini, Admin dapat dengan mudah mengelola data user dalam sistem, memastikan bahwa informasi user selalu akurat dan up-to-date, serta menghapus user yang tidak lagi diperlukan. Proses ini juga memastikan bahwa sistem menampilkan pesan kesalahan yang jelas jika terjadi kesalahan input, serta memberikan konfirmasi untuk setiap tindakan penting seperti penghapusan user, sehingga meminimalkan risiko kesalahan dan memastikan bahwa pengelolaan user dilakukan dengan tepat dan efisien.

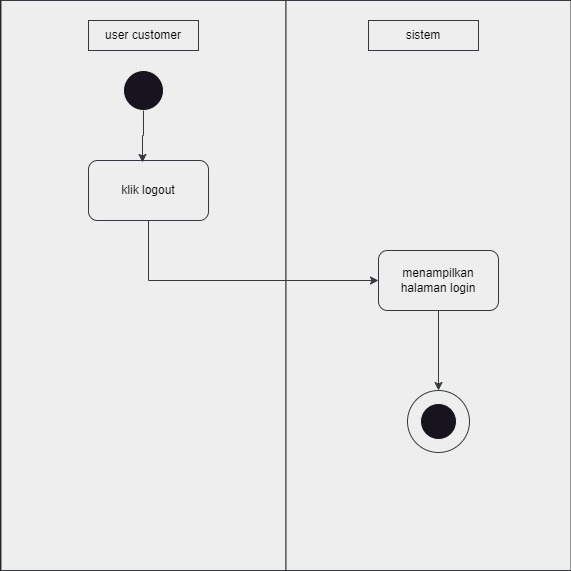
1. Activity diagram Login Customer



Gambar 4. 8 Activity Diagram Login Customer

Pada gambar *activity* diagram *login customer* ini menggambarkan proses *login customer* pada sistem, yaitu proses login customer pada sistem dimulai ketika customer mengklik tombol "Login" yang tersedia di halaman utama atau halaman login sistem. Tindakan ini mengarahkan customer ke halaman form login, di mana mereka diminta untuk memasukkan informasi login mereka, yaitu username atau email dan password yang telah terdaftar dalam sistem. Customer kemudian mengisi form login dengan memasukkan username atau email serta password mereka. Setelah mengisi form login, customer menekan tombol "Login" untuk mengirimkan data login tersebut ke sistem. Pada tahap ini, sistem akan mulai memproses dan memvalidasi input yang diberikan oleh customer. Jika validasi berhasil dan cocok dengan data yang ada, sistem akan memberikan akses kepada customer dan secara otomatis mengarahkan mereka ke halaman beranda. Namun, jika validasi gagal karena username atau email dan password yang dimasukkan tidak cocok sistem akan menampilkan pesan kesalahan pada halaman login.

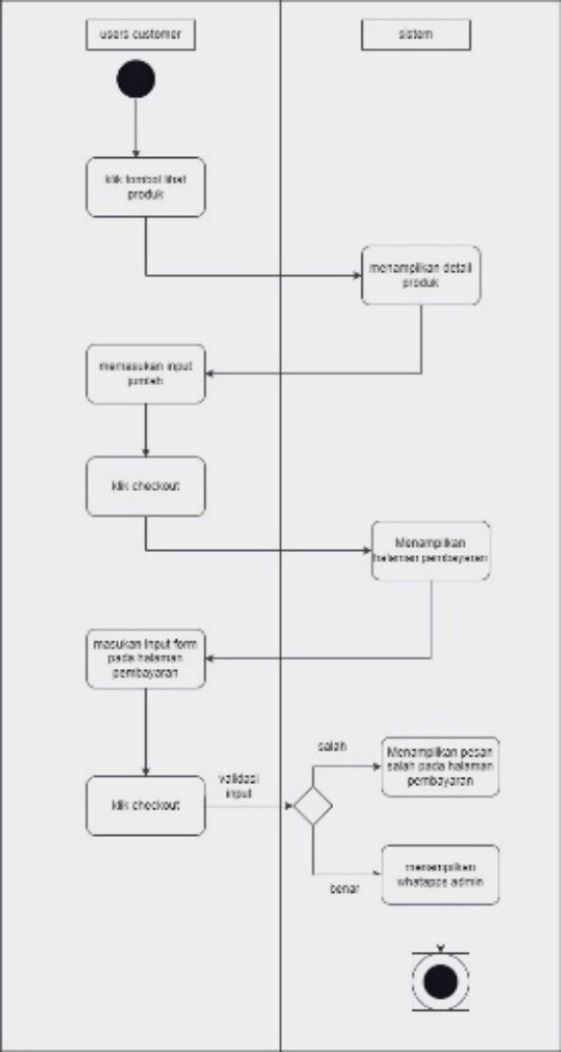
1. *Activity* diagram *Logout Customer*



Gambar 4. 9 Activity Diagram Logout Customer

Pada gambar *actvitiy* diagram *logout* *customer* menggambarkan proses *logout customer* pada sistem, yaitu klik tombol "Logout" Ketika customer mengklik tombol tersebut, Setelah sesi diakhiri, sistem secara otomatis mengarahkan customer kembali ke halaman login. Halaman login ini adalah halaman awal di mana customer pertama kali memasukkan informasi login mereka untuk mengakses sistem. Tampilan halaman login yang muncul kembali memastikan bahwa customer tidak lagi memiliki akses ke konten atau fitur yang memerlukan autentikasi, seperti halaman profil atau riwayat pesanan.

1. *Activity* diagram *Checkout* barang



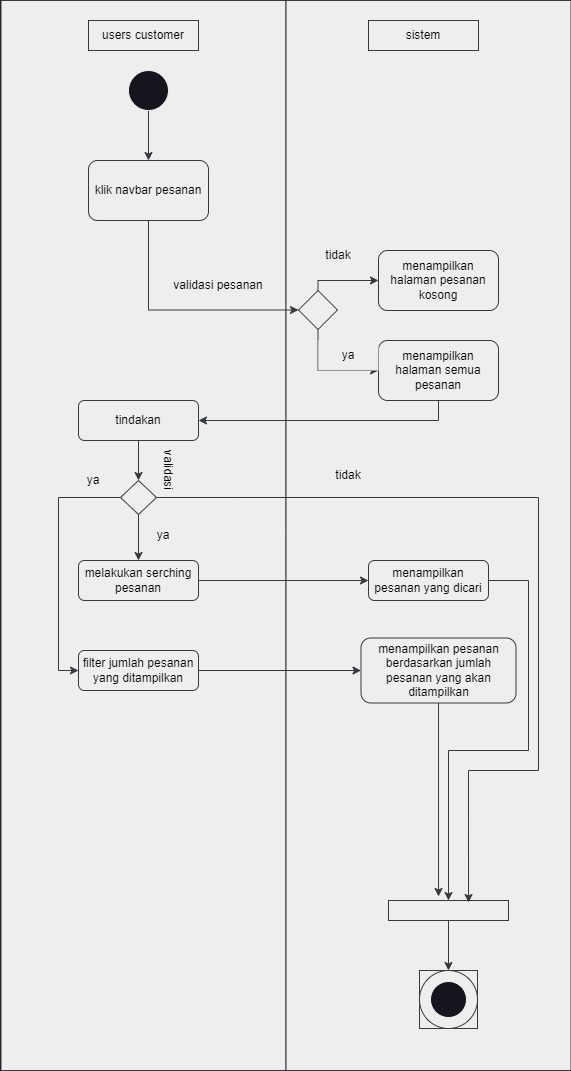
Gambar 4. 10 Activity Diagram Checkout Barang

Pada activity diagram checkout barang menggambarkan Proses pembelian produk di sistem e-commerce dimulai ketika pengguna, yang dalam hal ini adalah customer, pertama kali melihat daftar produk yang tersedia. Customer dapat melihat berbagai produk yang ditawarkan, termasuk deskripsi, harga, dan gambar produk. Setelah customer menemukan produk yang ingin dibeli, mereka akan memilih produk tersebut. Pada tahap ini, customer juga perlu memasukkan jumlah produk yang ingin dibeli sesuai dengan kebutuhan mereka. Setelah itu customer akan melanjutkan proses dengan menekan tombol checkout. Tindakan ini mengarahkan customer ke halaman checkout, di mana mereka diminta untuk mengisi formulir pembayaran. Formulir pembayaran ini mencakup informasi penting seperti nama, alamat pengiriman, metode pembayaran, dan detail kartu kredit atau metode pembayaran lainnya.

Setelah mengisi formulir pembayaran dengan lengkap, customer menekan tombol checkout untuk melanjutkan. Pada titik ini, sistem akan memvalidasi semua input yang telah dimasukkan oleh customer. Validasi ini penting untuk memastikan bahwa semua informasi yang diberikan adalah benar dan memenuhi persyaratan yang diperlukan. Jika sistem menemukan bahwa input yang diberikan tidak valid, sistem akan menampilkan pesan kesalahan yang jelas. Pesan kesalahan ini akan menjelaskan masalah yang ditemukan, seperti informasi alamat pengiriman yang tidak lengkap atau informasi lain nya, dan akan mengarahkan customer kembali ke formulir pembayaran untuk memperbaiki kesalahan tersebut.

Jika semua input valid, sistem akan melanjutkan dengan menampilkan halaman pembayaran. Halaman ini memberikan informasi ringkasan tentang pesanan, termasuk detail produk, total biaya, dan metode pembayaran yang dipilih. Pada tahap ini, customer akan melakukan pembayaran sesuai dengan metode yang telah dipilih. Setelah pembayaran berhasil dilakukan, sistem akan menampilkan pesan konfirmasi kepada customer yang menyatakan bahwa pembelian telah berhasil diproses. Selain menampilkan pesan konfirmasi kepada customer, sistem juga akan mengirimkan pesan notifikasi kepada admin. Pesan notifikasi ini berfungsi untuk memberi tahu admin bahwa ada pesanan baru yang perlu ditindaklanjuti. Admin kemudian akan memproses pesanan tersebut, termasuk menyiapkan produk untuk pengiriman dan mengatur logistik yang diperlukan untuk memastikan bahwa produk sampai ke tangan customer sesuai dengan yang dijanjikan.

1. Activity diagram Riwayat Pesanan



Gambar 4. 11 Activity Diagram Riwayat Pesanan

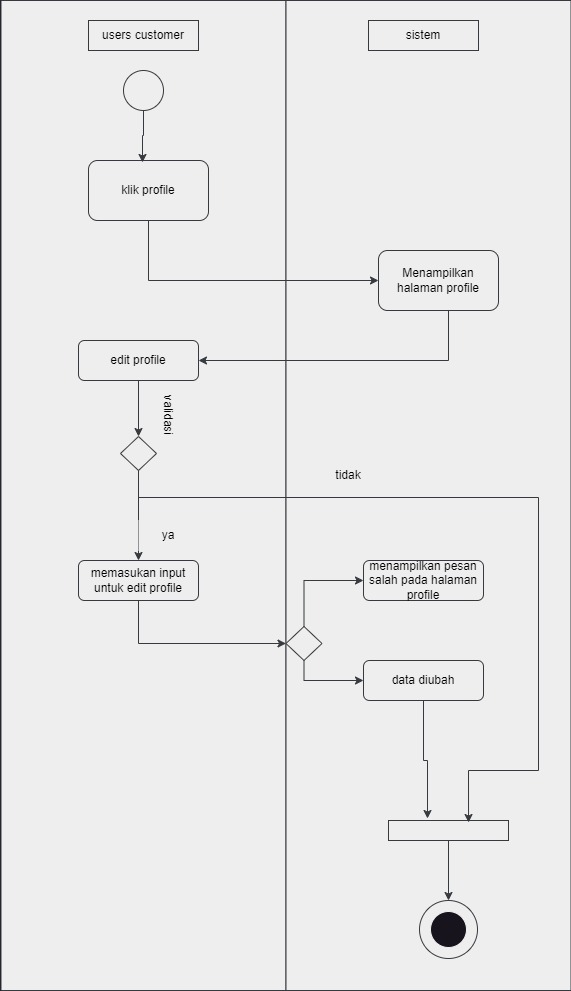
Pada gambar activity diagram riwayat pesanan menggambarkan alur kerja langkah-langkah yang diambil oleh pengguna ketika berinteraksi dengan sistem untuk melihat pesanan mereka. Proses dimulai ketika pengguna mengklik opsi "pesanan". Ini adalah langkah awal yang memicu sistem untuk mulai memproses permintaan pengguna.

Setelah pengguna mengklik "pesanan", sistem melakukan validasi terhadap pesanan pengguna. Validasi ini bertujuan untuk memastikan bahwa pengguna yang meminta untuk melihat pesanan memiliki hak akses yang tepat dan bahwa ada pesanan yang terkait dengan akun pengguna tersebut. Jika sistem menemukan bahwa pesanan tidak valid, misalnya karena tidak ada pesanan yang terkait dengan akun atau terjadi kesalahan teknis, sistem akan menampilkan halaman kosong untuk memberi tahu pengguna bahwa tidak ada pesanan yang dapat ditampilkan. Namun, jika validasi berhasil dan pesanan dianggap valid, sistem akan melanjutkan dengan menampilkan semua pesanan yang terkait dengan akun pengguna. Pada tahap ini, pengguna memiliki dua opsi: mencari pesanan yang spesifik atau melihat semua pesanan yang ada. Kedua opsi ini memberikan fleksibilitas bagi pengguna untuk mengelola dan memeriksa pesanan mereka sesuai kebutuhan.

Jika pengguna memilih untuk mencari pesanan yang spesifik, mereka akan memasukkan detail pencarian seperti nomor pesanan, tanggal pesanan, atau kata kunci tertentu yang terkait dengan pesanan. Sistem kemudian akan memproses permintaan pencarian ini dan menampilkan pesanan yang sesuai dengan kriteria pencarian yang dimasukkan. Ini memudahkan pengguna untuk menemukan pesanan tertentu tanpa harus menelusuri seluruh daftar pesanan. Sebaliknya, jika pengguna memilih untuk melihat semua pesanan, sistem akan menampilkan daftar semua pesanan yang ada, tetapi dengan fitur tambahan untuk memfilter jumlah pesanan yang ditampilkan. Misalnya, pengguna dapat memilih untuk melihat pesanan dalam rentang tanggal tertentu atau pesanan dengan status tertentu (seperti pesanan yang sedang diproses, dikirim, atau selesai). Setelah filter diterapkan, sistem akan menampilkan pesan yang difilter sesuai dengan kriteria yang telah dipilih oleh pengguna. Setelah sistem menampilkan pesanan yang difilter atau hasil pencarian, pengguna dapat melihat detail pesanan. Detail ini mungkin mencakup informasi seperti produk yang dipesan, jumlah, harga, status pengiriman, dan informasi lainnya yang relevan. Pengguna dapat memeriksa informasi ini untuk memastikan pesanan mereka telah diproses dengan benar dan memantau status pengiriman.

Selain melihat detail pesanan, pengguna juga dapat berinteraksi lebih lanjut dengan sistem. Ini bisa mencakup tindakan seperti mengubah alamat pengiriman, membatalkan pesanan, atau menghubungi layanan pelanggan jika ada masalah dengan pesanan.

1. Activity diagram Profil

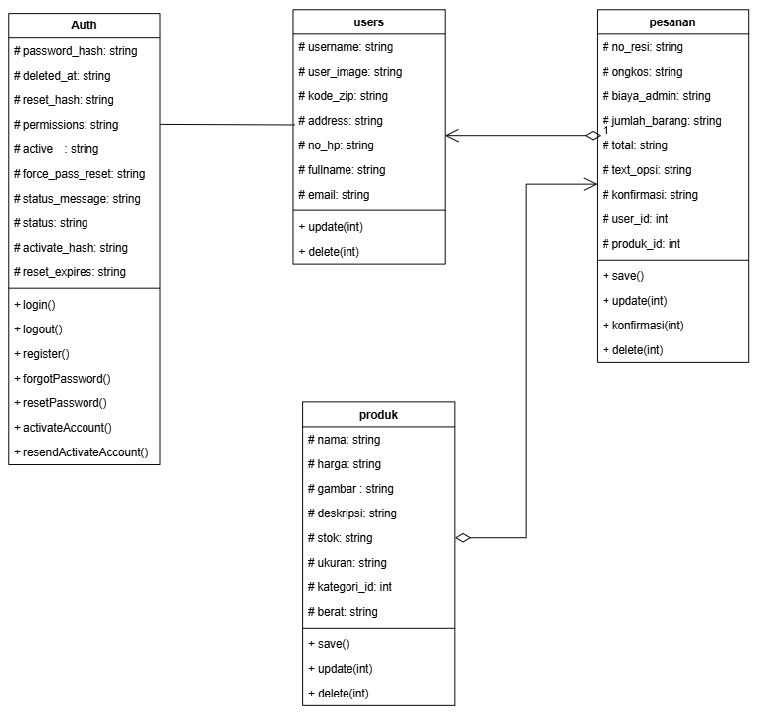


Gambar 4. 12 Activity Diagram Profil

Pada gambar activity diagram profil menggambarkan Proses edit profile pada sistem dimulai ketika user mengklik opsi profile. Setelah itu, sistem akan menampilkan halaman profile yang berisi informasi user saat ini. User kemudian dapat melakukan perubahan pada informasi profile mereka, seperti nama, email, atau informasi lainnya. Setelah melakukan perubahan, user mengirimkan input untuk validasi. Jika validasi berhasil, sistem akan menyimpan perubahan dan memperbarui data profile. Namun, jika validasi gagal, sistem akan menampilkan pesan kesalahan pada halaman profile, memberi tahu user tentang masalah yang perlu diperbaiki. Setelah proses edit profile selesai, sistem akan menampilkan halaman profile yang telah diperbarui, mengonfirmasi bahwa perubahan telah berhasil dilakukan, dan proses pun selesai.

* + - 1. *Class* Diagram

Class Diagram adalah diagram yang menunjukkan class-class yang ada dari sebuah sistem dan hubungannya dengan logika. Berikut perancangan class diagram:



Gambar 4. 13 Class Diagram

1. **Struktur Tabel**

Berikut adalah perancangan basis data yang dirancang untuk membangun *e-commerce* berbasis website. Perancangan basis data terdiri dari field, type, size, index dan keterangan. Adapun rancangan basis data dari website *e-commerce*:

Tabel 4. 5 Activation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Id\_User | Int | 11 | Primary  Key | Id User |
| 2 | name | Varchar | 255 |  | name |
| 3 | description | Varchar | 255 |  | description |

Tabel 4. 6 Groups

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Group\_id | Int | 11 | MUL | Group id |
| 2 | Permission\_id | Int | 11 | MUL | Permission id |

Tabel 4. 7 Groups User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Group\_id | Int | 11 | MUL | Group id |
| 2 | user\_id | Int | 11 | MUL | user id |

Tabel 4. 8 Groups

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary  Key | Id User |
| 2 | name | Varchar | 255 |  | name |
| 3 | description | Varchar | 255 |  | Description |

Tabel 4. 9 Login

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary Key | Id |
| 2 | Ip\_address | Varchar | 255 |  | Ip address |
| 3 | Email | Varchar | 255 | MUL | Email |
| 4 | User\_id | Int | 11 | MUL | User id |
| 5 | Date | Datetime |  |  | Date |
| 6 | Success | Etinyint | 1 |  | Success |

Tabel 4. 10 Permissions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary  Key | Id User |
| 2 | name | Varchar | 255 |  | name |
| 3 | description | Varchar | 255 |  | Description |

Tabel 4. 11 Reset

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary Key | Id |
| 2 | Email | Varchar | 255 |  | Email |
| 3 | Ip\_address | Varchar | 255 |  | Ip address |
| 4 | User\_agent | Varchar | 255 |  | User agent |
| 5 | Token | Varchar | 255 |  | Token |
| 6 | Created\_at | Datetime |  |  | Created at |

Tabel 4. 12 Tokens

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary Key | Id |
| 2 | selector | Varchar | 255 | MUL | Selector |
| 3 | Hashed validator | Varchar | 255 |  | Hashed validator |
| 4 | User\_id | Int | 11 | MUL | User id |
| 5 | expires | datetime |  |  | expires |

Tabel 4. 13 User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | user\_id | Int | 11 | MUL | user id |
| 2 | Permission\_id | Int | 11 | MUL | Permission id |

Tabel 4. 14 Kategori

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type | Size | Index | Keterangan |
| 1 | id | Int | 11 | MUL | Group id |
| 2 | Jenis | varchar | 255 | Unique | Permission id |

Tabel 4. 15 Order

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type S | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary  Key | Id |
| 2 | No\_Resi | Varchar | 50 |  | No resi |
| 3 | Ongkos | Varchar | 255 |  | Ongkos |
| 4 | Biaya\_admin | Varchar | 255 |  | Biaya admin |
| 5 | Jumlah\_barang | Varchar | 255 |  | Jumlah barang |
| 6 | Total | Varchar | 255 |  | Total |
| 7 | Konfirmasi | Varchar | 2 |  | Konfirmasi |
| 8 | Text\_opsi | Varchar | 255 |  | Text opsi |
| 9 | User\_id | Int | 11 | MUL | User id |
| 10 | Produk\_id | Int | 11 | MUL | Produk id |
| 11 | Created\_at | Datetime |  |  | Created id |
| 12 | Updated\_at | Datetime |  |  | Updated id |

Tabel 4. 16 Produk

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type S | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary  Key | Id |
| 2 | Nama | Varchar | 255 |  | Nama |
| 3 | Harga | Varchar | 255 |  | Harga |
| 4 | Gambar | Varchar | 255 |  | Gambar |
| 5 | Deskripsi | Varchar | 255 |  | Deskripsi |
| 6 | Stok | Varchar | 255 |  | Stok |
| 7 | Ukuran | Varchar | 255 |  | Ukuran |
| 8 | Kategori\_id | Int | 11 | MUL | Kategori id |
| 9 | Berat | Varchar | 255 |  | Berat |
| 10 | Created\_at | Datetime |  |  | created id |
| 11 | Updated\_at | Datetime |  |  | updated id |

Tabel 4. 17 User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Field | Type S | Size | Index | Keterangan |
| 1 | Id | Int | 11 | Primary  Key | Id |
| 2 | Email | Varchar | 255 | Unique | Email |
| 3 | Username | Varchar | 30 | Unique | Username |
| 4 | Fullname | Varchar | 255 |  | Full name |
| 5 | Address | Varchar | 255 |  | Address |
| 6 | No\_hp | Varchar | 20 |  | Nomer hp |
| 7 | Country | Varchar | 255 |  | Country |
| 8 | Kode\_zip | Varchar | 20 |  | Kode zip |
| 9 | User\_image | Varchar | 255 |  | User image |
| 10 | Password\_hash | Varchar | 255 |  | Password hash |
| 11 | Reset\_hash | Varchar | 255 |  | Reset hash |
| 12 | Reset\_at | Datetime |  |  | Reset at |
| 13 | Reset\_expires | Datetime |  |  | Reset expires |
| 14 | Activate\_hash | Varchar | 255 |  | Activate hash |
| 15 | Status | Varchar | 255 |  | Status |
| 16 | Status\_message | Varchar | 255 |  | Status message |
| 17 | Active | Tinyint | 1 |  | Active |
| 18 | Force\_pass\_reset | Tinyint | 1 |  | Force pass reset |
| 19 | Created\_at | Datetime |  |  | Created id |
| 20 | Updated\_at | Datetime |  |  | Updated id |
| 21 | Deleted\_at | datetime |  |  | Deleted id |

1. **Desain**

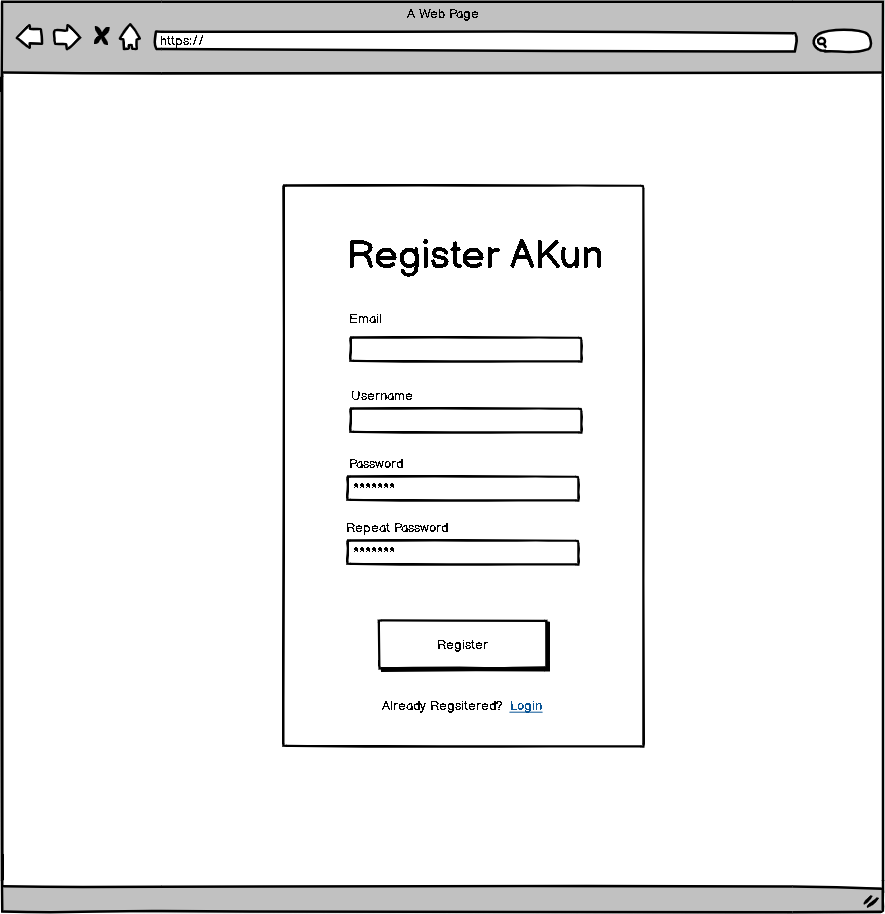
Desain user interface pada website *e-commerce* menggunakan figma diantaranya:

*User interface* login *customer*



Gambar 4. 14 User Interface Login Customer

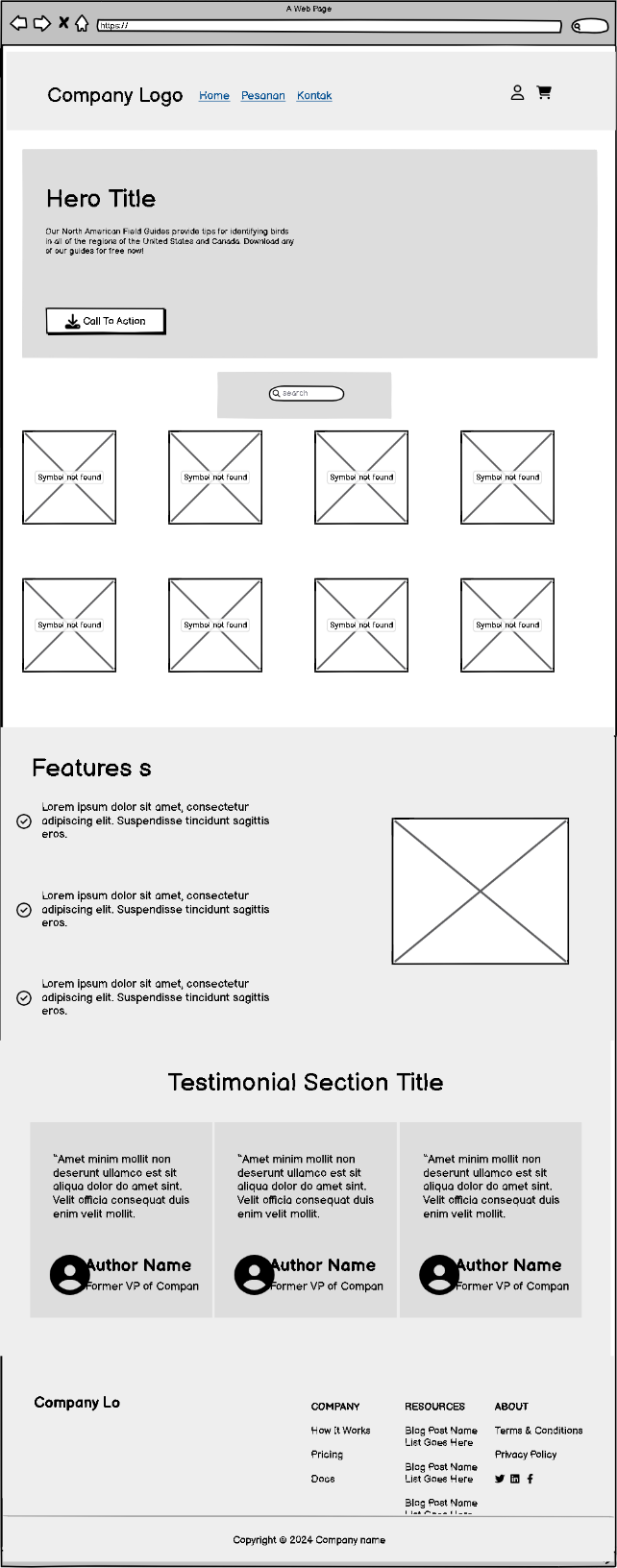
*User interface register customer*



Gambar 4. 15 User Interface Register Customer

*User interface* halaman utama

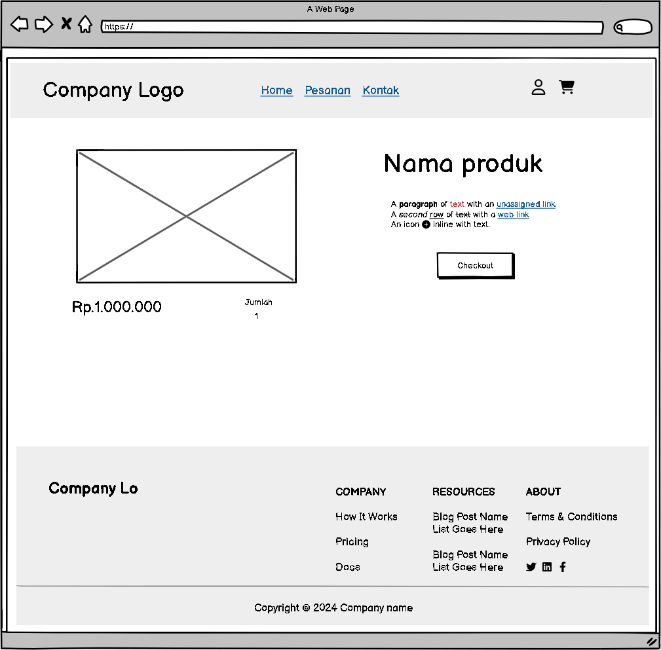
Pada *user interface* halaman utama *customer* bisa melihat halaman dashboard sebagai halaman utama, halaman ini memuat informasi produk yang dijual oleh toko 3 putri gorden.



Gambar 4. 16 User Interface Halaman Utama

*User interface* produk

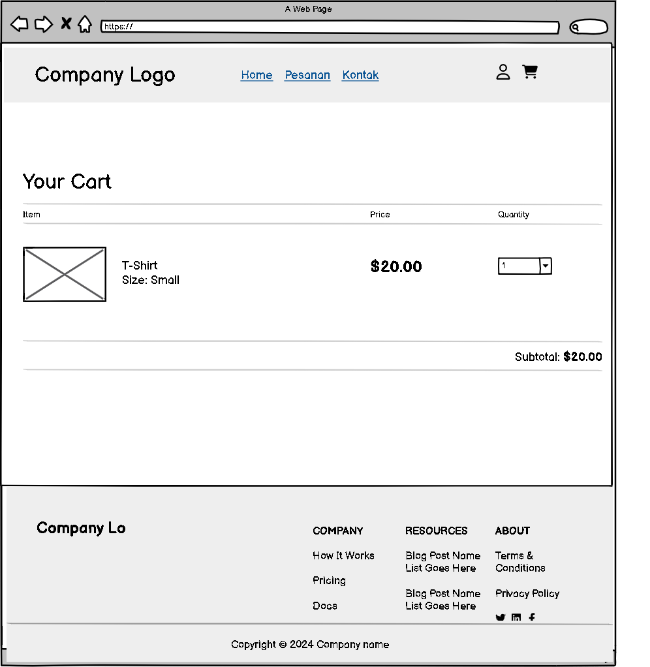
Pada *user interface* produk *customer* dapat melihat detail produk dengan cara mengklik produk tersebut, maka setelah di klik sistem langsung menampilkan detail produk yang menampilkan nama produk, keterangan, kategori, stok, harga, dan customer bisa mengklik checkout jika ingin belanja.



Gambar 4. 17 User Interface Produk

*User interface* pesanan

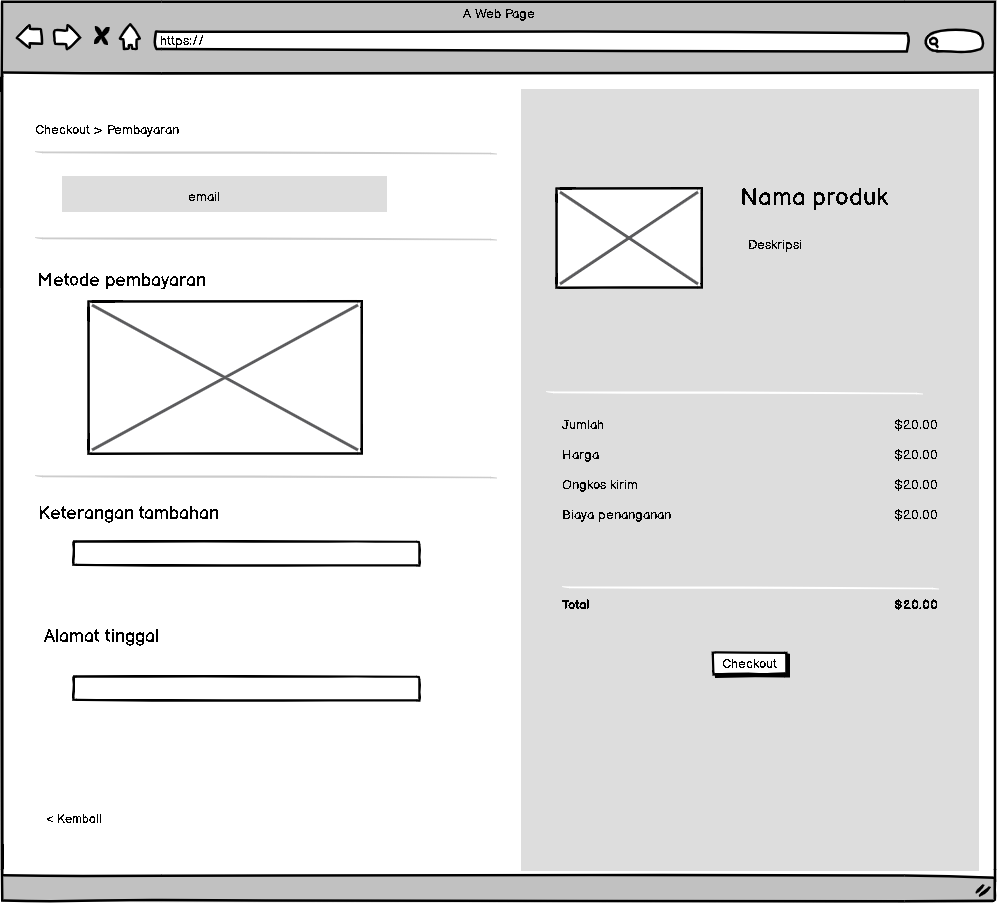
Pada *user interface* pesanan, customer bisa melihat detail harga barang yang sudah di klik checkout pada halaman produk sebelumnya.



Gambar 4. 18 User Interface Pesanan

*User interface* pembayaran

Pada *user interface* pembayaran, *custome*r bisa melihat detail pesanan seperti jumlah barang, total harga, biaya penanganan, ongkos kirim, tata cara pembayaran, serta customer bisa menambahkan alamat dikolom yang sudah tertera.



Gambar 4. 19 User Interface Pembayaran

1. *User interface* profil

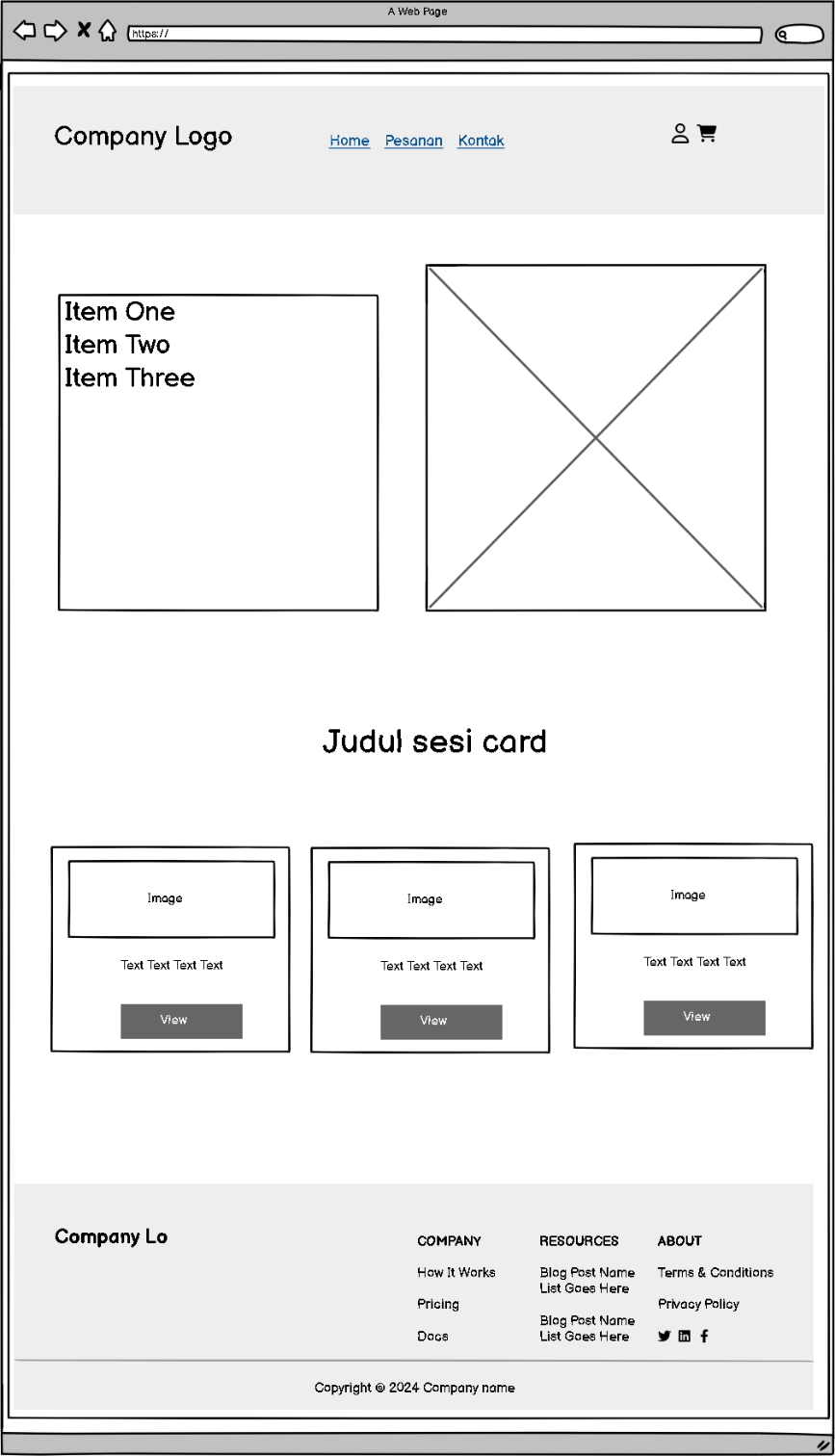
Pada *user interface* profil, terdapat kolom yang harus di isi seperti nama, email, alamat, contact, dan code zip, setelah semua informasi benar customer bisa klik submit.



Gambar 4. 20 User Interface Profil

1. *User interface contact*

Pada user interface contact, terdapat informasi tentang toko 3 putri gorden. Seperti alamat, nomer hp, instagram, dan facebook.



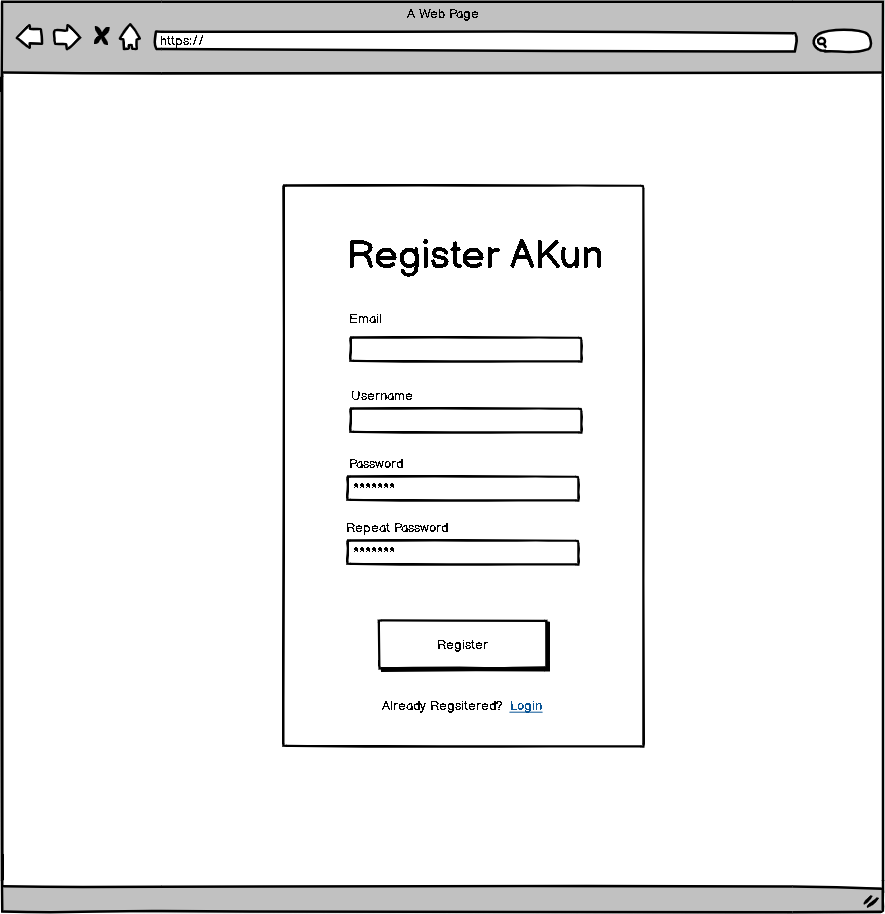
Gambar 4. 21 User Interface Contact

1. *User interface* login admin



Gambar 4. 22 User Interface Login Admin

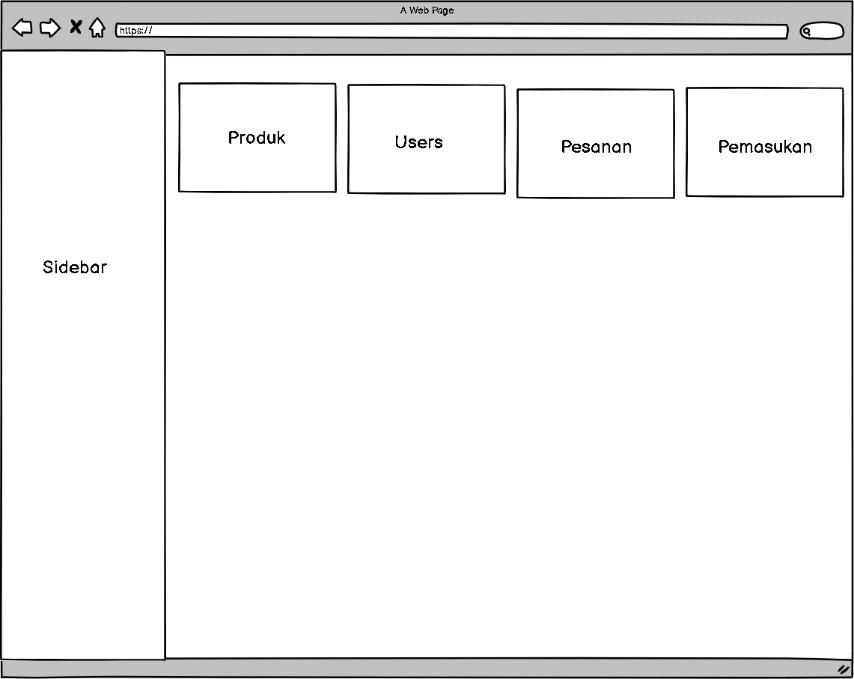
1. *User interface register* admin



Gambar 4. 23 User Interface Register Admin

1. *User interface* *home* admin

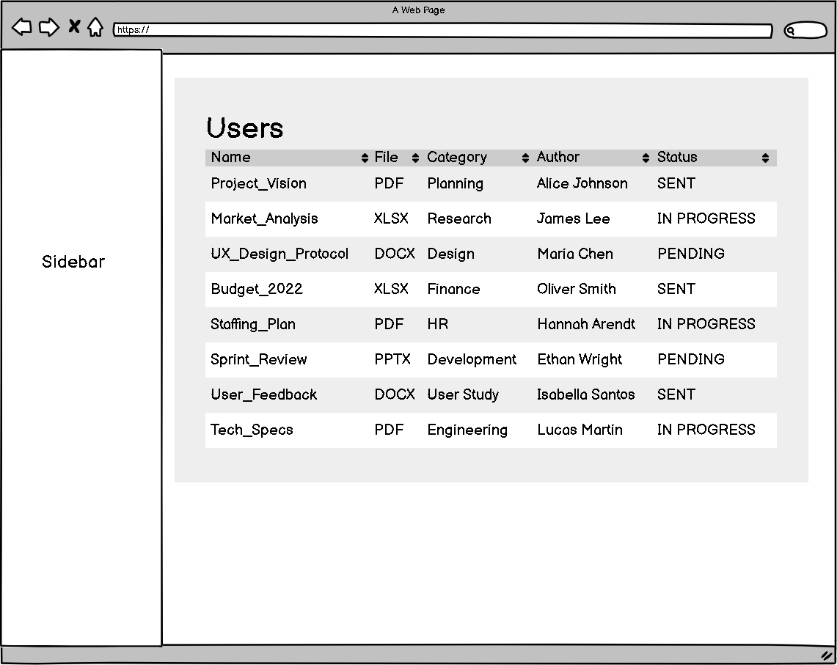
Pada *user interface home* admin, terdapat beberapa atribut. Diantaranya user, produk, pesanan, serta pendapatan dan informasi tentang pesanan.



Gambar 4. 24 User Interface Home Admin

1. *User interface* *user*

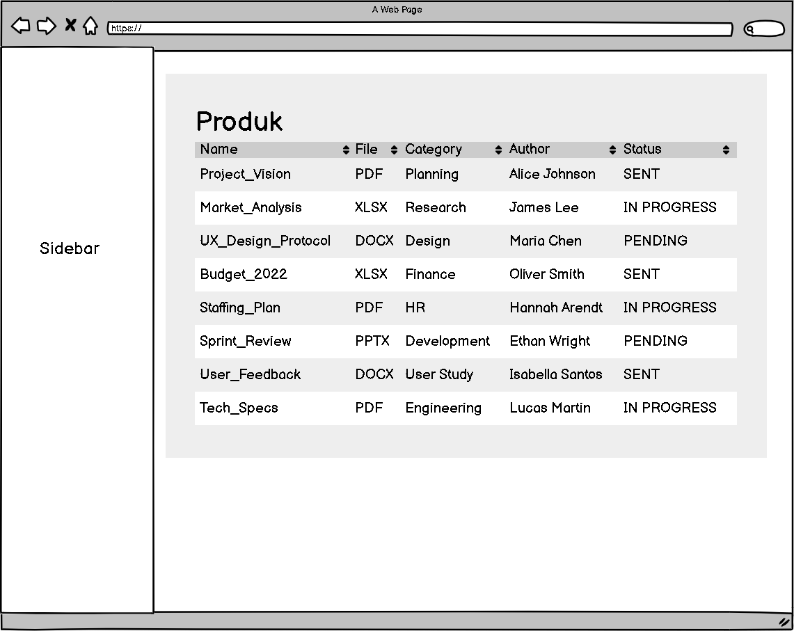
Pada *user interface* *user* admin, terdapat data pengguna akun yang telah mendaftar di website e-commerce 3 putri gorden.



Gambar 4. 25 User Interface User

1. *User interface* produk admin

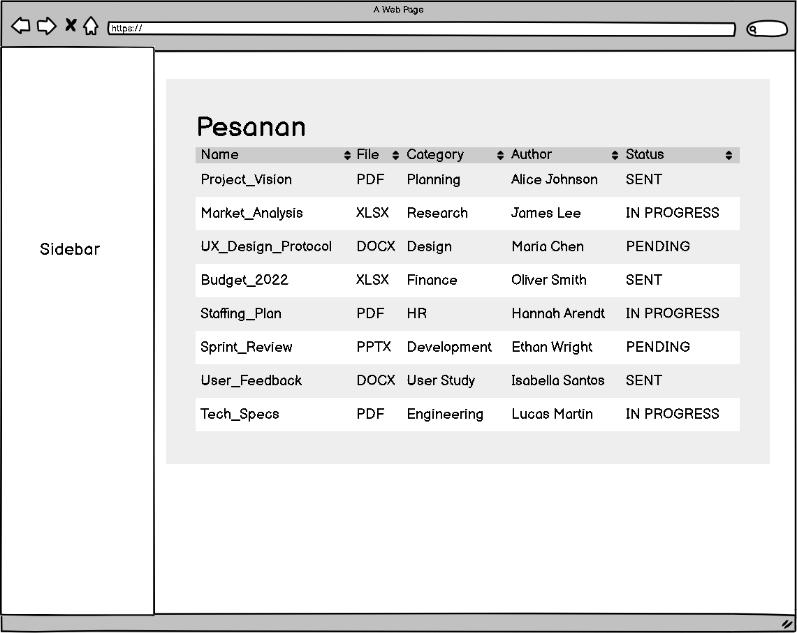
Pada *user interface* produk admin, admin dapat melakukan tambah barang, lalu melihat detail barang, ubah data barang, dan hapus data.



Gambar 4. 26 User Interface Produk Admin

1. *User interface* pesanan admin

Pada *user interface* pesanan admin, admin dapat melihat informasi pesanan yang sudah dilakukan *customer* dan admin pun bisa mengkonfirmasi pesanan tersebut.



Gambar 4. 27 User Interface Pesanan Admin

**4.2.4 Listing program**

*Listing* program merupakan susunan dari beberapa struktur data/*computer codes* yang digunakan dalam pengembangan sistem informasi terutama dalam mengatasi masalah yang sedang diteliti. Berikut *list* program dalam mengatasi masalah:

**ADMIN**

|  |
| --- |
| <?php  namespace App\Controllers;  use App\Controllers\BaseController;  use CodeIgniter\HTTP\ResponseInterface;  use Myth\Auth\Models\UserModel;  use App\Models\ProdukModel;  use App\Models\OrderModel;  class AdminPages extends BaseController  {      protected $usersModel, $db, $builder, $builderproduk, $builderorder, $produkModel, $orderModel;      protected $helpers = ['form'];      public function \_\_construct()      {          $this->usersModel = new UserModel();          $this->produkModel = new ProdukModel();          $this->orderModel = new OrderModel();          $this->db = \Config\Database::connect();          $this->builder = $this->db->table('users');          $this->builderproduk = $this->db->table('produk');          $this->builderorder = $this->db->table('orders');      }      public function index()      {          // Model          $totaluser = count($this->usersModel->findAll());          $totalorder = count($this->orderModel->findAll());          $totalproduk = count($this->produkModel->findAll());          $totalharga = 0;          $orders = $this->orderModel->findAll();          foreach ($orders as $order) {              $totalharga += (int)$order['total'];          }          $data = [              'title' => 'Ringkasan Perhitungan',              'navtitle' => 'home',              'totalusers' => $totaluser,              'totalorder' => $totalorder,              'totalproduk' => $totalproduk,              'totalharga' => $totalharga          ];          return view('dashboard/index', $data);      }      //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Users \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/      public function users()      {          $this->builder->select('users.id as userid, username,email,name,user\_image as img,created\_at');          $this->builder->join('auth\_groups\_users', 'auth\_groups\_users.user\_id = users.id');          $this->builder->join('auth\_groups', 'auth\_groups.id = auth\_groups\_users.group\_id');          $query = $this->builder->get();          $data = [              'title' => 'Users',              'navtitle' => 'users',              'users' => $query->getResult()          ];          return view('dashboard/users', $data);      }      public function userdetail($id = 0)      {          $data = [              'title' => 'Detail Users',              'navtitle' => 'users',              'users' =>  $this->usersModel->usersAuthRelations($id)->getRow()          ];          if (empty($data['users'])) {              return redirect()->to('/dashboard/users');          }          return view('dashboard/users/detail', $data);      }      public function usersupdate($id)      {          $data = [              'title' => 'Edit User',              'navtitle' => 'users',              'users' => $this->usersModel->usersAuthRelations($id)->getRow()          ];          return view('dashboard/users/detail', $data);      }      //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Produk \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/      public function produk()      {          $this->builderproduk->select('produk.\*,produk.id as prod\_id ,kategori.jenis ');          $this->builderproduk->join('kategori', 'produk.kategori\_id = kategori.id');          $query =  $this->builderproduk->get();          $data = [              'title' => 'Produk',              'navtitle' => 'produk',              'produk' => $query->getResult()          ];          return view('dashboard/produk', $data);      }      public function produkadd()      {          $query = $this->db->table('kategori')->get()->getResult();          $data = [              'title' => 'Tambah Produk',              'navtitle' => 'produk',              'kategori' =>  $query          ];          return view('dashboard/produk/add', $data);      }      public function produkdetail($id)      {          $query = $this->db->table('kategori')->get()->getResult();          $data = [              'title' => 'Detail Produk',              'navtitle' => 'produk',              'kategori' =>  $query,              'produk' =>  $this->produkModel->produkRelations($id)->getRow()          ];          return view('dashboard/produk/detail', $data);      }      public function produkupdate($id)      {          $query = $this->db->table('kategori')->get()->getResult();          $data = [              'title' => 'Edit Produk',              'navtitle' => 'produk',              'kategori' =>  $query,              'produk' =>  $this->produkModel->produkRelations($id)->getRow()          ];          return view('dashboard/produk/detail', $data);      }      //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Pesanan \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/      public function pesanan()      {          $data = [              'title' => 'Pesanan',              'navtitle' => 'pesanan',              'produkorder' => $this->orderModel->findAll()          ];          return view('dashboard/pesanan', $data);      }      public function pesananupdate()      {          $data = [              'title' => 'Edit Pesanan',              'navtitle' => 'pesanan',              'orders' => $this->orderModel->findAll()          ];          return view('dashboard/pesanan/detail', $data);      }  } |

**BASE CONTROLLER**

|  |
| --- |
| <?php  namespace App\Controllers;  use CodeIgniter\Controller;  use CodeIgniter\HTTP\CLIRequest;  use CodeIgniter\HTTP\IncomingRequest;  use CodeIgniter\HTTP\RequestInterface;  use CodeIgniter\HTTP\ResponseInterface;  use Psr\Log\LoggerInterface;  /\*\*   \* Class BaseController   \*   \* BaseController provides a convenient place for loading components   \* and performing functions that are needed by all your controllers.   \* Extend this class in any new controllers:   \*     class Home extends BaseController   \*   \* For security be sure to declare any new methods as protected or private.   \*/  abstract class BaseController extends Controller  {      /\*\*       \* Instance of the main Request object.       \*       \* @var CLIRequest|IncomingRequest       \*/      protected $request;      /\*\*       \* An array of helpers to be loaded automatically upon       \* class instantiation. These helpers will be available       \* to all other controllers that extend BaseController.       \*       \* @var list<string>       \*/      protected $helpers = ['auth'];      /\*\*       \* Be sure to declare properties for any property fetch you initialized.       \* The creation of dynamic property is deprecated in PHP 8.2.       \*/      // protected $session;      /\*\*       \* @return void       \*/      public function initController(RequestInterface $request, ResponseInterface $response, LoggerInterface $logger)      {          // Do Not Edit This Line          parent::initController($request, $response, $logger);          // Preload any models, libraries, etc, here.          // E.g.: $this->session = \Config\Services::session();      }  } |

**PAGES**

|  |
| --- |
| <?php  namespace App\Controllers;  use App\Controllers\BaseController;  use CodeIgniter\HTTP\ResponseInterface;  use Myth\Auth\Models\UserModel;  use App\Models\ProdukModel;  use App\Models\OrderModel;  class Pages extends BaseController  {        protected $usersModel, $db, $builder, $builderproduk, $builderorder, $produkModel, $orderModel;      protected $helpers = ['form'];      public function \_\_construct()      {          $this->usersModel = new UserModel();          $this->produkModel = new ProdukModel();          $this->orderModel = new OrderModel();          $this->db = \Config\Database::connect();          $this->builder = $this->db->table('users');          $this->builderproduk = $this->db->table('produk');          $this->builderorder = $this->db->table('orders');      }      public function index()      {          $currenPage = $this->request->getVar('page\_produk') ? $this->request->getVar('page\_produk') : 1;          $keyword = $this->request->getVar('keyword');          if ($keyword) {              if ($keyword == 'harga\_tinggi') {                  $produk = $this->produkModel->orderBy('harga', 'DESC');              } elseif ($keyword == 'harga\_rendah') {                  $produk = $this->produkModel->orderBy('harga', 'ASC');              } else {                  $produk = $this->produkModel->search($keyword);              }          } else {              $produk = $this->produkModel;          }          $data = [              "produk" => $produk->paginate(8, 'produk'),              "pager" => $this->produkModel->pager,              "currenPage" => $currenPage,              "navtitle" => "home"          ];          return view('pages/index', $data);      }      public function produk($id)      {          //          $data = [              "produk" => $this->produkModel->produkRelations($id)->getRow(),              "navtitle" => "home"          ];          return view('pages/produk', $data);      }      public function pembayaran($id)      {          $produk = $this->produkModel->produkRelations($id)->getRow();          $admin = 6000;          $jumlah = $this->request->getPost('jumlah') ?? session()->get('jumlah');          $total\_biaya = ($produk->harga \* $jumlah) + $admin;          $users = $this->usersModel->usersAuthRelations(user()->id)->getRow();          session()->set('jumlah', $jumlah);          $data = [              "produk" => $produk,              "jumlah" => $jumlah,              "admin" => $admin,              "navtitle" => "home",              "total\_biaya" =>  $total\_biaya,              "users" => $users          ];          return view('pages/produk/pembayaran', $data);      }      public function profil($id)      {          $data = [              'title' => 'Edit User',              'navtitle' => 'profil',              'users' => $this->usersModel->usersAuthRelations($id)->getRow()          ];          return view('pages/profil', $data);      }      public function pesanan()      {          //          $orderData = $this->orderModel->where('user\_id', user()->id)->findAll();          $produkIds = array\_column($orderData, 'produk\_id');          $produkData = $produkIds ? $this->produkModel->whereIn('id', $produkIds)->findAll() : [];          $imgproduk = $produkData ? array\_column($produkData, 'gambar') : [];          $data = [              'title' => 'Pesanan',              "orders" => array\_map(function ($order, $img) {                  $order['gambar'] = $img;                  return $order;              }, $orderData, $imgproduk),              "navtitle" => "pesanan"          ];          return view('pages/pesanan', $data);      }      public function kontak()      {          //          $data = [              "navtitle" => "kontak"          ];          return view('pages/kontak', $data);      }  } |

**PESANAN**

|  |
| --- |
| <?php  namespace App\Controllers;  use App\Controllers\BaseController;  use CodeIgniter\HTTP\ResponseInterface;  use App\Models\OrderModel;  use Myth\Auth\Models\UserModel;  class Pesanan extends BaseController  {      protected $usersModel, $db, $builderorder, $orderModel;      protected $helpers = ['form'];      public function \_\_construct()      {          $this->usersModel = new UserModel();          $this->orderModel = new OrderModel();      }      public function save()      {          $rules = [              'produk\_id' => [                  'rules' => 'required',                  'errors' => [                      'required' => '{field} harus diisi'                  ],              ],              'biaya\_admin' => [                  'rules' => 'required|numeric',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus angka'                  ],              ],              'jumlah\_barang' => [                  'rules' => 'required|numeric',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus angka'                  ],              ],              'total' => [                  'rules' => 'required|numeric',                  'errors' => [                      'required' => '{field} harus diisi silakan tambahkan alamat baru',                      'numeric' => '{field} harus angka'                  ],              ],              'text\_opsi' => [                  'rules' => 'max\_length[100]',                  'errors' => [                      'max\_length' => '{field} tulisan melebihi batas'                  ],              ],              'alamat\_baru' => [                  'rules' => 'required|max\_length[200]',                  'errors' => [                      'required' => 'Alamat harus diisi',                      'max\_length' => '{field} tulisan melebihi batas'                  ],              ],          ];          if (!$this->validate($rules)) {              return redirect()->to('/produk/pembayaran/' . $this->request->getVar("produk\_id"))->withInput()->with('errors', $this->validator->getErrors());          }          $user\_id = user()->id;          if (!user()->address) {              $this->usersModel->save([                  "id" => $user\_id,                  "address" => $this->request->getVar("alamat\_baru")              ]);          }          $this->orderModel->save([              "produk\_id" => $this->request->getVar("produk\_id"),              "user\_id" => $user\_id,              "biaya\_admin" => $this->request->getVar("biaya\_admin"),              "jumlah\_barang" => $this->request->getVar("jumlah\_barang"),              "total" => $this->request->getVar("total"),              "text\_opsi" => $this->request->getVar("text\_opsi"),          ]);          $no\_wa = getenv('NO\_WA');          $message = 'Produk berhasil di checkout :' . "\n" .              'Id User: ' . user()->id . "\n" .              'Id Produk: ' . $this->request->getVar("produk\_id") . "\n" .              'Jumlah Barang: ' . $this->request->getVar("jumlah\_barang") . "\n" .              'Biaya Admin: ' . $this->request->getVar("biaya\_admin") . "\n" .              'Total: ' . $this->request->getVar("total") . "\n" .              'Keterangan: ' . $this->request->getVar("text\_opsi");          $encoded\_message = urlencode($message);          $whatsapp\_link = 'https://api.whatsapp.com/send?phone=' . $no\_wa . '&text=' . $encoded\_message;          return redirect()->to($whatsapp\_link);      }      public function update($id)      {          $rules = [              'no\_resi' => [                  'rules' => 'required',                  'errors' => [                      'required' => '{field} harus diisi'                  ],              ],              'ongkos' => [                  'rules' => 'required',                  'errors' => [                      'required' => '{field} harus diisi'                  ],              ]          ];          if (!$this->validate($rules)) {              return redirect()->to('/dashboard/pesanan/update/' . $id)->withInput()->with('errors', $this->validator->getErrors());          }          $this->orderModel->save([              "id" => $id,              "no\_resi" => $this->request->getVar("no\_resi"),              "ongkos" => $this->request->getVar("ongkos")          ]);          session()->setFlashdata('pesan', 'Data order berhasil diubah');          return redirect()->to('/dashboard/pesanan');      }      public function konfirmasi($id)      {          //          $rules = [              'konfirmasi' => [                  'rules' => 'required|in\_list[y,n]',                  'errors' => [                      'required' => '{field} harus diisi',                      'in\_list' => '{field} harus memilih "y" atau "n"'                  ],              ],          ];          if (!$this->validate($rules)) {              return redirect()->to('/dashboard/pesanan')->withInput()->with('errors', $this->validator->getErrors());          }          // Ubah data berdasarkan id          $this->orderModel->update($id, [              'konfirmasi' => $this->request->getVar("konfirmasi"),          ]);          session()->setFlashdata('pesan', 'Pesanan produk berhasil dikonfirmasi');          return redirect()->to('/dashboard/pesanan');      }      public function delete($id)      {          $this->orderModel->delete($id);          session()->setFlashdata('pesan', 'Pesanan produk berhasil dihapus');          return redirect()->to('/dashboard/pesanan');      }  } |

**PRODUK**

|  |
| --- |
| <?php  namespace App\Controllers;  use App\Controllers\BaseController;  use App\Models\ProdukModel;  class Produk extends BaseController  {      protected $produkModel;      protected $helpers = ['form'];      function \_\_construct()      {          $this->produkModel = new ProdukModel();      }      protected function getRules()      {          return [              'nama' => [                  'rules' => 'required|max\_length[100]',                  'errors' => [                      'required' => '{field} harus diisi',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'harga' => [                  'rules' => 'required|numeric',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus angka'                  ],              ],              'deskripsi' => [                  'rules' => 'required|max\_length[200]',                  'errors' => [                      'required' => '{field} harus diisi',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'stok' => [                  'rules' => 'required|numeric',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus angka'                  ],              ],              'ukuran' => [                  'rules' => 'required|max\_length[30]',                  'errors' => [                      'required' => '{field} harus diisi',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'jenis' => [                  'rules' => 'required',                  'errors' => [                      'required' => '{field} harus diisi'                  ],              ],              'berat' => [                  'rules' => 'required|numeric',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus angka'                  ],              ],              'gambar' => [                  'rules' => 'max\_size[gambar,1024]|is\_image[gambar]|mime\_in[gambar,image/jpg,image/jpeg,image/png]',                  'errors' => [                      'max\_size' => 'ukuran gambar terlalu besar',                      'is\_image' => 'Yang anda pilih bukan gambar',                      'mime\_in' => 'Yang anda pilih bukan gambar'                  ],              ],          ];      }      public function save()      {          $rules = $this->getRules();          if (!$this->validate($rules)) {              return redirect()->to('/dashboard/produk/add')->withInput()->with('errors', $this->validator->getErrors());          }          //\*\*\*\* file gambar \*/          // File          $fileImg = $this->request->getFile('gambar');          //cek fileImg          if (!$fileImg | $fileImg->getError() == 4) {              $namaSampul = 'produk.png';          } else {              // generate random              $namaSampul = $fileImg->getRandomName();              // Pindahkan File ke folder img/produk              $fileImg->move('img/produk/', $namaSampul);              // hapus file sebelumnya          }          // Simpan data ke database          $this->produkModel->save([              'nama' => $this->request->getVar("nama"),              'harga' => $this->request->getVar("harga"),              'deskripsi' => $this->request->getVar("deskripsi"),              'stok' => $this->request->getVar("stok"),              'ukuran' => $this->request->getVar("ukuran"),              'berat' => $this->request->getVar("berat"),              'kategori\_id' => $this->request->getVar('jenis'),              'gambar' => $namaSampul          ]);          session()->setFlashdata('pesan', 'Data produk berhasil ditambahkan');          return redirect()->to('/dashboard/produk');      }      public function update($id)      {          $rules = $this->getRules();          if (!$this->validate($rules)) {              return redirect()->to('/dashboard/produk/add')->withInput()->with('errors', $this->validator->getErrors());          }          //\*\*\*\* file gambar \*/          // File          $fileImg = $this->request->getFile('gambar');          //cek fileImg          if (!$fileImg | $fileImg->getError() == 4) {              $namaSampul = $this->request->getVar('gambarLama');          } else {              // generate random              $namaSampul = $fileImg->getRandomName();              // Pindahkan File ke folder img/produk              $fileImg->move('img/produk/', $namaSampul);              // hapus file sebelumnya              if ($this->request->getVar('gambarLama')) {                  // hapus file gambar                  $produkImg = $this->produkModel->find($id);                  // hapus file gambar                  if ($produkImg['gambar'] != 'produk.png') {                      if (file\_exists('img/produk/' . $produkImg['gambar'])) {                          unlink('img/produk/' . $this->request->getVar('gambarLama'));                      }                  }              }          }          // Simpan data ke database          $this->produkModel->save([              'id' => $id,              'nama' => $this->request->getVar("nama"),              'harga' => $this->request->getVar("harga"),              'deskripsi' => $this->request->getVar("deskripsi"),              'stok' => $this->request->getVar("stok"),              'ukuran' => $this->request->getVar("ukuran"),              'berat' => $this->request->getVar("berat"),              'kategori\_id' => $this->request->getVar('jenis'),              'gambar' => $namaSampul          ]);          session()->setFlashdata('pesan', 'Data produk berhasil diubah');          return redirect()->to('/dashboard/produk');      }      public function delete($id)      {          // hapus file gambar          $produkImg = $this->produkModel->find($id);          // hapus file gambar          if ($produkImg['gambar'] != 'produk.png') {              if (file\_exists('img/produk/' . $produkImg['gambar'])) {                  unlink('img/produk/' . $produkImg['gambar']);              }          }          $this->produkModel->delete($id);          session()->setFlashdata('pesan', 'Data produk berhasil dihapus');          return redirect()->to('/dashboard/produk');      }  } |

**USER**

|  |
| --- |
| <?php  namespace App\Controllers;  use App\Controllers\BaseController;  use Myth\Auth\Models\UserModel;  use Myth\Auth\Models\GroupModel;  use Myth\Auth\Models\PermissionModel;  class User extends BaseController  {      protected $usersModel, $groupModel, $permissionModel, $db, $builder;      protected $helpers = ['form'];      protected $session;      public function \_\_construct()      {          $this->usersModel = new UserModel();          $this->groupModel = new GroupModel();          $this->permissionModel = new PermissionModel();          $this->db = \Config\Database::connect();          $this->builder = $this->db->table('users');          $this->session = service('session');      }      public function update($id)      {          //\*\*\*\*\*\*\*\*\* Validation \*/          // cek judul          $rules = [              'fullname' => [                  'rules' => 'required|max\_length[200]',                  'errors' => [                      'required' => '{field} harus diisi',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'address' => [                  'rules' => 'required|max\_length[200]',                  'errors' => [                      'required' => '{field} harus diisi',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'no\_hp' => [                  'rules' => 'required|numeric|max\_length[15]',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus berupa angka',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'country' => [                  'rules' => 'required|max\_length[200]',                  'errors' => [                      'required' => '{field} harus diisi',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'kode\_zip' => [                  'rules' => 'required|numeric|max\_length[10]',                  'errors' => [                      'required' => '{field} harus diisi',                      'numeric' => '{field} harus berupa angka',                      'max\_length' => '{field} terlalu panjang'                  ],              ],              'img' => [                  'rules' => 'max\_size[img,1024]|is\_image[img]|mime\_in[img,image/jpg,image/jpeg,image/png]',                  'errors' => [                      'max\_size' => 'ukuran gambar terlalu besar',                      'is\_image' => 'Yang anda pilih bukan gambar',                      'mime\_in' => 'Yang anda pilih bukan gambar'                  ],              ],          ];          if (!$this->validate($rules)) {              return redirect()->to('/dashboard/users/detail/' . $id)->withInput()->with('errors', $this->validator->getErrors());          }          //\*\*\*\* file sampul \*/          // File          $fileImg = $this->request->getFile('img');          //cek fileImg          if ($fileImg->getError() == 4) {              $namaSampul = $this->request->getVar('profilLama');          } else {              // generate random              $namaSampul = $fileImg->getRandomName();              // Pindahkan File ke folder img/profile              $fileImg->move('img/profile/', $namaSampul);              // hapus file sebelumnya              if ($this->request->getVar('profilLama')) {                  unlink('img/profile/' . $this->request->getVar('profilLama'));              }          }          $this->usersModel->save([              'id' => $id,              'fullname' => $this->request->getVar("fullname"),              'address' => $this->request->getVar("address"),              'no\_hp' => $this->request->getVar("no\_hp"),              'country' => $this->request->getVar("country"),              'kode\_zip' => $this->request->getVar("kode\_zip"),              'country' => $this->request->getVar("country"),              'user\_image' => $namaSampul          ]);          session()->setFlashdata('pesan', 'Data users berhasil diubah');          return redirect()->back();      }      public function delete($id)      {          // Temukan pengguna berdasarkan ID          $users = $this->usersModel->find($id);          $groupAuth =  $this->db->table('auth\_groups\_users')->where('user\_id', $id)->get()->getResult();          // Hapus file gambar jika bukan gambar default          if ($users->user\_image != 'default.jpg') {              $filePath = 'img/profile/' . $users->user\_image;              if (is\_file($filePath)) {                  unlink($filePath);              }          }          // Hapus entri di tabel auth\_groups\_users          $this->db->table('auth\_groups\_users')->where('user\_id', $id)->delete();          // Hapus secara paska          $this->db->table('auth\_groups\_permissions')->where('group\_id', $groupAuth[0]->group\_id)->delete();          $this->db->table('users')->where('id', $id)->delete();          // Set pesan flashdata untuk notifikasi sukses          session()->setFlashdata('pesan', 'Data users berhasil dihapus');          // Redirect kembali ke halaman sebelumnya          return redirect()->back();      }  } |

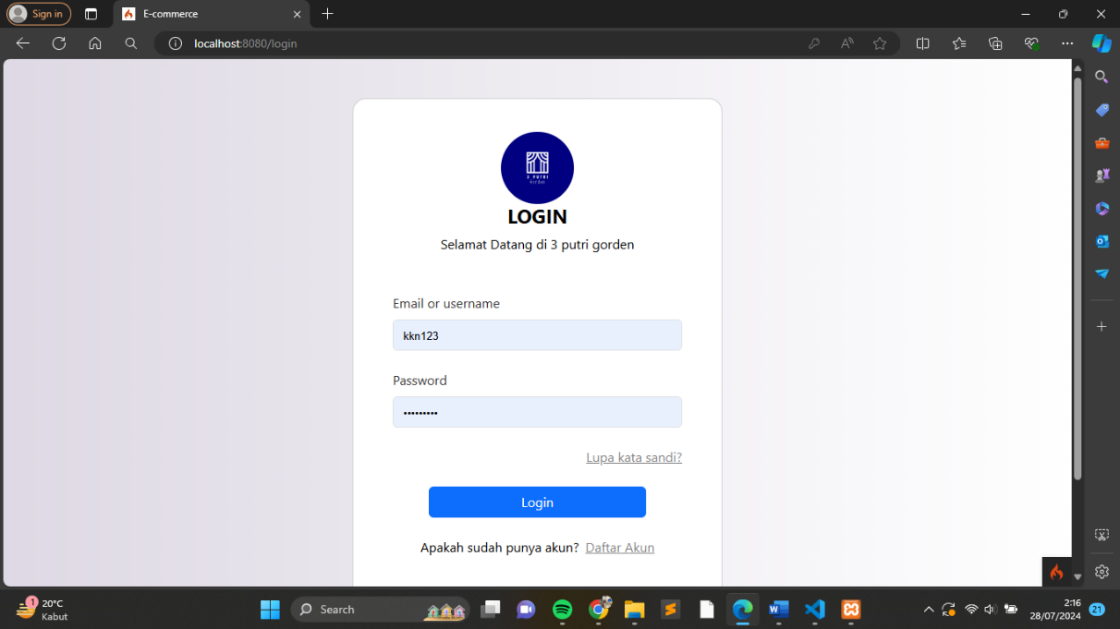
# BAB V

# IMPLEMENTASI DAN PENGUJIAN

## Hasil Tampilan

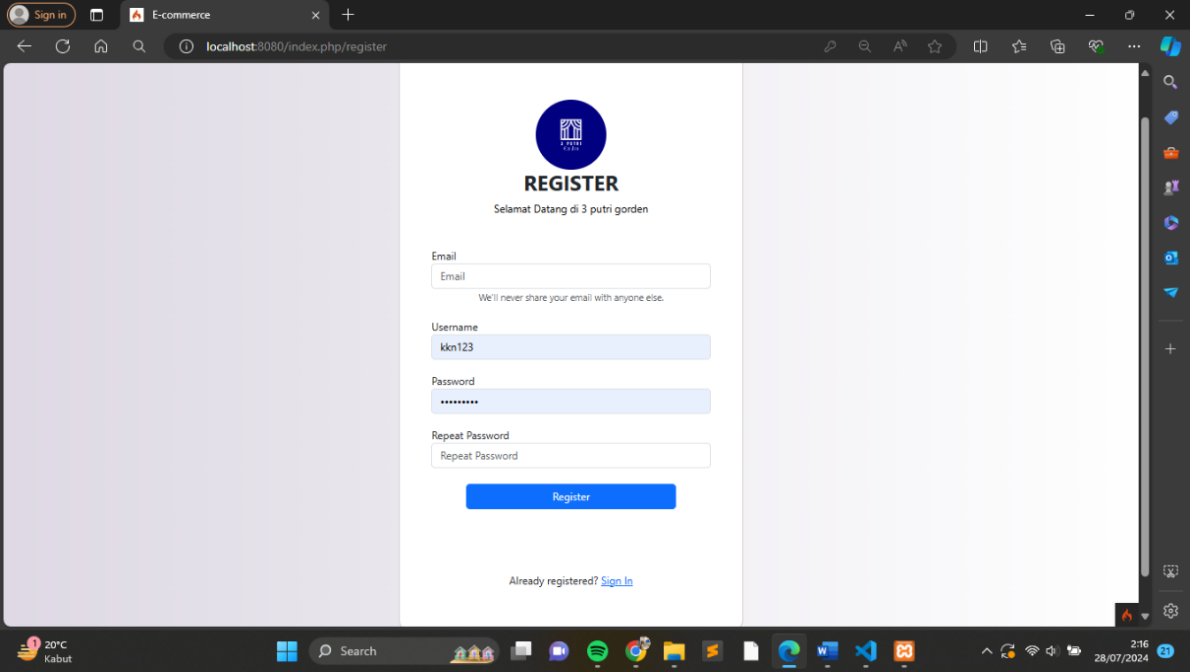
Setelah melakukan perancangan website, maka tahap selanjutnya adalah implementasi. Implementasi merupakan tahap penerapan bagi sistem baru dan merupakan tahap dimana website siap digunakan. Implementasi bertujuan untuk menjelaskan modul-modul perancangan.

Tampilan login admin



Gambar 5. 1 Tampilan Login Admin

Tampilan register admin



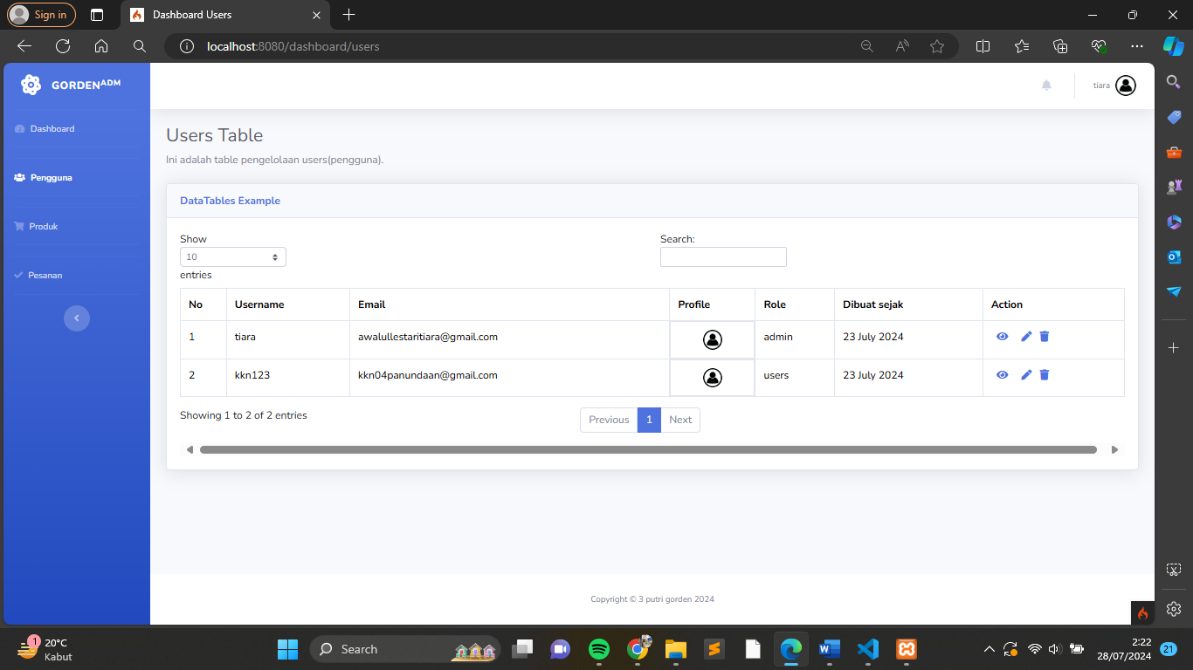
Gambar 5. 2 Tampilan Register Admin

Tampilan Dashboard admin



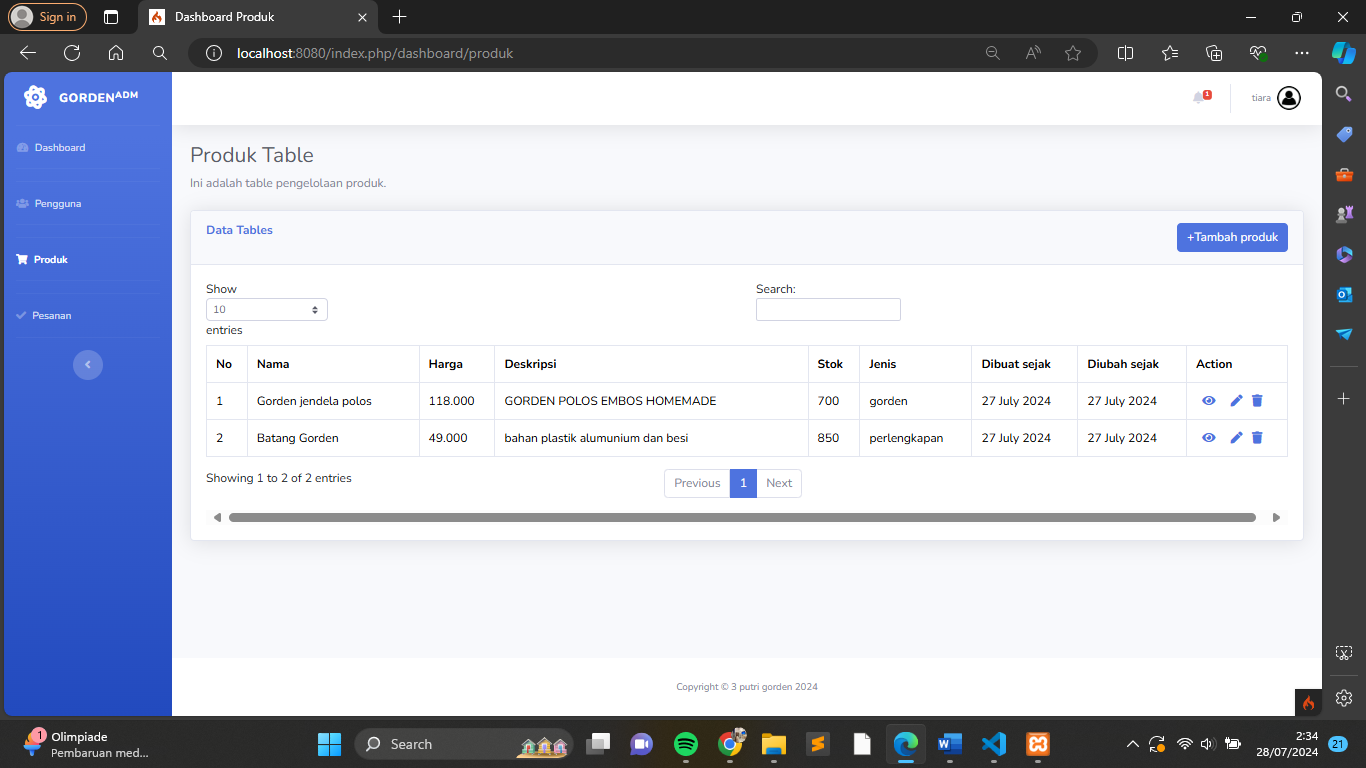
Gambar 5. 3 Tampilan Dashboard Admin

Tampilan user admin



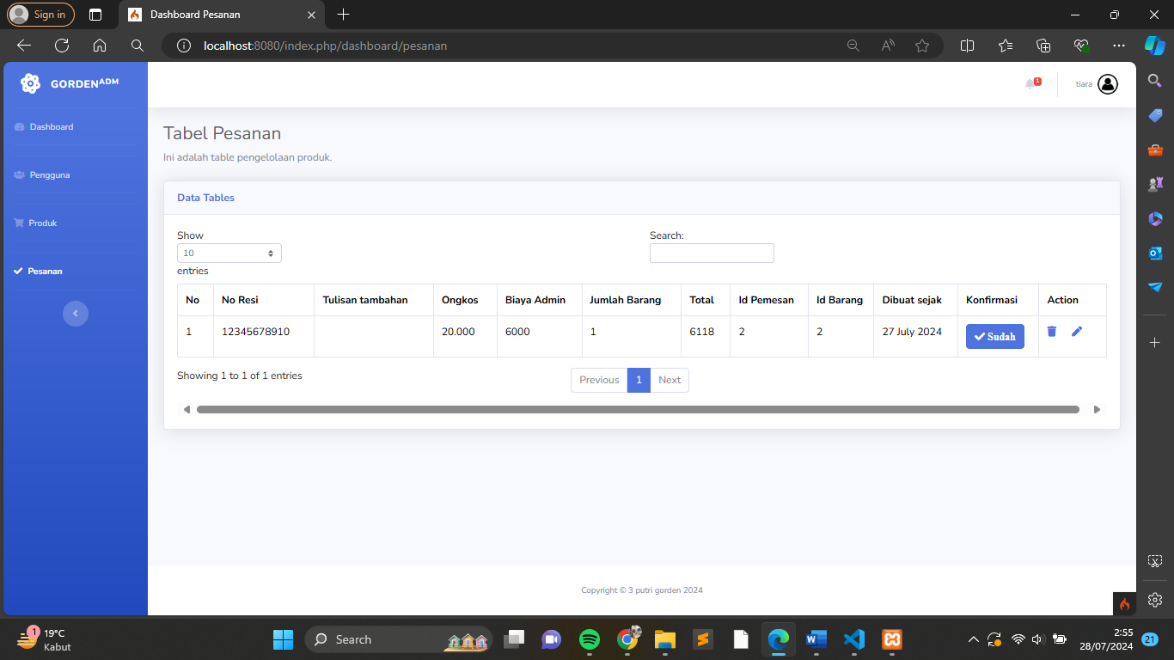
Gambar 5. 4 Tampilan User Admin

Tampilan produk admin



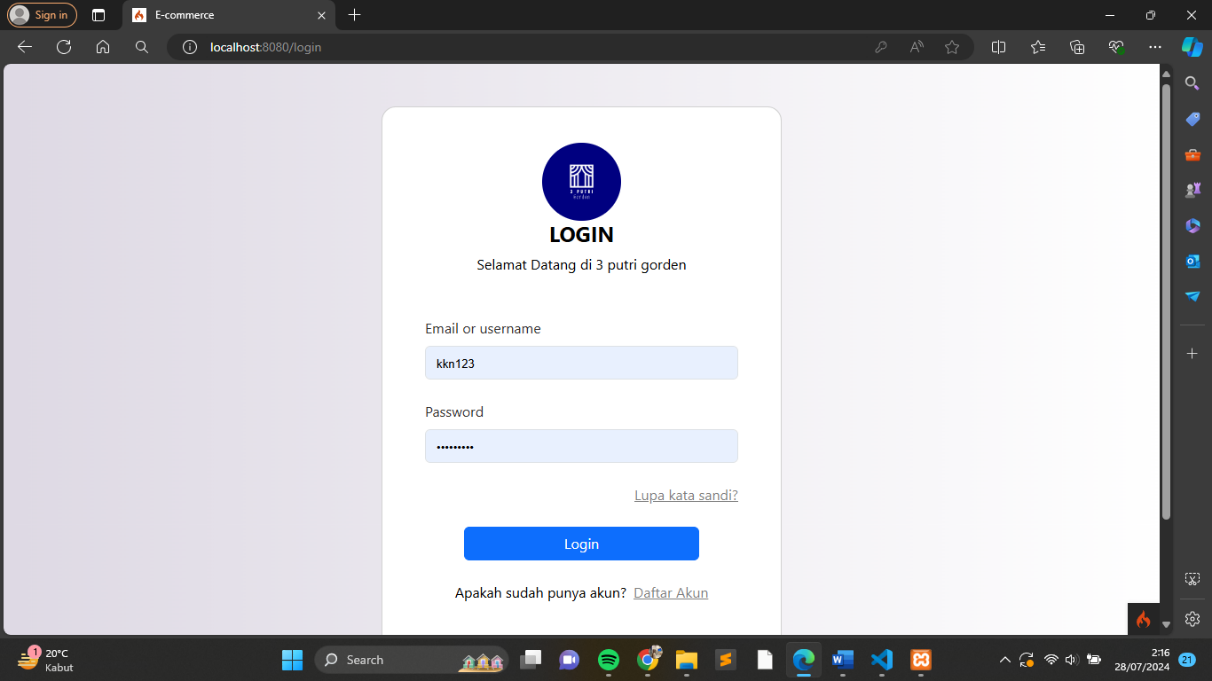
Gambar 5. 5 Tampilan Produk Admin

Tampilan pesanan admin



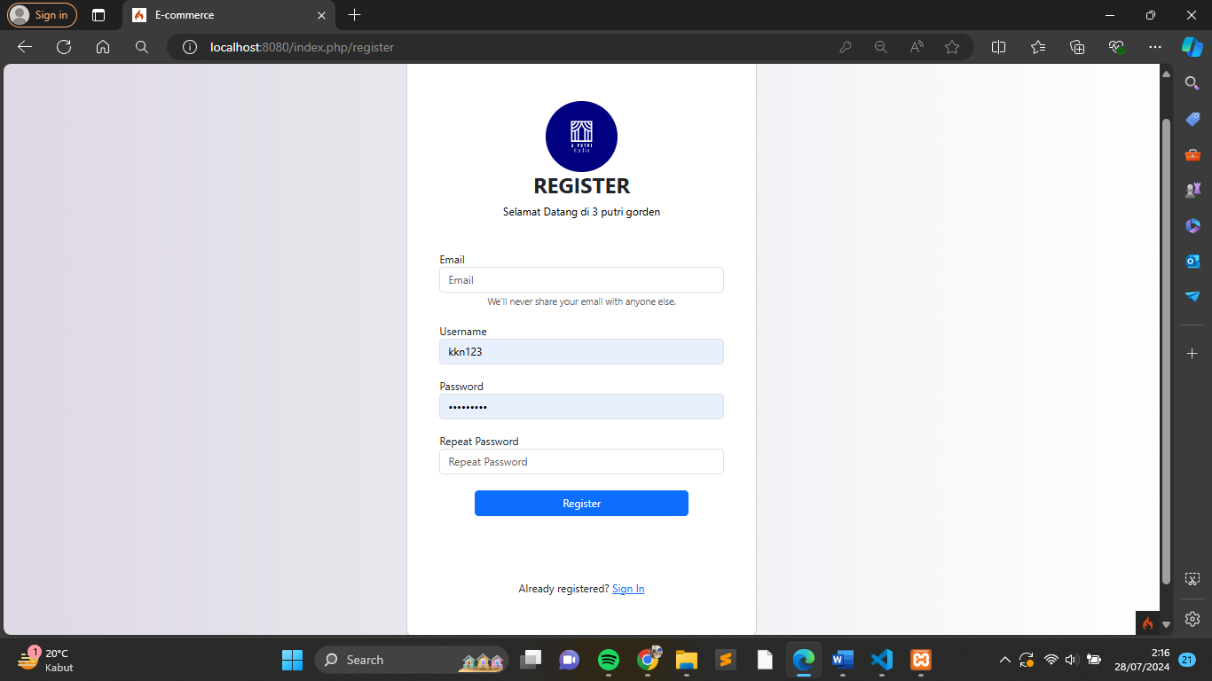
Gambar 5. 6 Tampilan Pesanan Admin

Tampilan login customer



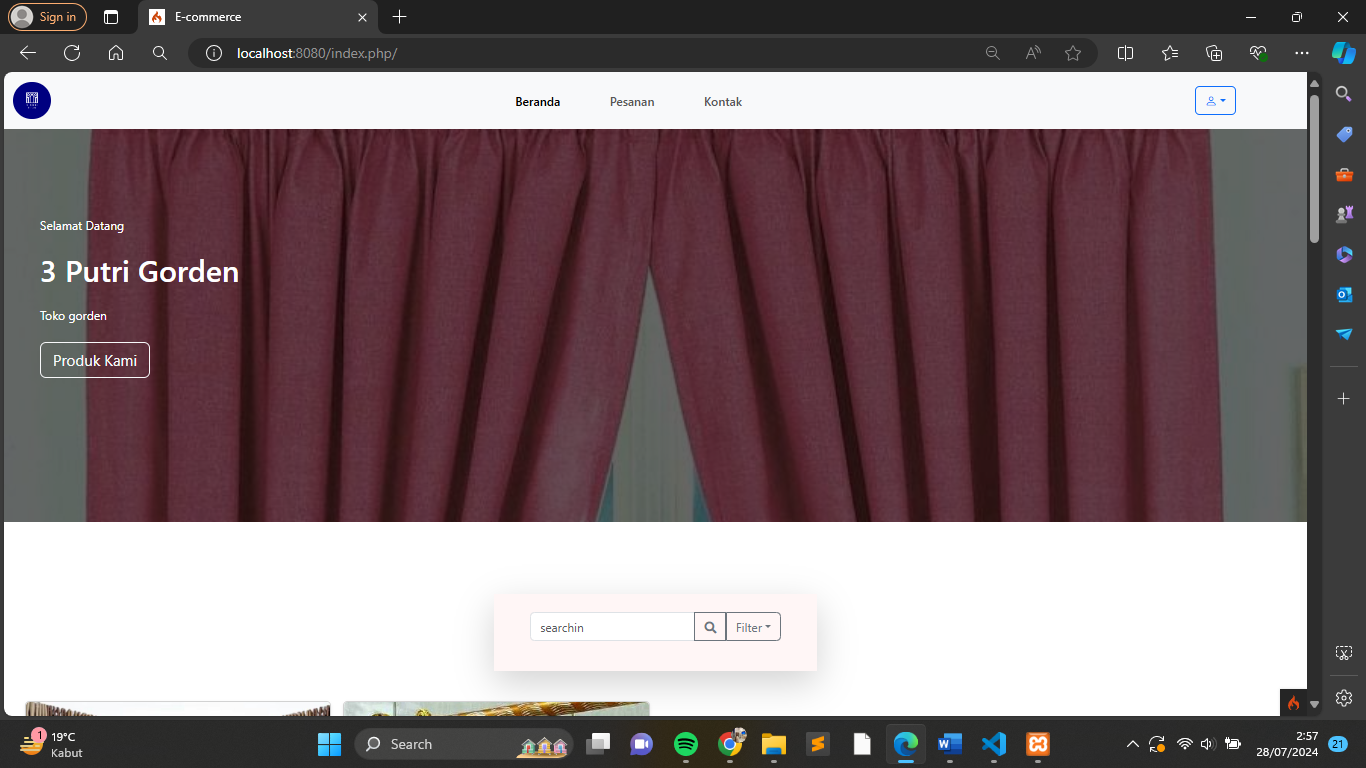
Gambar 5. 7 Tampilan Login Customer

Tampilan registrasi customer

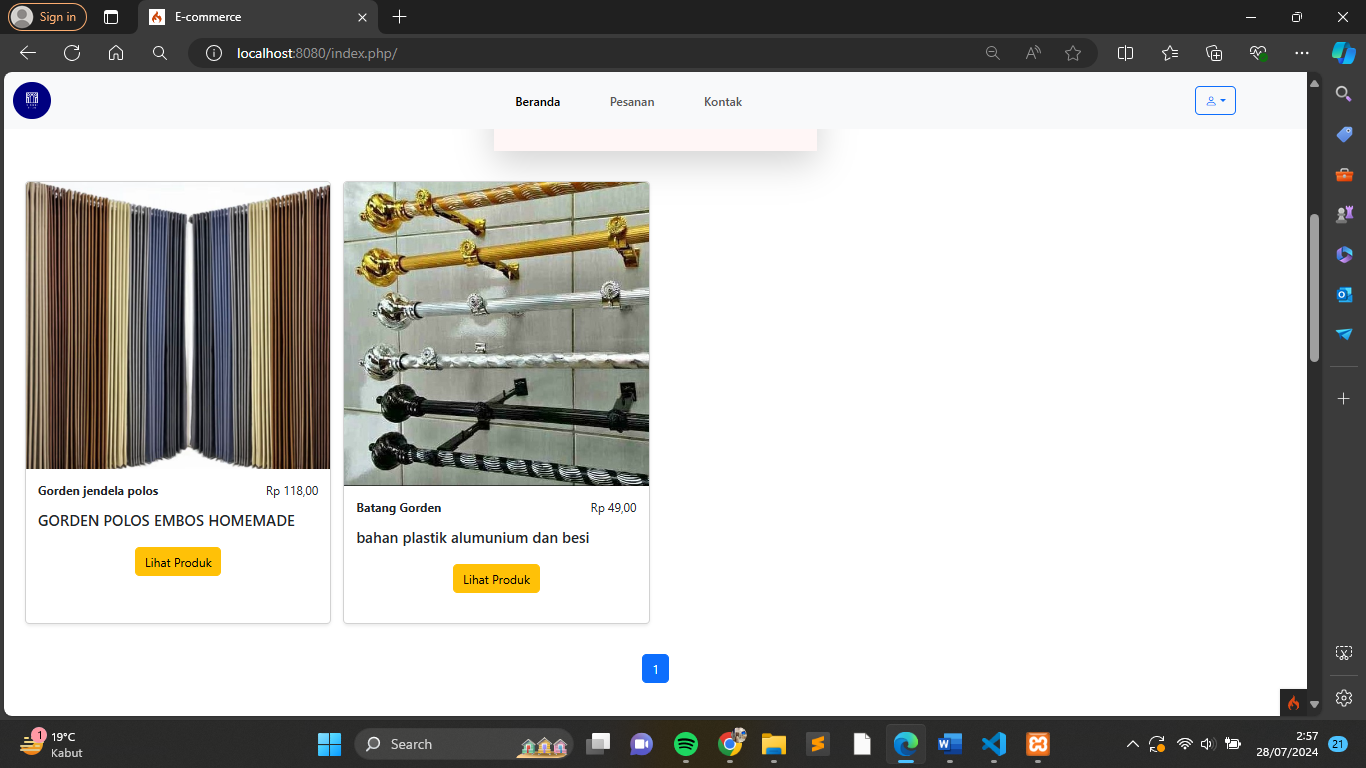


Gambar 5. 8 Tampilan Registrasi Customer

Tampilan beranda customer

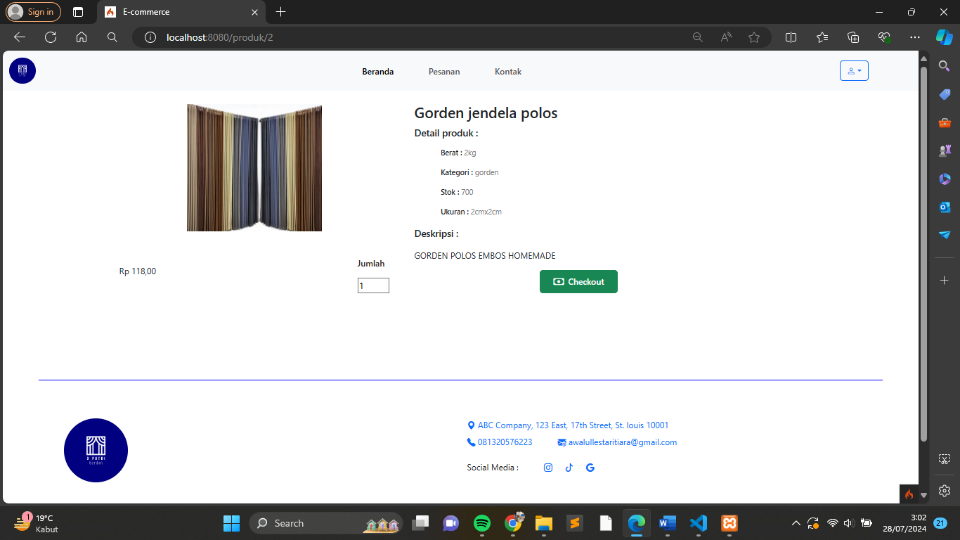


Gambar 5. 9 Tampilan Beranda Customer



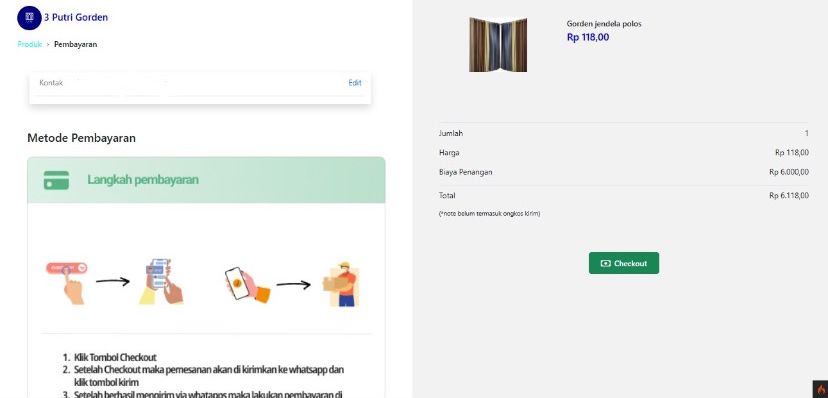
Gambar 5. 10 Tampilan Beranda Customer

Tampilan produk



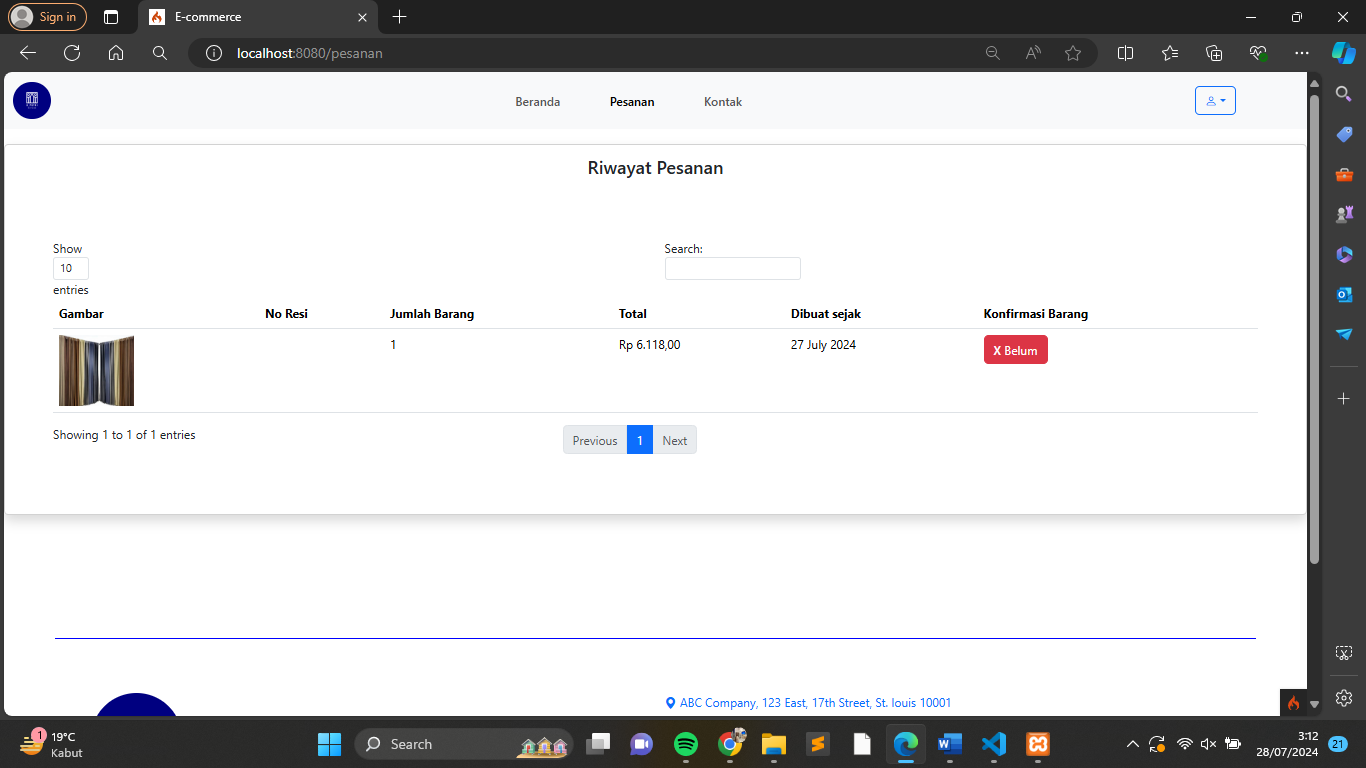
Gambar 5. 11 Tampilan Produk

Tampilan Pembayaran



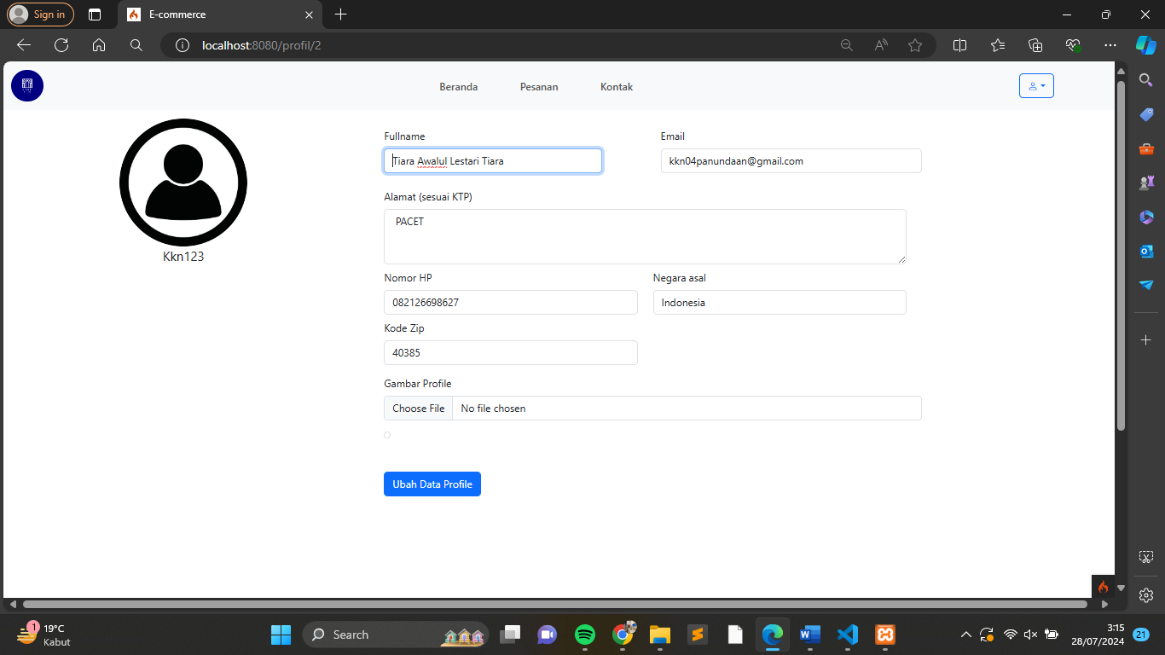
Gambar 5. 12 Tampilan Pembayaran

Tampilan pesanan/ Riwayat pesanan



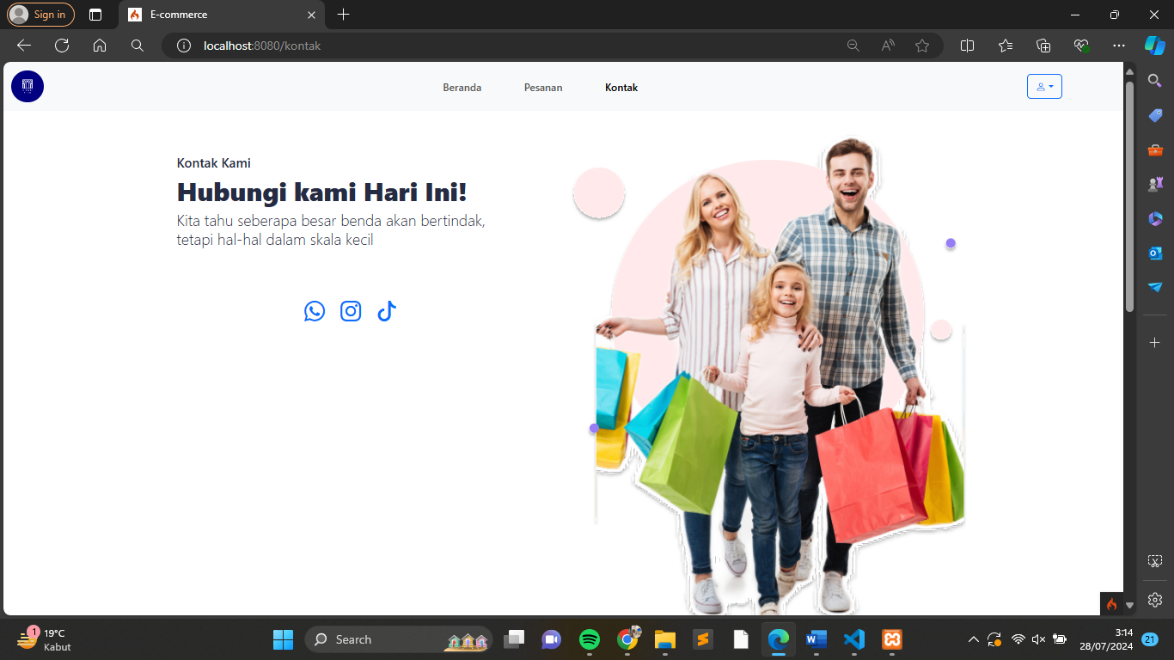
Gambar 5. 13 Tampilan Pesanan / Riwayat Pesanan

Tampilan Profil

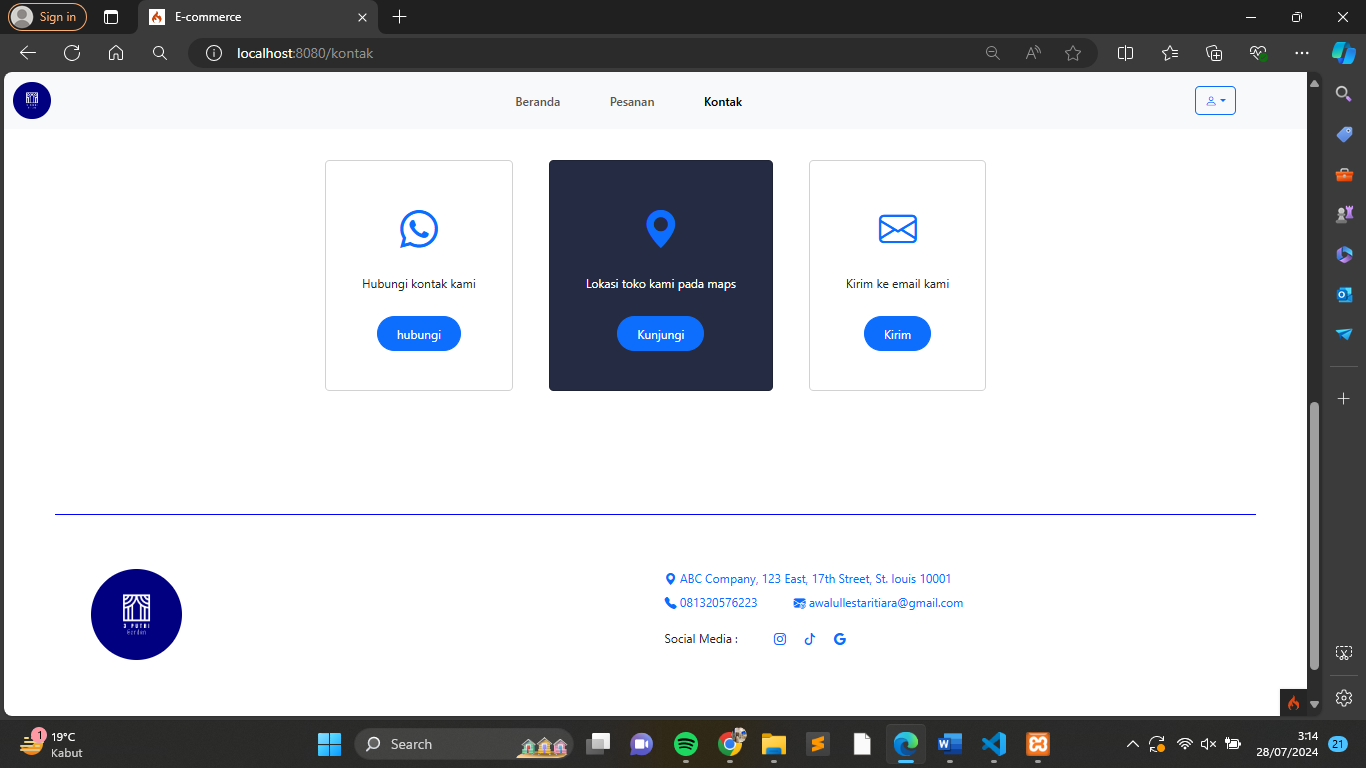


Gambar 5. 14 Tampilan Profil

Tampilan kontak



Gambar 5. 15 Tampilan Kontak



Gambar 5. 16 Tampilan Kontak

## Hasil Pengujian

Perlu adanya proses pengujian untuk menentukan kesalahan pada aplikasi sebelum aplikasi diterapkan di lapangan. Pada tahap pengujian, penulis menggunakan metode black box, yaitu metode pengujian yang mengabaikan mekanisme internal sistem atau komponen dan hanya berfokus pada output yang dihasilkan dalam menanggapi input yang dipilih dan kondisi eksekusi. Berikut merupakan hasil pengujian yang telah dilakukan:

Tabel 5. 1 Rencana Pengujian

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Item uji** | **Skenario uji** | **Hasil yang diharapkan** | **Hasil** | **Kesimpulan** |
| 1. | Form Login Admin User | Mengisi *username* dan pasword | Admin dan user dapat masuk ke sistem | Berhasil | Valid |
| 2. | Form Register admin user | Mendaftar username dan pasword | Terdaftarnya akun di sistem | Berhasil | Valid |
| 3. | Halaman home admin | Menambahkan jumlah pesanan dan pendapatan | Munculnya data data yang diharapkan | Berhasil | Valid |
| 4. | Halaman user admin | Klik ikon edit atau hapus user customer | Admin dapat mengedit atau menghapus user customer | Berhasil | Valid |
| 5. | Halaman produk | Klik tambah produk kemudian sistem menampilkan from input tambah produk | Admin dapat menambahkan produk dan menyimpannya | Berhasil | Valid |
| 6. | Edit data produk | Klik edit data produk kemudian sistem menampilkan from edit data produk | Admin dapat mengubah produk dan menyimpan nya | Berhasil | Valid |
| 7. | Halaman pesanan | Klik konfirmasi | Admin dapat mengkonfirmasi pesanan | Berhasil | Valid |
| 8. | Halaman beranda customer | User atau konsumen mengisi form login terlebih dahulu kemudian sistem menampilkan halaman utama | User atau customer dapat mengakes halaman utama yang berisi tampilan produk yang dijual | Berhasil | Valid |
| 9. | Searching produk | Klik ikon searching dan ketik nama produk yang ingin dicari | User atau customer dapat melihat hasil dari pencarian produkyang diinginkan | Berhasil | Valid |
| 10 | Lihat produk | Klik ikon lihat produk kemudian sistem menampilkan detail produk | User atau customer dapat melihat detail produk yang ditampilkan | Berhasil | Valid |
| 11 | Checkout | Klik ikon checkout | User atau customer dapat melihat detail produk dan pembayaran yang telah dicheckout | Berhasil | Valid |
| 12 | Halaman profil | Klik ikon profil | User atau customer bisa mengisi data diri yang dibutuhkan | Berhasil | Valid |
| 13 | Halaman pesanan | Klik pesanan kemudian sistem menampilkan riwayat pesanan | User atau cutomer dapat melihat riwayat pesanan produk apa saja yang sudah di checkout | Berhasil | Valid |
| 14 | Halaman kontak | Klik ikon kontak | User atau customer dapat melihat informasi tentang toko |  |  |

# BAB VI

# KESIMPULAN DAN SARAN

## Kesimpulan

Membangun *e-commerce* berbasis webite menggunakan *framework* codeignite*r* ini merupakan perancangan dari sistem yang berjalan. Berbagai masalah yang muncul telah coba untuk diselesaikan dengan sistem yang baru ini. Adapun kesimpulan yang dapat diambil antara lain:

* + 1. Dengan terancangnya *e-commerce* untuk Toko 3 Putri Gorden, toko ini akan mampu mengatasi berbagai kendala dalam menjangkau pelanggan yang lebih luas, khususnya mereka yang tidak berada di sekitar lokasi fisik toko. Sistem ini memberikan kemudahan bagi pelanggan untuk melihat dan membeli produk seperti gorden dan aksesoris rumah lainnya tanpa perlu datang langsung ke toko, cukup melalui perangkat mereka dari mana saja.
    2. Dengan terancangnya *e-commerce* untuk Toko 3 Putri Gorden, toko ini akan mendapatkan banyak keuntungan. Salah satunya adalah bisa menjangkau pelanggan yang lebih luas, termasuk di luar daerah lokal, bahkan hingga seluruh Indonesia. Hal ini membuka peluang penjualan yang lebih besar dibandingkan hanya mengandalkan toko fisik, Toko juga dapat dengan mudah menjalankan promosi online untuk menarik lebih banyak pelanggan.

## Saran

Berdasarkan kesimpulan yang telah dijabarkan, *e-commerce* berbasis webite menggunakan *framework* codeignite*r* masih jauh dari kesempurnaan. Maka dari itu, penulis memberikan saran sebagai bahan pertimbangan untuk pengembangan sistem kedepanya. Adapun sarannya sebagai berikut:

* + 1. Saran untuk penelitian selanjutnya dapat menggunakan metode lain. Hal ini bertujuan untuk mengetahui kelebihan serta kekurangan dari masing-masing metode yang digunakan sesuai dengan kebutuhan dari sistem yang akan dibangun.
    2. Sistem yang telah dibangun ini hendaknya di implementasikan dan dipelihara dengan baik agar dapat digunakan secara maksimal.

# DAFTAR PUSTAKA

Abas, Wahiddin. 2021. “Analisa Kepuasan Mahasiswa Terhadap Website Universitas Negeri Yogyakarta (UNY).” *Publikasi Ilmiah Unwahas* 1–6.

Abdillah, Rahman. 2021. “Pemodelan Uml Untuk Sistem Informasi Persewaan Alat Pesta.” *Jurnal Fasilkom* 11(2):79–86. doi: 10.37859/jf.v11i2.2673.

Abdul Wahid, Aceng. 2020. “Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi.” *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK* (November):1–5.

Agustina, Isna Fitria. 2024. Buku Ajar Pengantar Sistem Ekonomi Indonesia.

Anggraini, Yeni, Donaya Pasha, Damayanti Damayanti, and Aan Setiawan. 2020. “Sistem Informasi Penjualan Sepeda Berbasis Web Menggunakan Framework Codeigniter.” *Jurnal Teknologi Dan Sistem Informasi* 1(2):64–70. doi: 10.33365/jtsi.v1i2.236.

Amri, Kurnia Cahya Lestari dan Arni Muarifah, (2021). 2014. “Sistem Informasi Akuntansi.” *Akuntansi* 7(2):107–15

Christian, Yefta, and Denny Alfath. 2021. “Perancangan Sistem Manajemen Kerja Harian Berbasis Website Menggunakan Framework Codeigniter Di Universitas Internasional Batam.” 1(1):577–88.

Fathur Rohman. 2019. “Perubahan Budaya Jual Beli Tradisional Menjadi Jual Beli Online(Studi Kasus Penjualan Online Pada Tokabagus.Com).” *Journal of Chemical Information and Modeling* 53(9):1689–99.

Flora, Sylva, Ninta Tarigan, and Tri Septian Maksum. 2022. “PEMANFAATAN LAYANAN SISTEM INFORMASI E-PUSKESMAS UTILIZATION OF E-PUSKESMAS INFORMATION SYSTEM SERVICES USING THE.” 4(1):29–36.

Gusnita, E., Ali, H., & Rosadi, K. I. (2021). Model Sistemdalam Kontekspengertian, Jenis, Konstruksi, Berpikir Kesisteman Dalam Pendidikanislam. *Jurnal Manajemen Pendidikan Dan Ilmu Sosial*, *2*(2), 948–956.

Hapzi. 2019. “Sistem Informasi « sistem Informasi.” *Sistem Informasi* 2:2019.

Heriyanto, Yunahar. 2018. “Perancangan Sistem Informasi Rental Mobil Berbasis Web Pada PT.APM Rent Car.” *Jurnal Intra-Tech* 2(2):64–77. Josi, Ahmat. 2017. “Penerapan Metode Prototyping Dalam Membangun Website Desa (Studi Kasus Desa Sugihan Kecamatan Rambang).” *Jti* 9(1):50–57.

Josi, Ahmat. 2017. “Penerapan Metode Prototyping Dalam Membangun Website Desa (Studi Kasus Desa Sugihan Kecamatan Rambang).” *Jti* 9(1):50–57.

Krisna, Widatama, Hamid Jumasa Muhammad, and Nadia Ambadar. 2022. “Rancang Bangun Sistem Informasi Akademik Menggunakan Framework Codeigniter Pada Universitas Muhammadiyah Purworejo.” *Jurnal Sistem Cerdas* 5(2):107–16. doi: 10.37396/jsc.v5i2.187.

Kurniadi, Rahman, Cecep Riki, and Milah Nurkamilah. 2022. “Rancang Bangun Aplikasi Perpustakaan Berbasis Web Dengan Menggunakan Framework CodeIgniter.” *Formosa Journal of Science and Technology* 1(5):507–18. doi: 10.55927/fjst.v1i5.1209.

Kurniawati, putri. 2017. “No Titleالابتزاز الإلكتروني.. جرائم تتغذى على طفرة «التواصل ال.” *Universitas Nusantara PGRI Kediri* 01:1–7.

Muhyidin, Muhammad Agus, Muhammad Afif Sulhan, and Agus Sevtiana. 2020. “Perancangan Ui/Ux Aplikasi My Cic Layanan Informasi Akademik Mahasiswa Menggunakan Aplikasi Figma.” *Jurnal Digit* 10(2):208.

doi: 10.51920/jd.v10i2.171.

Mulyadi, 2001. Sistem Akuntansi. Jakarta : Salemba Empat.

Nopriandi, Helpi. 2018. “Perancangan Sistem Informasi Registrasi Mahasiswa.” *Jurnal Teknologi Dan Open Source* 1(1):73–79. doi: 10.36378/jtos.v1i1.1.

Nurdam, Nofriyadi. 2014. “Sequence Diagram Sebagai Perkakas Perancangan Antarmuka Pemakai.” *Jurnal ULTIMATICS* 6(1):21–25. doi: 10.31937/ti.v6i1.328.

Nurhayati, Ana Naela, Ahmat Josi, and Nur Aini Hutagalung. 2018. “Penjualan.” *Jurnal Teknologi Dan Informasi* 7(2):13–23.

Rachmawati, Rina. 2011. “Peranan Bauran Pemasaran (Marketing Mix) Terhadap Peningkatan Penjualan.” *Jurnal Kompetensi Teknik* 2(2):143–50.

Ramadhan, Rizky Fajar, and Riki Mukhaiyar. 2020. “Penggunaan Database Mysql Dengan Interface PhpMyAdmin Sebagai Pengontrolan Smarthome Berbasis Raspberry Pi.” *JTEIN: Jurnal Teknik Elektro Indonesia* 1(2):129–34. doi: 10.24036/jtein.v1i2.55.

Richter, Luiz Egon, Augusto Carlos, and De Menezes Beber. n.d. “E-Commerce.”

Saputra, Andika, and Yuli Astuti. 2018. “Analisis Pengaruh Struktur Html Terhadap.” *Jurnal Mantik Penusa* 2(2):34–39.

Sarwindah, S. (2018). Sistem Pendaftaran Siswa Baru Pada SMP N 1 Kelapa Berbasis Web Menggunakan Model UML. *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, *7*(2), 110–115.

SIHOTANG, HENGKI TAMANDO. 2019. “Sistem Informasi Pengagendaan Surat Berbasis Web Pada Pengadilan Tinggi Medan.” 3(1):6–9. doi: 10.31227/osf.io/bhj5q.

Sudarto Hasugian, P. (2018). *PERANCANGAN WEBSITE SEBAGAI MEDIA PROMOSI*. *3*(1), 82–86.

Susanto, Azhar. 2008. Sistem Informasi Akuntansi. Bandung : Lingga Jaya.

Soejono, Ajie Wibowo, Arief Setyanto, and Amir Fatah Sofyan. 2018. “Evaluasi Usability Website UNRIYO Menggunakan System Usability Scale (Studi Kasus: Website UNRIYO).” *Respati* 13(1):29–37. doi: 10.35842/jtir.v13i1.213

Trengginaz, Riro Bregas, Ade Yusup, Daniel Sovian Sunyoto, Muhammad Ruhul Jihad, and Yulianti Yulianti. 2020. “Pengujian Aplikasi Pemesanan Tiket Kereta Berbasis Website Menggunakan Metode Black Box Dengan Teknik Equivalence Partitioning.” Jurnal Teknologi Sistem Informasi Dan Aplikasi 3(3):144. doi: 10.32493/jtsi.v3i3.5349.

Tulungen, Evans E. W., David P. E. Saerang, and Joubert B. Maramis. 2022. “Transformasi Digital : Peran Kepemimpinan Digital.” *Jurnal EMBA : Jurnal Riset Ekonomi, Manajemen, Bisnis Dan Akuntansi* 10(2):1116–23. doi: 10.35794/emba.v10i2.41399.

Yadewani, Dorris, and Reni Wijaya. 2017. “Pengaruh E-Commerce Terhadap Minat Berwirausaha.” *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)* 1(1):64–69. doi: 10.29207/resti.v1i1.6.

Yusuf Amir, Achmad, Putri Aisyiyah, and Rakhma Devi. 2022. “Sistem Informasi Manajemen Penggajian Berbasis Framework Codeigniter Di CV. Citra Mandiri Gresik.” *Jurnal Fasilkom* Volume 12(1):35–42.

Yusuf, Muhammad, Sutrisno Sutrisno, PA Andiena Nindya Putri, Muhammad Asir, and Pandu Adi Cakranegara. 2022. “Prospek Penggunaan E-Commerce Terhadap Profitabilitas Dan Kemudahan Pelayanan Konsumen: Literature Review.” *Jurnal Darma Agung* 30(3):505. doi: 10.46930/ojsuda.v30i3.2268.

# LAMPIRAN

# BERITA ACARA HASIL WAWANCARA

Hari/tanggal : Minggu/24 Maret 2024

Tempat : Kp. Nunuk Kulon RT/RW 01/02 Ds. Mekarsari

Kec. Pacet Kab. Bandung.

**Profil Narasumber**

Nama : U. Yayat Hendayana

Umur : 49 Tahun

Jenis kelamin : Laki-laki

Jabatan : Pemilik

**Hasil Wawancara**

|  |  |  |  |
| --- | --- | --- | --- |
| Penulis | | **:** | Selamat siang pak. Terima kasih telah bersedia berbicara dengan kami mengenai produk yang dijual di toko "3 Putri Gorden". |
| Narasumber | | **:** | Selamat siang juga. Tentu, saya senang bisa berbagi informasi tentang produk kami. |
| Penulis | | **:** | Pertama-tama, bisakah bapak memberikan gambaran umum tentang jenis barang yang dijual di toko "3 Putri Gorden"? |
| Narasumber | | **:** | Tentu, di toko kami kami menawarkan berbagai macam gorden, tirai, perlengkapan dekorasi, dan aksesori terkait untuk rumah dan ruang kerja. Kami memiliki berbagai pilihan bahan, pola, dan gaya untuk memenuhi berbagai kebutuhan dekorasi pelanggan kami. |
| Penulis | | **:** | Apa yang membuat produk-produk "3 Putri Gorden" unik dibandingkan dengan pesaing lainnya? |
| Narasumber | | **:** | Salah satu hal yang membuat produk kami unik adalah kualitasnya. Kami hanya bekerja dengan pemasok terpercaya dan memastikan bahwa semua produk yang kami jual memiliki standar kualitas yang tinggi. Selain itu, kami juga menawarkan berbagai pilihan desain yang trendi dan beragam, sehingga pelanggan kami dapat menemukan sesuatu yang sesuai dengan gaya dan preferensi mereka. |
| Penulis | | **:** | Sejauh ini apa kendala yang bapak alami? |
| Narasumber | | **:** | Saat ini, kami lebih mengandalkan penjualan langsung melalui toko fisik kami dan juga promosi melalui media sosial. Namun, kami menyadari bahwa dengan adanya tren e-commerce yang semakin berkembang, penting bagi kami untuk memiliki kehadiran online yang kuat, jadi mungkin itu kendala yang saya alami saat ini. |
| Penulis | | **:** | Baik pak, setelah saya mendengar apa yang bapak alami tadi, saya jadi mempunya inspirasi untuk membuat e-commer seperti yang bapak maksud untuk keberlangsungan penjualan bapak dan juga unuk membantu lancarnya tugas akhir saya, apakah bapak berkenan akan itu? |
| Narasumber | | **:** | Tentu, karna itu yang saya harapkan |
| Penulis | | **:** | Sebelumnya saya ingin bertanya tentang apa saja produk yang paling populer atau laris di toko "3 Putri Gorden"? |
| Narasumber | | **:** | Produk-produk yang paling populer di toko kami adalah gorden blackout, tirai motif floral, vitrase, batang gorden, dan aksesoris aksesoris lainya juga. |
| Penulis | | **:** | Bagaimana proses penentuan harga untuk produk-produk di toko "3 Putri Gorden"? |
| Narasumber | | **:** | Kami mempertimbangkan beberapa faktor seperti biaya bahan baku, biaya produksi, serta keuntungan yang kami inginkan. Selain itu, kami juga memantau harga pasar dan strategi harga dari pesaing kami agar dapat menawarkan harga yang kompetitif namun tetap menguntungkan bagi kami. |
| Penulis | | **:** | Apa yang bapak harapkan dari terbangunya sistem e-commerce ini |
| Narasumber | | **:** | Yang saya harapakan jika terbangunya sistem ini adalah memudahkan saya untuk memasarkan produk saya secara luas dan juga membantu pelanggan agar dengan mudah membeli barang yang ada ditoko saya tanpa harus jauh jauh datang langsung. |
| Penulis | | **:** | Apakah ada keinginan yang spesifik untuk fitur fitur yang ada di dalam e-commerce ini? |
| Narasumber | | **:** | Mungkin saya tidak memiliki pengetahuan yang cukup untuk membuat keputusan terkait hal ini. Saya sangat menghargai bantuan dan panduan anda dalam memutuskan hal-hal terbaik yang perlu dilakukan. Saya mengikuti apapun yang anda buat karna saya yakin itu pasti yang terbaik, Dan juga tidak lupa saya bersedia untuk memberikan data yang Anda butuhkan untuk mendukung proses pengambilan keputusan Anda. |
| Penulis | | **:** | Baik, Pak. Terima kasih atas informasi nya. Untuk perkembangan lebih lanjut saya informasi kan lagi kepada Bapak. |
| Narasumber | **:** | Sama sama, saya juga mengucapkan terima kasih atas bantuan nya, semoga semuanya dilancarkan.  Ujang Yayat Hendayana  Pemilik Toko | |



Gambar Toko 3 Putri Gorden

Gambar Saat Observasi

# RIWAYAT HIDUP



**Data Diri**

Nama : Tiara Awalul Lestari

Tempat/ Tgl Lahir : Bandung, 24 Juli 2002

Jenis Kelamin : Perempuan

Agama : Islam

Status : Belum Menikah

Nama Ayah : Ujang Yayat Hendayana

Nama Ibu : Rosi Handayani

|  |  |  |
| --- | --- | --- |
| **Riwayat Pendidikan** | |  |
| * MI Rohmatul Ummah | 2008 - 2014 | |
| * MTS Quwatul Iman | 2014 - 2016 | |
| * MA Quwatul Iman | 2017 - 2020 | |
| * Universitas Bale Bandung (S1- Teknik Sistem Informasi) | 2020 - 2024 | |
| **Organisasi** | |  |
| * Departemen Luar Negri BEM FTI * Divisi Sumberdaya Dan Usaha HIMA * Wakil ketua Organisasi Siswa Intra Sekolah (MA) * Divisi Hubungan Masyarakat Organisasi Siswa Intra Sekolah (MTS)   **Motto**   * Untungnya, bumi masih berputar   Untungnya, ku tak pilih menyerah  Itu memang paling mudah  Untungnya, kupilih yang lebih susah  Untungnya, kupakai akal sehat  Untungnya, hidup terus berjalan  Untungnya, ku bisa rasa  Hal-hal baik yang datangnya belakangan  -Bernadya | |  |
| ***Contact***   * No. Telepon : 0821-2669-8627 * Instagram : tiaraawalull   E-mail : awalullestaritiara@gmail.com | |  |