PROPOSAL TUGAS AKHIR

**PERANCANGAN APLIKASI *POINT OF SALES* BERBASIS *WEBSITE* PADA FITUR *FRONT OFFICE* UNTUK KANTIN ASRAMA UNIVERSITAS TELKOM MENGGUNAKAN METODE *SCRUM***

**Oleh:**

**Alvin Pranata Kusuma**

**1202190313**



**PROGRAM STUDI STRATA 1 SISTEM INFORMASI**

**FAKULTAS REKAYASA INDUSTRI**

**UNIVERSITAS TELKOM**

**2023**

LEMBAR PENGESAHAN

Tugas Akhir dengan judul:

**PERANCANGAN APLIKASI *POINT OF SALES* BERBASIS *WEBSITE* PADA FITUR *FRONT OFFICE* DI KANTIN ASRAMA UNIVERSITAS TELKOM MENGGUNAKAN METODE *SCRUM***

Telah disetujui dan disahkan untuk *desk evaluation* proposal tugas akhir

Program Studi S1 Sistem Informasi

Fakultas Rekayasa Industri

Tahun 2023

Oleh

**ALVIN PRANATA KUSUMA**

**120219313**

Bandung, 07 Januari 2023

Disetujui oleh,

| Pembimbing 1 | Pembimbing 2 |
| --- | --- |
|  |  |
|
| Faishal Mufied Al-Anshary, S.Kom., M.Kom. MSc | Taufik Nur Adi, S.Kom., M.T., Ph.D. |
| NIP. 14900050 | NIP. 14830024 |

LEMBAR PERNYATAAN ORISINALITAS

ABSTRAK

ABSTRACT

KATA PENGANTAR

Daftar Isi

[**BAB I Pendahuluan 1**](#_1t3h5sf)

[I.1 Latar Belakang 1](#_sz1cbdd9qfig)

[I.2 Perumusan Masalah 2](#_2s8eyo1)

[I.3 Tujuan Penelitian 3](#_17dp8vu)

[I.4 Manfaat Penelitian 3](#_26in1rg)

[I.5 Batasan Penelitian 4](#_3rdcrjn)

[**BAB II Tinjauan Pustaka 5**](#_1ksv4uv)

[II.1 Aplikasi Point of Sales 5](#_b1ficogs7xvs)

[II.2 Website Pemesanan 5](#_2bb2qyyqv80)

[II.3 Metode Scrum 6](#_p7nang74gmq1)

[II.4 Javascript 7](#_954lv4c0m2uc)

[II.5 React 7](#_16jc5h3md2ut)

[II.6 Postman 8](#_7w0nv4mth74b)

[II.7 Penelitian Terdahulu 9](#_4tir2o3x35qi)

[**BAB III Metodologi Penelitian 11**](#_rp0f3x5tqxn8)

[III.1 Model Konseptual 11](#_umfqojetp16s)

[III.2 Sistematika Penyelesaian Masalah 12](#_1ci93xb)

[III.3 Alasan Pemilihan Metode 14](#_zevighenw8l1)

[III.4 Pengumpulan Data 14](#_3whwml4)

[III.5 Pengolahan Data 15](#_2bn6wsx)

[III.6 Metode Evaluasi 16](#_qsh70q)

[III.7 Rencana Jadwal Kegiatan 17](#_at45un60xzy3)

[**BAB IV Analisis dan Perancangan 18**](#_7e8qh623smzg)

[IV.1 Proses Bisnis 18](#_av3u34mi5amh)

[IV.2 Product Backlog 19](#_l1538dr7acmr)

[IV.3 Use Case Diagram 22](#_i9v1wvurmzyg)

[IV.4 Use Case Scenario 22](#_c6c35gq3sakv)

[IV.5 Activity Diagram 31](#_gy6mbngnk8lb)

[IV.6 Class Diagram 41](#_cstjaf1wwwk0)

[IV.7 Sequence Diagram 42](#_c608xp1970si)

[IV.8 Entity Relationship Diagram 49](#_g9eml8qy24rg)

[IV.9 Relational Table Diagram 50](#_7gksqkiwun9m)

[IV.10 Deployment Diagram 51](#_qmc7i41tz7t)

[IV.11 Low Fidelity 52](#_aiaj5ywillwd)

[**DAFTAR PUSTAKA 57**](#_fz2xre1i1zmh)

Daftar Gambar

[Gambar III.1 Model Konseptual 11](#_8ya13kglmyxb)

[Gambar III.2 Metode Scrum 12](#_wodr2j3p0n0m)

[Gambar III.3 Sprint 17](#_quzyu0j2yhkz)

[Gambar IV.1 Proses Bisnis Front Office Pada Kantin Telkom University 18](#_44s2jv19fbb)

[Gambar IV.2 Use Case Diagram Reztopia Fitur Front Office 22](#_groonhydcrza)

[Gambar IV.4 Activity Diagram Melakukan Register 32](#_t4tyvht797ts)

[Gambar IV.5 Activity Diagram Melakukan Login 33](#_us9k2padokp1)

[Gambar IV.6 Activity Diagram Memilih Tenant 34](#_5c1m152emybh)

[Gambar IV.7 Activity Diagram Memilih Menu 35](#_kmc063kgxau8)

[Gambar IV.I 8 Activity Diagram Memilih Keranjang 36](#_25dme3rzxax0)

[Gambar IV.9 Activity Diagram Melakukan Pembayaran 36](#_sv4p66g3egiu)

[Gambar IV.10 Activity Diagram Detail Pesanan 37](#_afh8wm3i2j)

[Gambar IV.11 Activity Diagram Riwayat Transaksi 38](#_8o112l4ge889)

[Gambar IV.12 Activity Diagram Mengelola Profil 38](#_lz9c0rurfees)

[Gambar IV.13 Activity Diagram Melakukan Logout 39](#_1n5f74xujk7s)

[Gambar IV.14 Activity Diagram Mengelola Pesanan 40](#_obu4mnlkr22b)

[Gambar IV.15 Activity Diagram Mengelola Riwayat Pesanan 41](#_trywm2x2e6we)

[Gambar IV.16 Class Diagram Reztopia Fitur Front Office 42](#_weyaecgsz18w)

[Gambar IV.17 Sequence Diagram Melakukan Register 42](#_eztl5cwjlax6)

[Gambar IV.18 Sequence Diagram Melakukan Login 43](#_mu8jv2jp6onl)

[Gambar IV.19 Sequence Diagram Memilih Tenant 43](#_r1uhlnpd98x7)

[Gambar IV.20 Sequence Diagram Memilih Menu 44](#_iomvdy9evqgb)

[Gambar IV.21 Sequence Diagram Melihat Keranjang 44](#_gv0icfkc99zq)

[Gambar IV.22 Sequence Diagram Melakukan Pembayaran 45](#_mclvjk7e64hz)

[Gambar IV.23 Sequence Diagram Detail Pesanan 46](#_mwdep97r73th)

[Gambar IV.24 Sequence Diagram Mengelola Riwayat Transaksi 46](#_q6uuatgcn1wr)

[Gambar IV.25 Sequence Diagram Mengelola Profil 47](#_89lsnibd99o2)

[Gambar IV.26 Sequence Diagram Mengelola Logout 47](#_yjw44rqjs054)

[Gambar IV.27 Sequence Diagram Mengelola Pesanan 48](#_8z9hjt2elg44)

[Gambar IV.28 Sequence Diagram Riwayat Pesanan 48](#_qeixyhe5bz2p)

[Gambar IV.29 Entity Relationship Diagram Fitur Front Office 49](#_mwll7v3eil4q)

[Gambar IV.30 Relational Table Diagram Fitur Front Office 50](#_gaerbkchw25p)

[Gambar IV.31 Relational Table Diagram Fitur Front Office 51](#_4mwxkz4dngrr)

[Gambar IV.32 Low Fidelity: Reztopia POS - Login 52](#_spmjo8a4pcms)

[Gambar IV.32 Low Fidelity: Reztopia POS - Order 53](#_dga61855676x)

[Gambar IV.32 Low Fidelity: Reztopia POS - Checkout 54](#_vro4plergfhh)

[Gambar IV.32 Low Fidelity: Reztopia POS - History 54](#_str96ha0yfie)

[Gambar IV.32 Low Fidelity: Reztopia POS - Profile 55](#_zepnnk7lv7nd)

[Gambar IV.32 Low Fidelity: Reztopia POS - OrderManager 56](#_8yv9blbkof30)

Daftar Tabel

[Tabel II.1 Penelitian Terdahulu: Ilham Fannani 9](#_gmb1837yj76s)

[Tabel II.2 Penelitian Terdahulu: Nailul Marom 10](#_hayuufwz6t0k)

[Tabel II.3 Penelitian Terdahulu: Muhamad Noufal Ajriya Siddik 10](#_jmo8b4wuxupu)

[Tabel IV.1 Kebutuhan Fungsionalitas Reztopia Fitur Front Office 20](#_t55hurjoe1sa)

[Tabel IV.2 Kebutuhan Fungsionalitas Reztopia Fitur Front Office 21](#_ay52k4fa80q2)

[Tabel IV.3 Aktor Reztopia Fitur Front Office 21](#_u01chiaxcmvu)

Daftar Lampiran

# Pendahuluan

## **Latar Belakang**

Kantin Asrama Universitas Telkom merupakan sebuah pusat kuliner yang berada di lingkungan asrama Universitas Telkom. Kantin ini berisi berbagai tenant yang menjual aneka makanan dan minuman untuk memenuhi kebutuhan mahasiswa. Sebagai tempat yang strategis dan sering dikunjungi, kantin ini dikelola secara profesional oleh Direktorat Logistik dan Aset Universitas Telkom. Namun, terdapat beberapa masalah operasional yang dihadapi oleh mahasiswa dalam menggunakan fasilitas ini (Putri & Wulandari, 2020).

Mahasiswa yang ingin memesan makanan atau minuman di kantin ini seringkali mengalami kesulitan dan ketidaknyamanan. Salah satunya adalah mereka harus berkeliling kantin untuk melihat menu yang disediakan oleh setiap tenant. Hal ini tidak efisien dan seringkali menghabiskan waktu yang seharusnya dapat digunakan untuk beristirahat atau belajar. Selain itu, proses pembayaran juga menjadi kendala. Mahasiswa harus berjalan bolak-balik dari tenant ke kasir untuk melakukan pembayaran dan kembali lagi ke tenant untuk menyerahkan bukti pembayaran. Proses ini memakan waktu dan tenaga, serta berpotensi menimbulkan antrian dan kerumunan.

Selain masalah yang dialami oleh mahasiswa, tenant di kantin asrama Universitas Telkom juga menghadapi beberapa tantangan. Tenant seringkali merasa kesulitan dalam mengelola pesanan karena metode yang digunakan masih bersifat manual. Dalam kondisi ramai, hal ini bisa menjadi sangat membingungkan dan menghambat efisiensi operasional. Selain itu, mahasiswa seringkali berpindah tempat duduk, yang membuat tenant kesulitan dalam menemukan mereka ketika pesanan sudah siap. Proses pencatatan dan penandaan pesanan yang manual juga sering kali memunculkan kesalahan, baik dalam hal penyalahan pesanan atau jumlah pembayaran.

Untuk mengatasi masalah ini, sebuah solusi berbentuk aplikasi point of sales (POS) berbasis website dapat diterapkan pada fitur front office di kantin Asrama Universitas Telkom. Aplikasi ini akan memfasilitasi mahasiswa untuk melihat menu dan memesan makanan atau minuman dari tenant mana pun langsung dari perangkat mereka. Proses pembayaran dapat dilakukan secara digital melalui aplikasi, sehingga mengurangi kebutuhan untuk bergerak bolak-balik antara tenant dan kasir. Di sisi lain, aplikasi ini juga membantu tenant dalam mengelola pesanan dan pembayaran, serta menandai lokasi mahasiswa untuk memudahkan pengantaran pesanan. Dengan demikian, metode Scrum yang fleksibel dan iteratif dapat digunakan dalam pengembangan aplikasi ini, sehingga dapat memenuhi kebutuhan dan menyelesaikan masalah yang dihadapi oleh semua pihak yang terlibat.

## **Perumusan Masalah**

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, rumusan masalah yang dapat dibuat adalah:

1. Bagaimana merancang aplikasi point of sales berbasis website yang memungkinkan mahasiswa untuk melihat menu dan memesan makanan atau minuman secara self-service di meja masing-masing, sehingga tidak perlu berkeliling kantin?
2. Bagaimana merancang sistem pembayaran dalam aplikasi point of sales berbasis website tersebut agar mahasiswa dapat melakukan pembayaran secara efisien dan mudah tanpa perlu berjalan bolak-balik dari tenant ke kasir dan kembali lagi ke tenant untuk menyerahkan bukti pembayaran?
3. Bagaimana merancang aplikasi point of sales berbasis website yang memungkinkan tenant untuk mengelola pesanan secara efisien, memudahkan penemuan mahasiswa yang berpindah tempat duduk, serta mengurangi kesalahan dalam pencatatan dan penandaan pesanan secara manual?

## **Tujuan Penelitian**

Berdasarkan rumusan masalah yang telah dibuat, tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang sistem pemesanan dalam aplikasi point of sales, yang dapat memfasilitasi mahasiswa untuk melihat menu dan melakukan pemesanan makanan atau minuman dari meja mereka masing-masing.
2. Merancang sistem pembayaran yang terintegrasi dalam aplikasi point of sales, yang memungkinkan mahasiswa untuk melakukan pembayaran secara efisien dan mudah dari meja mereka masing-masing.
3. Merancang sistem manajemen pesanan dalam aplikasi point of sales, yang memungkinkan tenant untuk mengelola pesanan dengan efisiensi tinggi, memudahkan penemuan mahasiswa yang berpindah tempat duduk, serta mengurangi kesalahan dalam pencatatan dan penandaan pesanan secara manual.

## **Manfaat Penelitian**

Diharapkan, penelitian ini akan memberikan manfaat dalam tiga aspek, yaitu:

1. Bagi mahasiswa, penelitian ini diharapkan dapat memberikan solusi teknologi yang memudahkan proses pemesanan dan pembayaran di kantin. Dengan adanya aplikasi point of sales berbasis website ini, mereka dapat melihat menu dan memesan makanan atau minuman langsung dari meja mereka, serta melakukan pembayaran secara efisien tanpa perlu berjalan bolak-balik antara tenant dan kasir. Ini akan mengoptimalkan waktu dan tenaga yang biasanya terbuang dalam proses pemesanan dan pembayaran.
2. Bagi Universitas Telkom, hasil penelitian ini diharapkan dapat membantu meningkatkan efisiensi operasional kantin dan kepuasan mahasiswa terhadap fasilitas yang disediakan. Aplikasi ini akan mempermudah manajemen kantin dalam mengelola proses pemesanan dan pembayaran, sehingga dapat mengurangi antrian dan kerumunan. Hal ini pada gilirannya dapat meningkatkan citra positif universitas di mata mahasiswa dan masyarakat umum.
3. Bagi peneliti lain yang bergerak dalam sistem informasi pendidikan tinggi, hasil rancangan website point of sales ini dapat menjadi referensi untuk penelitian lainnya. Metodologi, teknik, dan temuan dari penelitian ini dapat digunakan sebagai acuan atau inspirasi dalam merancang dan mengimplementasikan solusi teknologi informasi lainnya yang bertujuan untuk meningkatkan efisiensi dan kenyamanan dalam lingkungan pendidikan tinggi.

## **Batasan Penelitian**

Batasan dari penelitian ini adalah sebagai berikut:

1. Penelitian ini hanya berfokus pada perancangan dan implementasi aplikasi point of sales berbasis website pada fitur front office untuk kantin asrama Universitas Telkom.
2. Aplikasi yang dirancang hanya mencakup fungsi pemesanan makanan atau minuman, pembayaran self-service, dan pengelolaan pesanan dengan menyediakan informasi daftar pesanan yang jelas.
3. Meski metode scrum digunakan dalam perancangan aplikasi ini, penelitian ini tidak akan membahas secara detail tentang metode scrum itu sendiri. Fokus penelitian lebih kepada penerapan metode tersebut dalam konteks pengembangan aplikasi point of sales.
4. Evaluasi kinerja dan kegunaan aplikasi ini hanya akan dilakukan terhadap mahasiswa Universitas Telkom yang menggunakan kantin universitas.

# Tinjauan Pustaka

## **Aplikasi Point of Sales**

Aplikasi Point of Sale (POS) adalah sebuah perangkat lunak yang digunakan untuk mengelola dan memfasilitasi berbagai kegiatan penjualan dan transaksi di sebuah toko atau bisnis ritel. POS menjadi sangat penting dalam operasional harian bisnis, karena menggantikan peran sistem kasir tradisional dengan solusi digital yang lebih efisien dan akurat. Aplikasi POS biasanya mencakup fitur seperti pembayaran, manajemen inventaris, pelacakan penjualan, dan laporan keuangan.

Dalam konteks bisnis, penggunaan aplikasi POS dapat memberikan sejumlah manfaat. Pengguna dapat meningkatkan efisiensi operasional, mengurangi kesalahan dalam penghitungan, mempercepat waktu transaksi, dan mengoptimalkan pengelolaan inventaris. Selain itu, aplikasi POS juga dapat memberikan data dan analisis yang canggih, yang memungkinkan pemilik bisnis untuk memahami tren penjualan, melacak performa produk, dan mengidentifikasi peluang bisnis yang baru (Suryadi & Setiawan, 2022).

## **Website Pemesanan**

Website Pemesanan adalah sebuah situs web yang digunakan untuk melakukan pemesanan atau reservasi secara online. Tujuan utama dari website ini adalah mempermudah proses pemesanan bagi pelanggan dan meningkatkan efisiensi operasional bagi bisnis yang menggunakan layanan pemesanan tersebut. Dalam konteks umum, website pemesanan dapat ditemukan dalam berbagai sektor, mulai dari perhotelan, restoran, tempat hiburan, hingga layanan transportasi.

Dalam bidang F&B (Food and Beverage), website pemesanan memiliki peran yang penting dalam memfasilitasi pemesanan makanan dan minuman secara online. Pelanggan dapat mengakses website tersebut melalui perangkat mereka, melakukan pemesanan, memilih menu, dan melakukan pembayaran. Selain itu, website pemesanan F&B juga dapat menyediakan informasi tambahan seperti jam operasional, lokasi, ulasan, dan promosi spesial.

Dengan menggunakan website pemesanan, bisnis di bidang F&B dapat mengoptimalkan proses pemesanan dan meningkatkan kepuasan pelanggan. Penggunaan website ini dapat mengurangi ketergantungan pada pesanan telepon atau langsung, yang seringkali dapat menimbulkan kesalahan dan keterlambatan. Selain itu, website pemesanan juga dapat memberikan informasi yang lebih akurat tentang ketersediaan produk dan memungkinkan bisnis untuk melakukan manajemen inventaris yang lebih baik (Munawar, 2018).

## **Metode Scrum**

Metode Scrum adalah sebuah kerangka kerja manajemen proyek yang tergolong dalam metodologi Agile. Metode ini digunakan untuk mengelola proyek yang kompleks dan terus berkembang, terutama dalam pengembangan perangkat lunak. Scrum berfokus pada pengaturan waktu dan pendekatan iteratif serta kolaboratif dalam menghasilkan produk yang bernilai. Dalam Scrum, tim proyek bekerja dalam periode waktu yang disebut sprint, yang biasanya berlangsung selama 1 hingga 4 minggu.

Proses kerja dalam Scrum melibatkan tiga peran utama: Product Owner, Scrum Master, dan Tim Pengembang. Product Owner bertanggung jawab untuk mengartikulasikan kebutuhan pengguna dan merancang produk yang sesuai. Scrum Master berperan sebagai pemimpin tim yang membantu memastikan penerapan Scrum yang efektif dan menghilangkan hambatan yang mungkin muncul. Tim Pengembang terdiri dari individu yang bertanggung jawab untuk mengerjakan pekerjaan dalam sprint.

Scrum juga menggunakan artefak seperti Product Backlog, Sprint Backlog, dan Burndown Chart untuk membantu pengelolaan proyek. Product Backlog adalah daftar tugas yang harus diselesaikan dalam proyek, sedangkan Sprint Backlog adalah subset dari Product Backlog yang harus diselesaikan dalam sprint tertentu. Burndown Chart digunakan untuk melacak kemajuan dan transparansi dalam menyelesaikan pekerjaan (Arifianto, 2021).

## **Javascript**

JavaScript adalah bahasa pemrograman tingkat tinggi yang sering digunakan untuk mengembangkan aplikasi web interaktif. Dikembangkan pada tahun 1995 oleh Brendan Eich, JavaScript awalnya dirancang sebagai bahasa scripting untuk meningkatkan interaktivitas halaman web. Namun, sejak saat itu, JavaScript telah berkembang menjadi bahasa pemrograman yang kuat dengan kemampuan yang luas. Bahasa ini berjalan di sisi klien (client-side) pada browser web, yang memungkinkan pengembang untuk membuat aplikasi yang responsif dan interaktif.

Salah satu fitur utama JavaScript adalah kemampuannya untuk memanipulasi elemen HTML dan CSS pada halaman web secara dinamis. Dengan menggunakan JavaScript, pengembang dapat menangani event (peristiwa) seperti klik tombol, mengubah isi halaman secara real-time, memvalidasi formulir, dan banyak lagi. Bahasa ini juga mendukung pemrograman berorientasi objek, yang memungkinkan pengembang untuk mengorganisir kode ke dalam objek yang dapat digunakan kembali, meningkatkan modularitas dan skalabilitas.

Selain itu, JavaScript juga memiliki berbagai pustaka dan kerangka kerja (framework) yang populer, seperti React, Angular, dan Vue.js, yang memperluas kemampuan bahasa ini dalam membangun aplikasi web yang kompleks. Dengan menggunakan kerangka kerja ini, pengembang dapat mempercepat proses pengembangan, memanfaatkan komponen siap pakai, dan mengoptimalkan kinerja aplikasi (Firdaus, & Taufik, 2020)

## **React**

React adalah sebuah perpustakaan JavaScript yang sangat populer untuk membangun antarmuka pengguna (user interface) yang interaktif dan responsif. Dikembangkan oleh Facebook, React menggunakan pendekatan pemrograman berbasis komponen, di mana antarmuka pengguna dibagi menjadi komponen kecil yang dapat digunakan kembali. Ini memungkinkan pengembang untuk memisahkan logika aplikasi dan antarmuka pengguna menjadi bagian yang terpisah, sehingga memudahkan pengelolaan kode dan meningkatkan skalabilitas.

Salah satu fitur utama React adalah Virtual DOM (Document Object Model). Virtual DOM adalah representasi virtual dari struktur DOM aktual dalam memori, yang memungkinkan React untuk melakukan pembaruan yang efisien pada halaman web. Ketika ada perubahan pada data atau keadaan (state) komponen, React akan memperbarui Virtual DOM dan membandingkannya dengan DOM aktual untuk mengidentifikasi perubahan yang harus diterapkan. Proses ini membantu mengurangi jumlah manipulasi langsung pada DOM, yang pada gilirannya meningkatkan kinerja aplikasi.

Selain itu, React juga menggunakan sintaks yang disebut JSX (JavaScript XML), yang menggabungkan HTML dan JavaScript. Dengan JSX, pengembang dapat membuat struktur antarmuka pengguna secara deklaratif dan menyatukan logika JavaScript di dalamnya. Ini memungkinkan pengembang untuk dengan mudah menggabungkan logika dan tampilan dalam satu tempat, meningkatkan keterbacaan kode dan memudahkan pemeliharaan (Wibisono & Setiawan, 2021)

## **Postman**

Postman adalah sebuah aplikasi pengujian API (Application Programming Interface) yang sangat populer di kalangan pengembang perangkat lunak. Dikembangkan oleh Postman Inc., Postman menyediakan antarmuka yang intuitif dan lengkap untuk menguji, memvalidasi, dan mendokumentasikan API. Dengan Postman, pengembang dapat mengirim permintaan HTTP ke API, menangani respons, dan melakukan debugging untuk memastikan bahwa API berfungsi dengan baik.

Salah satu fitur utama Postman adalah kemampuannya untuk mengirim permintaan HTTP ke berbagai endpoint API. Pengembang dapat dengan mudah menentukan metode HTTP yang digunakan (seperti GET, POST, PUT, DELETE), melampirkan header dan parameter, dan mengirim permintaan ke server. Postman juga menyediakan fitur pengelolaan koleksi permintaan, yang memungkinkan pengembang untuk mengorganisir dan menyimpan permintaan yang sering digunakan untuk pengujian dan debugging yang lebih efisien.

Selain itu, Postman juga dilengkapi dengan fitur pemantauan API dan pengujian otomatis. Pengembang dapat membuat skrip pengujian yang dapat dijalankan secara otomatis untuk memverifikasi kinerja dan keandalan API. Ini memungkinkan pengembang untuk mengotomatisasi pengujian dan mempercepat siklus pengembangan (Ahmad Ramadhan, 2019

## **Penelitian Terdahulu**

| **Nama Peneliti** | **Judul** | **Masalah** |
| --- | --- | --- |
| Ilham Fannani | Perancangan Aplikasi POINT OF SALES Berbasis Website Togoide Studi Kasus UKM Laundry Menggunakan Metode Iterative dan Incremental | Penelitian ini mengangkat masalah Website Togoide Studi Kasus UMKM Laundry |
| **Solusi dan Hasil :** User Interface Pada Aplikasi Customer Layanan Pemesanan Makanan “EatAja” | | |
| **Perbandingan :** Penelitian terdahulu ini mencakup perancangan aplikasi Point of Sales (POS) berbasis website untuk usaha kecil menengah (UKM) laundry. Metode yang digunakan dalam penelitian ini adalah metode iterative dan incremental, yang berfokus pada pengembangan bertahap dan peningkatan secara berulang terhadap aplikasi tersebut. | | |

### Tabel II.1 Penelitian Terdahulu: Ilham Fannani

Sumber : (Fannani, 2020)

| **Nama Peneliti** | **Judul** | **Masalah** |
| --- | --- | --- |
| Nailul Marom | Perancangan User Interface Pada Aplikasi Customer Layanan Pemesanan Makanan “EatAja” Menggunakan Metode User Centered Design | Penelitian ini mengangkat masalah User Interface Pada Aplikasi Customer Layanan Pemesanan Makanan “EatAja” |
| **Solusi dan Hasil :** User Interface Pada Aplikasi Customer Layanan Pemesanan Makanan “EatAja” | | |
| **Perbandingan :** Penelitian terdahulu ini berfokus pada perancangan antarmuka pengguna (user interface) pada aplikasi "EatAja" yang merupakan aplikasi layanan pemesanan makanan bagi pelanggan. Metode yang digunakan dalam penelitian ini adalah metode User Centered Design, yang berfokus pada kebutuhan dan pengalaman pengguna untuk menghasilkan antarmuka yang intuitif dan mudah digunakan. | | |

### Tabel II.2 Penelitian Terdahulu: Nailul Marom

Sumber : (Nailul, 2020)

| **Nama Peneliti** | **Judul** | **Masalah** |
| --- | --- | --- |
| Muhamad Noufal Ajriya Siddik | Perancangan User Interface Mitra Sebagai Penerima Pesanan Pada Aplikasi EATAJA Mitra Dengan Metode Goal-Directed Design | Penelitian ini mengangkat masalah Perancangan User Interface Mitra Sebagai Penerima Pesanan Pada Aplikasi EATAJA |
| **Solusi dan Hasil :** Perancangan User Interface Mitra Sebagai Penerima Pesanan Pada Aplikasi EATAJA | | |
| **Perbandingan :** Penelitian terdahulu ini bertujuan untuk merancang antarmuka pengguna bagi mitra sebagai penerima pesanan pada aplikasi "EATAJA Mitra". Metode yang digunakan dalam penelitian ini adalah metode Goal-Directed Design, yang berfokus pada pemahaman tujuan dan kebutuhan pengguna dalam merancang antarmuka yang mendukung pencapaian tujuan tersebut. | | |

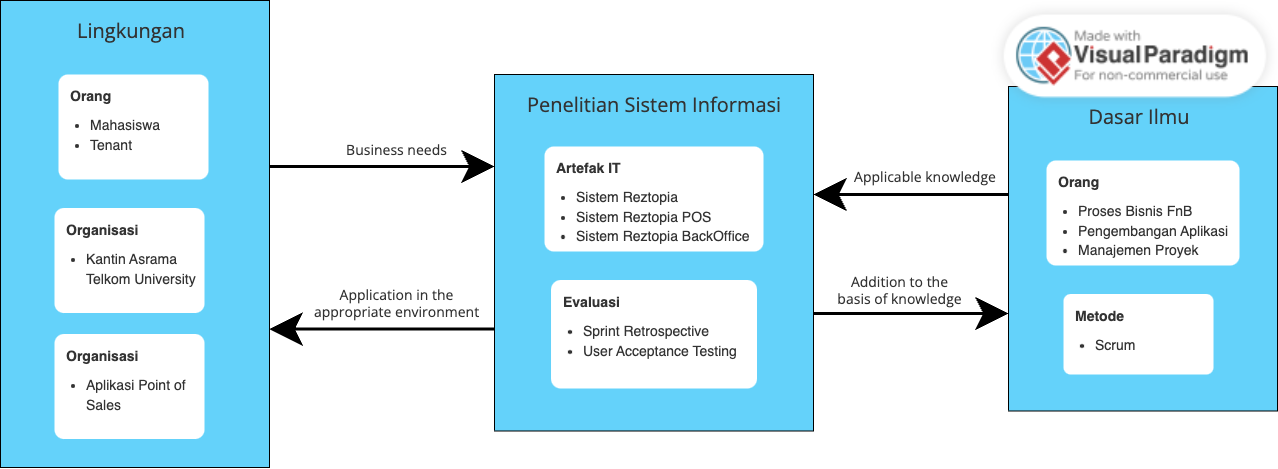
### Tabel II.3 Penelitian Terdahulu: Muhamad Noufal Ajriya Siddik

Sumber : (Siddik, 2020)

# 

# Metodologi Penelitian

## **Model Konseptual**

Pengembangan model konseptual merupakan representasi abstrak yang digunakan oleh peneliti untuk memahami, menerapkan, dan mengevaluasi penelitian. Penelitian ini mengadopsi model konseptual berdasarkan paradigma design science yang bertujuan untuk mengembangkan solusi berbasis teknologi yang terkait dengan permasalahan pada sisi desain dan bisnis. 

### Gambar III.01 Model Konseptual

Model konseptual di atas menjelaskan perancangan aplikasi point of sales berbasis website untuk kantin Telkom University dengan menggunakan metode Scrum. Lingkungan yang terlibat dalam aplikasi ini adalah mahasiswa sebagai pelanggan dan tenant sebagai pihak yang berjualan. Organisasi yang terlibat adalah Kantin Telkom University sebagai UMKM yang memerlukan aplikasi ini untuk memudahkan transaksi dan manajemen penjualan. Teknologi yang digunakan meliputi website dan aplikasi point of sales dengan integrasi mesin kasir dan sistem inventaris. Penelitian Sistem Informasi menggunakan artefak IT berupa reztopia, reztopia pos, dan reztopia backoffice. Evaluasi dilakukan melalui sprint retrospective dan user acceptance testing (UAT). Dasar ilmu yang digunakan meliputi proses bisnis fnb, pengembangan aplikasi, dan manajemen proyek dengan metode pengembangan menggunakan metode Scrum. Dengan model konseptual ini, diharapkan perancangan aplikasi point of sales berbasis website dapat membantu kantin Telkom University dalam mengoptimalkan transaksi dan manajemen penjualan (Pramono & Utami, 2021).

## **Sistematika Penyelesaian Masalah**

### 

### 

### 

### 

### 

### Gambar III.02 Metode Scrum

Sistematika penyelesaian masalah dilakukan dengan tahapan sebagai berikut (Utami & Pramono, 2020):

1. Perencanaan awal (Initial Planning): Pada tahap ini, tim proyek harus menetapkan tujuan jangka panjang dan visi dari proyek yang akan dilakukan. Kemudian, tim perlu mengumpulkan kebutuhan proyek dari stakeholder dan memprioritaskan kebutuhan tersebut dalam Product Backlog. Hal ini dilakukan untuk memastikan bahwa proyek dapat menghasilkan solusi yang memenuhi kebutuhan stakeholder dan sesuai dengan tujuan dan visi proyek.
2. Sprint Planning: Pada tahap ini, tim proyek akan memilih item-item dari Product Backlog yang akan dikerjakan pada Sprint saat ini dan mencatat item-item tersebut ke dalam Sprint Backlog. Selanjutnya, tim akan membagi item-item Sprint Backlog menjadi tugas-tugas yang lebih kecil dan konkret, menentukan waktu dan usaha yang diperlukan untuk setiap tugas. Tujuan dari tahap ini adalah untuk mengoptimalkan penggunaan waktu dan sumber daya, sehingga proyek dapat diselesaikan dengan efisien.
3. Daily Stand-up: Pertemuan harian singkat untuk melaporkan kemajuan, hambatan, dan koordinasi tugas serta sinkronisasi dan kolaborasi antara anggota tim. Pertemuan harian ini bertujuan untuk memastikan bahwa semua anggota tim terus berada pada jalur yang sama, menjaga proyek tetap berjalan sesuai jadwal, serta mengatasi masalah atau hambatan yang muncul sebelum menjadi masalah yang lebih besar.
4. Sprint: Pada tahap ini, tim proyek akan melaksanakan tugas-tugas yang direncanakan dalam Sprint Backlog dan berkoordinasi dengan anggota tim lainnya. Selama Sprint, anggota tim akan berfokus pada tugas-tugas yang ditugaskan dan memastikan bahwa semua tugas tersebut diselesaikan sesuai dengan waktu yang ditentukan dalam Sprint Planning.
5. Sprint Review: Pada tahap ini, tim proyek akan mengevaluasi hasil pengembangan, melakukan demonstrasi hasil kerja kepada stakeholder, dan membahas feedback serta penyesuaian rencana. Hal ini dilakukan untuk memastikan bahwa hasil pengembangan sesuai dengan kebutuhan stakeholder dan memperbaiki rencana jika diperlukan. Sprint Retrospective: Pada tahap ini, tim proyek akan mengevaluasi proses dan hasil Sprint, mengidentifikasi area perbaikan dan rencana aksi, serta mengembangkan strategi untuk meningkatkan efektivitas dan efisiensi tim. Tujuan dari tahap ini adalah untuk memperbaiki proses pengembangan dan meningkatkan efisiensi tim untuk Sprint berikutnya.
6. Pengujian (Testing): Pada tahap ini, tim proyek akan menyusun skenario dan kasus uji, melaksanakan pengujian unit, integrasi, dan sistem, serta mencatat dan melacak bug serta penyelesaiannya. Tahap ini dilakukan untuk memastikan bahwa solusi yang dikembangkan dapat berfungsi dengan baik dan memenuhi kebutuhan stakeholder.
7. Peluncuran (Release): Pada tahap ini, tim proyek akan menyusun dan memvalidasi dokumen peluncuran, melaksanakan rencana peluncuran dan penyebaran solusi. Tahap ini bertujuan
8. Kesimpulan (Conclusion): tahap terakhir dalam metode Scrum. Pada tahap ini, dilakukan evaluasi keseluruhan proyek, penutupan proyek dan transfer pengetahuan ke tim pemeliharaan atau proyek lainnya, serta penyusunan laporan akhir proyek. Hal ini bertujuan untuk memaksimalkan nilai bisnis dan pengembangan tim pada proyek selanjutnya.

## **Alasan Pemilihan Metode**

Alasan pemilihan metode Scrum adalah karena Scrum memiliki beberapa kelebihan dalam pengembangan perangkat lunak, yaitu (Septiandani & Fauziah, 2019):

1. Fleksibilitas dalam pengembangan: Scrum memungkinkan tim untuk menyesuaikan perubahan kebutuhan atau prioritas secara cepat dan efektif.
2. Kolaborasi yang baik: Scrum mempromosikan kerja sama antara anggota tim, memperkuat tim, dan memastikan bahwa setiap anggota tim memiliki peran yang jelas.
3. Menghasilkan produk yang lebih baik: Scrum memungkinkan tim untuk menghasilkan produk yang lebih baik dengan fokus pada nilai bisnis yang diinginkan.
4. Responsif terhadap perubahan: Scrum memungkinkan tim untuk merespons perubahan yang terjadi pada produk atau lingkungan bisnis dengan lebih cepat.

Dalam konteks proyek, metode Scrum dipilih karena proyek tersebut melibatkan banyak pemangku kepentingan dan memiliki kebutuhan yang terus berubah, sehingga memerlukan pendekatan pengembangan perangkat lunak yang lebih fleksibel dan adaptif seperti Scrum.

## **Pengumpulan Data**

Data dikumpulkan dengan menggunakan teknik pengumpulan data sebagai berikut (Daryanto & Rahardjo, 2021):

1. Observasi

* Observasi langsung di kantin Telkom University untuk memperoleh informasi tentang kebutuhan sistem Point of Sales dan proses bisnis di kantin.
* Observasi melalui wawancara dengan beberapa pihak terkait di kantin, seperti owner kantin, dan mahasiswa.

1. Wawancara

Wawancara dilakukan dengan owner kantin, dan mahasiswa untuk memperoleh informasi detail tentang proses bisnis dan kebutuhan sistem Point of Sales di kantin.

1. Studi dokumentasi

Studi dokumentasi dilakukan untuk memperoleh informasi tentang kebutuhan bisnis dan teknologi yang diperlukan untuk pengembangan aplikasi Point of Sales berbasis website.

Data yang terkumpul kemudian dianalisis menggunakan teknik analisis data yang sesuai dengan jenis data yang dikumpulkan, seperti analisis kualitatif dan analisis kuantitatif. Hasil analisis data kemudian digunakan sebagai acuan dalam merancang sistem Point of Sales berbasis website yang sesuai dengan kebutuhan dan proses bisnis di kantin Telkom University.

## **Pengolahan Data**

Setelah data terkumpul, data kemudian diolah dengan menggunakan langkah-langkah sebagai berikut (Darmawan & Nugroho, 2021):

1. Pembersihan Data

Data yang diperoleh dari hasil pengumpulan data perlu dibersihkan dan dihapus data yang tidak relevan atau duplikat.

1. Analisis Data

Data dianalisis dengan teknik analisis data yang sesuai, seperti analisis deskriptif, analisis korelasi, dan analisis regresi.

1. Visualisasi Data

Hasil analisis data kemudian divisualisasikan dengan menggunakan grafik, tabel, dan diagram agar mudah dipahami dan diinterpretasikan. Interpretasi Data Hasil analisis data kemudian diinterpretasikan untuk mendapatkan pemahaman yang lebih dalam tentang data dan masalah yang sedang diteliti.

Data yang telah diolah kemudian digunakan untuk merancang sistem Point of Sales berbasis website yang sesuai dengan kebutuhan dan proses bisnis di kantin Telkom University. Selain itu, data juga digunakan untuk melakukan evaluasi terhadap aplikasi Point of Sales yang telah dikembangkan.

## **Metode Evaluasi**

Setelah aplikasi Point of Sales berbasis website pelanggan telah dikembangkan aplikasi tersebut kemudian dievaluasi menggunakan metode evaluasi sebagai berikut (Maulidia, & Santoso 2021):

1. User Acceptance Testing (UAT)

UAT dilakukan oleh pengguna yang telah ditentukan untuk menguji fungsionalitas dan kinerja aplikasi. Hasil dari UAT kemudian digunakan untuk menentukan apakah aplikasi siap digunakan secara langsung atau perlu dilakukan perbaikan. Performance

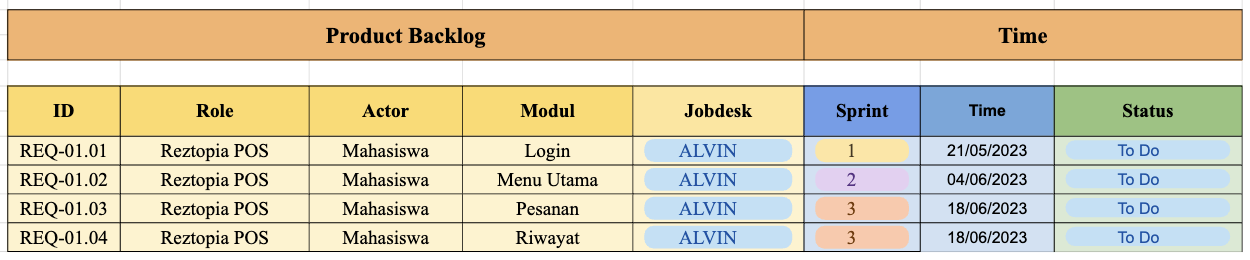
1. Testing Performance

testing dilakukan untuk menguji kemampuan aplikasi dalam menangani beban pengguna dan ketersediaan aplikasi saat digunakan secara bersamaan oleh banyak pengguna.

1. Usability Testing Usability

testing dilakukan untuk menguji sejauh mana aplikasi mudah digunakan oleh pengguna dan memenuhi kebutuhan pengguna.

## **Rencana Jadwal Kegiatan**

Sprint adalah periode waktu dalam metode Scrum dimana tim pengembang bekerja secara terfokus untuk menyelesaikan item backlog yang telah dipilih pada Sprint Planning (Utami & Pramono, 2020).

### Gambar III.03 Sprint

# Analisis dan Perancangan

## **Proses Bisnis**

### Gambar IV.01 Proses Bisnis Back Office Pada Kantin Telkom University

Diagram di atas menggambarkan proses bisnis untuk mahasiswa dalam melakukan transaksi di kantin Telkom University melalui aplikasi Reztopia. Mahasiswa memulai dengan melakukan scan barcode di meja kantin untuk mengidentifikasi diri mereka dan kemudian melakukan login atau registrasi ke aplikasi Reztopia. Setelah masuk, mereka dapat memilih tenant kantin, memilih menu makanan yang diinginkan, dan melihat keranjang pesanan mereka. Setelah memilih metode pembayaran, mereka dapat melihat detail pesanan mereka. Pada sisi tenant, mereka dapat melakukan login ke Reztopia BackOffice untuk melihat daftar pesanan yang masuk dan memberikan tanda bahwa pesanan telah selesai diproses.

## **Product Backlog**

1. Kebutuhan Fungsionalitas

| **ID** | **Apps** | **Actor** | **Epic** | **Story** | **User Story** |
| --- | --- | --- | --- | --- | --- |
|
| REQ-02.01 | BackOffice | Tenant | Login | Melakukan Login | Sebagai tenant, saya ingin dapat login ke dalam aplikasi agar dapat mengakses fungsionalitasnya. |
| REQ-02.02 | BackOffice | Tenant | Logout | Melakukan Logout | Sebagai tenant, saya ingin melakukan logout dari aplikasi BackOffice ketika saya selesai menggunakannya untuk menjaga keamanan akun saya. |
| REQ-02.03 | BackOffice | Tenant | Dashboard | Melihat Dashboard Tenant | Sebagai tenant, saya ingin melihat dashboard yang menyediakan fitur laporan penjualan dan secara visual dalam bentuk grafik dan juga daftar produk terlaris. |
| REQ-02.04 | BackOffice | Tenant | Menu | Mengelola Kategori | Sebagai tenant, saya ingin dapat membuat, mengubah, dan menghapus kategori pada halaman menu dalam aplikasi BackOffice. |
| REQ-02.05 | BackOffice | Tenant | Menu | Mengelola Menu | Sebagai tenant, saya ingin dapat membuat, mengubah, dan menghapus menu pada halaman menu dalam aplikasi BackOffice. |
| REQ-02.06 | BackOffice | Tenant | Laporan Penjualan  Tenant | Melihat Laporan Penjualan Tenant | Sebagai tenant, saya ingin dapat melihat rincian penjualan harian atau periode tertentu seperti nama pemesan, nomor meja, dan total pembayaran |
| REQ-02.07 | BackOffice | Tenant | Profil | Mengelola Profil Tenant | Sebagai tenant, saya ingin dapat mengatur dan memperbarui informasi profil saya seperti nama, BackOfficeisi, foto profi, dan kata sandi pada halaman profil. |
| REQ-02.10 | BackOffice | Admin | Login | Melakukan Login | Sebagai admin, saya ingin dapat login ke dalam aplikasi agar dapat mengakses fungsionalitasnya. |
| REQ-02.11 | BackOffice | Admin | Logout | Melakukan Logout | Sebagai admin, saya ingin melakukan logout dari aplikasi BackOffice ketika saya selesai menggunakannya untuk menjaga keamanan akun saya. |
| REQ-02.12 | BackOffice | Admin | Dashboard | Melihat Dashboard Admin | Sebagai admin, saya ingin melihat dashboard yang menyediakan fitur laporan penjualan pada pujasera, jumlah orderan, jumlah tenant yang aktif, jumlah tenant tidak aktif dan juga daftar tenant terlaris |
| REQ-02.13 | BackOffice | Admin | Tenant | Mengelola Akun Tenant | Sebagai admin, saya ingin mengatur data profil, menambahkan data tenant baru, dan mengatur tenant untuk aktif maupun non aktif |
| REQ-02.14 | BackOffice | Admin | Laporan Penjualan  Tenant | Melihat Laporan Penjualan Seluruh Tenant | Sebagai admin, saya ingin dapat melihat rincian penjualan harian atau periode tertentu seperti nama pemesan, nomor meja, dan total pembayaran dari setiap tenant |
| REQ-02.15 | BackOffice | Admin | Profil | Mengelola Profil Admin | Sebagai tenant, saya ingin dapat mengatur dan memperbarui informasi profil saya seperti BackOffice, foto profi, dan kata sandi pada halaman profil. |

### Tabel IV.01 Kebutuhan Fungsionalitas Reztopia Back Front Office

1. Kebutuhan Non Fungsionalitas

| **No** | **Aspek** | **Detail** | **Deskripsi** |
| --- | --- | --- | --- |
|
| 1 | Operasional | Portability | Aplikasi Reztopia harus dapat berfungsi dengan baik di berbagai browser web dan responsif untuk berbagai ukuran layar. |
| Maintainability | Aplikasi Reztopia harus mudah diperbarui dan diperbaiki. |
| Availability | Aplikasi Reztopia harus selalu tersedia untuk pengguna. |
| Scalability | Aplikasi Reztopia harus dapat menangani peningkatan beban penggunaan jika diperlukan. |
| 2 | Performa | Response Time | Aplikasi Reztopia harus dapat memberikan respon yang cepat kepada pengguna. |
| Efficiency | Aplikasi Reztopia menggunakan sumber daya sistem secara efisien. |
| Reliability | Aplikasi Reztopia berfungsi secara konsisten dan dapat diandalkan. |
| 3 | Keamanan | Data Protection | Aplikasi Reztopia melindungi data pengguna melalui enkripsi data |
| Access Control | Aplikasi Reztopia mengontrol akses berdasarkan peran pengguna. |

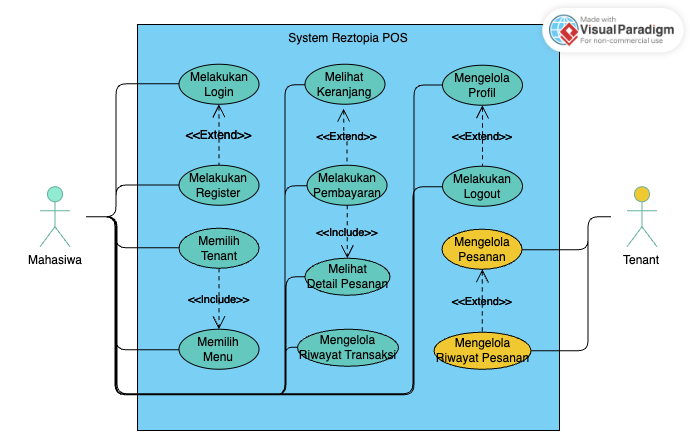
### Tabel IV.02 Kebutuhan Fungsionalitas Reztopia Fitur Back Office

1. Aktor

| **No** | **Aktor** | **Deskripsi** |
| --- | --- | --- |
|
| 1 | Mahasiswa | Mahasiswa Telkom University merupakan pengguna utama aplikasi Reztopia. Mereka dapat menggunakan aplikasi ini untuk membantu menghemat waktu dan meningkatkan kenyamanan dalam bertransaksi di kantin asrama Telkom University. |
| 2 | Tenant | Tenant merupakan pelaku usaha yang menjual makanan atau minuman yang beroperasi di kantin asrama Telkom University. Dengan bantuan aplikasi Reztopia, mereka mampu mengelola operasional usaha mereka dengan cara yang lebih efisien dan efektif. |

### Tabel IV.03 Aktor Reztopia Fitur Back Office

## **Use Case Diagram**



### Gambar IV.02 Use Case Diagram Reztopia Fitur Back Office

Use case diagram tersebut menunjukkan interaksi antara aktor "Mahasiswa" dan "Tenant" dalam aplikasi Reztopia POS dan Reztopia BackOffice. Mahasiswa dapat melakukan registrasi, login, memilih tenant, memilih menu, melihat keranjang pesanan, melakukan pembayaran, melihat detail pesanan, mengelola riwayat transaksi, mengelola profil, dan melakukan logout. Tenant dapat mengelola pesanan dan riwayat pesanan.

## **Use Case Scenario**

| ID | REQ-01.01 |
| --- | --- |
| Title | Melakukan Register di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa mendaftarkan akun di aplikasi Reztopia POS. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa berada di halaman pendaftaran Reztopia POS. |
| Postcondition | Mahasiswa berhasil membuat akun dan diberitahu bahwa pendaftaran berhasil. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Register" | 1. Menampilkan halaman pendaftaran |
| 1. Memasukkan data diri, data email dan password | 1. Memvalidasi data |
|  | 1. Jika data valid, membuat akun mahasiswa, dan menampilkan pesan sukses pendaftaran. Jika data tidak valid, menampilkan pesan error dan kembali ke halaman pendaftaran. |
| Extension | Validasi gagal:   1. Sistem menampilkan pesan error 2. Sistem kembali ke halaman pendaftaran |
| Frequency of use | Proses ini hanya dilakukan sekali oleh setiap mahasiswa. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.04 Use Case Scenario REQ-01.01

| ID | REQ-01.02 |
| --- | --- |
| Title | Melakukan Login di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa login ke aplikasi Reztopia POS. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah mendaftar dan berada di halaman login Reztopia POS. |
| Postcondition | Mahasiswa berhasil login dan dapat mengakses fungsionalitas aplikasi. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Login" | 1. Menampilkan halaman login |
| 1. Memasukkan data email dan password | 1. Memvalidasi data |
|  | 1. Jika data valid, memberikan akses ke aplikasi dan menampilkan halaman utama. Jika data tidak valid, menampilkan pesan error dan kembali ke halaman login. |
| Extension | Validasi gagal   1. Sistem menampilkan pesan error 2. Sistem kembali ke halaman login |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa ingin mengakses aplikasi. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.05 Use Case Scenario REQ-01.02

| ID | REQ-01.03 |
| --- | --- |
| Title | Memilih Tenant di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa memilih tenant yang diinginkan di aplikasi Reztopia POS.a |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah login dan berada di halaman utama Reztopia POS. |
| Postcondition | Mahasiswa berhasil memilih tenant dan dapat melihat informasi detail tenant tersebut. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Cari Tenant" | 1. Menampilkan halaman pencarian tenant |
| 1. Memasukkan kriteria pencarian dan memilih tenant yang diinginkan | 1. Menampilkan halaman tenant terpilih |
| Extension | - |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa ingin memilih tenant. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.06 Use Case Scenario REQ-01.03

| ID | REQ-01.04 |
| --- | --- |
| Title | Memilih Menu di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa memilih menu dari tenant yang telah dipilih sebelumnya di aplikasi Reztopia POS. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah memilih tenant dan berada di halaman menu tenant |
| Postcondition | Mahasiswa berhasil memilih menu dan dapat melihat detail menu tersebut. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih menu dari tenant terpilih | 1. Menampilkan daftar menu |
| 1. Memilih menu yang diinginkan | 1. Menampilkan detail menu terpilih |
| Extension | - |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa ingin memilih menu. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.07 Use Case Scenario REQ-01.04

| ID | REQ-01.05 |
| --- | --- |
| Title | Melihat Keranjang Pesanan di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa melihat keranjang pesanannya di aplikasi Reztopia POS. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah memilih menu dan menambahkannya ke keranjang. |
| Postcondition | Mahasiswa berhasil melihat keranjang pesanan dan dapat memeriksa pesanan sebelum melakukan checkout. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Lihat Keranjang" | 1. Menampilkan halaman keranjang pesanan |
| Extension | - |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa ingin memeriksa keranjang pesanannya sebelum checkout. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.08 Use Case Scenario REQ-01.05

| ID | REQ-01.06 |
| --- | --- |
| Title | Melakukan Pembayaran di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa melakukan pembayaran pesanannya di aplikasi Reztopia POS. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah memilih menu, menambahkannya ke keranjang dan memilih opsi checkout. |
| Postcondition | Mahasiswa berhasil melakukan pembayaran dan menerima konfirmasi dari sistem. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Checkout" | 1. Menampilkan halaman pembayaran |
| 1. Memilih metode pembayaran dan memasukkan data pembayaran | 1. Memvalidasi pembayaran dan jika pembayaran berhasil, menampilkan pesan sukses. Jika tidak berhasil, menampilkan pesan error dan kembali ke halaman pembayaran. |
| Extension | Pembayaran gagal   1. Sistem menampilkan pesan error 2. Sistem kembali ke halaman pembayaran |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa melakukan checkout. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.09 Use Case Scenario REQ-01.06

| ID | REQ-01.07 |
| --- | --- |
| Title | Melihat Detail Pesanan di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa melihat detail pesanannya di aplikasi Reztopia POS untuk memastikan pesanan yang akan diterima sesuai dengan yang diinginkan. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah melakukan pemesanan dan pembayaran. |
| Postcondition | Mahasiswa dapat melihat detail pesanan yang telah dipesan. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Lihat Pesanan" | 1. Menampilkan halaman detail pesanan |
| Extension | - |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa ingin melihat detail pesanan yang telah dilakukan. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.10 Use Case Scenario REQ-01.07

| ID | REQ-01.08 |
| --- | --- |
| Title | Mengelola Riwayat Transaksi di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa dapat mengelola riwayat transaksi di aplikasi Reztopia POS untuk melacak pesanan sebelumnya. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah melakukan pemesanan dan pembayaran sebelumnya. |
| Postcondition | Mahasiswa dapat melihat dan mengelola riwayat transaksi. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Riwayat Transaksi" | 1. Menampilkan halaman riwayat transaksi |
| 2. Melihat riwayat transaksi yang ingin dilihat detailnya | 1. Menampilkan detail transaksi terpilih |
| Extension | - |
| Frequency of use | Proses ini dilakukan setiap kali mahasiswa ingin melihat atau mengelola riwayat transaksi. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.11 Use Case Scenario REQ-01.08

| ID | REQ-01.09 |
| --- | --- |
| Title | Mengelola Profil di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa dapat mengelola profil di aplikasi Reztopia POS untuk memastikan data pribadi mereka sesuai. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah terdaftar dan masuk ke dalam sistem Reztopia POS. |
| Postcondition | Mahasiswa berhasil memperbarui profil mereka di aplikasi. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Masuk ke halaman profil | 1. Menampilkan halaman profil |
| 1. Melakukan perubahan profil |  |
| 1. Menyimpan perubahan | 1. Menampilkan halaman terbaru |
| Extension | - |
| Frequency of use | Proses ini dapat dilakukan oleh mahasiswa setiap kali mereka ingin memperbarui informasi pribadi mereka di profil. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.12 Use Case Scenario REQ-01.09

| ID | REQ-01.10 |
| --- | --- |
| Title | Melakukan Logout di Reztopia POS |
| Description | Use case ini menjelaskan bagaimana seorang mahasiswa dapat melakukan logout dari aplikasi Reztopia POS setelah selesai menggunakannya untuk menjaga keamanan akun mereka. |
| Primary Actor | Mahasiswa |
| Precondition | Mahasiswa telah masuk ke dalam sistem Reztopia POS. |
| Postcondition | Mahasiswa berhasil logout dan kembali ke halaman login. |
| Main Success Scenario | |
| Mahasiswa | Reztopia POS |
| 1. Memilih opsi "Logout" | 1. Melakukan proses logout |
|  | 1. Kembali ke halaman login |
| Extension | - |
| Frequency of use | Proses ini dapat dilakukan oleh mahasiswa setiap kali mereka selesai menggunakan aplikasi dan ingin menjaga keamanan akun mereka. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.13 Use Case Scenario REQ-01.10

| ID | REQ-02.08 |
| --- | --- |
| Title | Mengelola Pesanan di Reztopia BackOffice |
| Description | Use case ini menjelaskan bagaimana seorang tenant dapat mengelola pesanan dari mahasiswa di aplikasi Reztopia BackOffice untuk dapat memenuhi pesanan dengan efisien. |
| Primary Actor | Tenant |
| Precondition | Tenant telah masuk ke dalam sistem Reztopia BackOffice. |
| Postcondition | Tenant berhasil memperbarui status pesanan dan Reztopia BackOffice menampilkan pesanan yang telah selesai. |
| Main Success Scenario | |
| Tenant | Reztopia BackOffice |
| 1. Masuk ke Reztopia BackOffice |  |
| 1. Memilih sidebar POS | 1. Menampilkan daftar pesanan |
| 1. Melihat daftar pesanan |  |
| 1. Memperbarui status pesanan | 1. Menampilkan pesanan selesai |
| Extension | - |
| Frequency of use | Proses ini akan dilakukan oleh tenant setiap kali ada pesanan baru dari mahasiswa. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

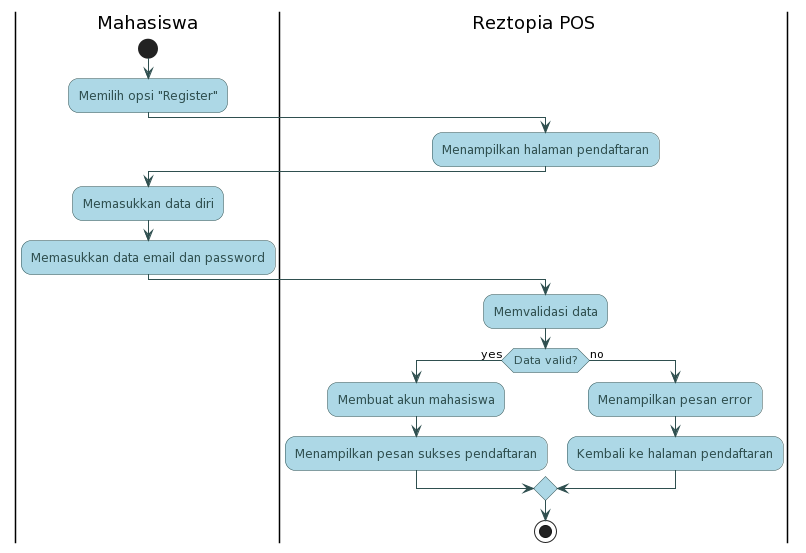
### Tabel IV.14 Use Case Scenario REQ-02.08

| ID | REQ-02.09 |
| --- | --- |
| Title | Mengelola Riwayat Pesanan di Reztopia BackOffice |
| Description | Use case ini menjelaskan bagaimana seorang tenant dapat mengelola riwayat pesanan di aplikasi Reztopia BackOffice untuk meninjau kembali pesanan yang sudah diselesaikan dalam satu hari. |
| Primary Actor | Tenant |
| Precondition | Tenant telah masuk ke dalam sistem Reztopia BackOffice. |
| Postcondition | Tenant berhasil melihat detail dari riwayat pesanan. |
| Main Success Scenario | |
| Tenant | Reztopia BackOffice |
| 1. Masuk ke Reztopia BackOffice |  |
| 1. Memilih sidebar POS | 1. Menampilkan daftar riwayat pesanan |
| 1. Memilih pesanan yang ingin dilihat detailnya | 1. Menampilkan detail pesanan |
| Extension | - |
| Frequency of use | Proses ini akan dilakukan oleh tenant setiap kali ingin meninjau kembali pesanan yang sudah diselesaikan dalam satu hari. |
| Status | Disetujui |
| Owner | Tim proyek |
| Priority | Tinggi |

### Tabel IV.15 Use Case Scenario REQ-02.09

## **Activity Diagram**

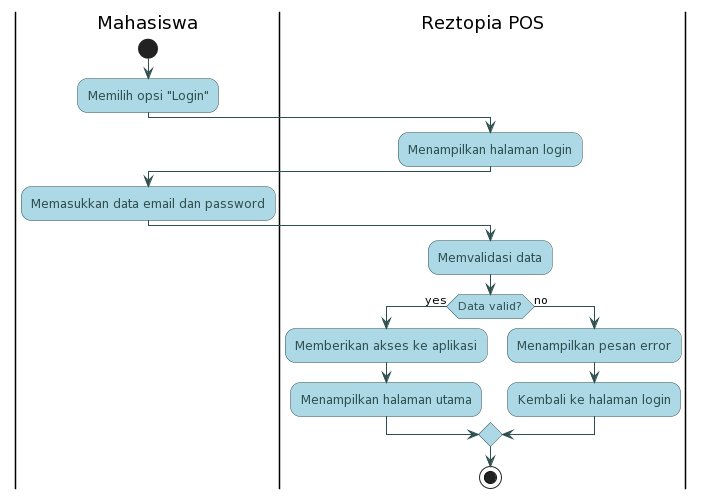
1. Melakukan Register



### Gambar IV.03 Activity Diagram Melakukan Register

Diagram ini secara umum menggambarkan aktivitas pendaftaran yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memilih opsi "Register" dan memasukkan data diri, email, dan password. Data tersebut kemudian divalidasi, dan jika valid, akun mahasiswa dibuat dan pesan sukses pendaftaran ditampilkan. Jika data tidak valid, pesan error ditampilkan dan mahasiswa kembali ke halaman pendaftaran.

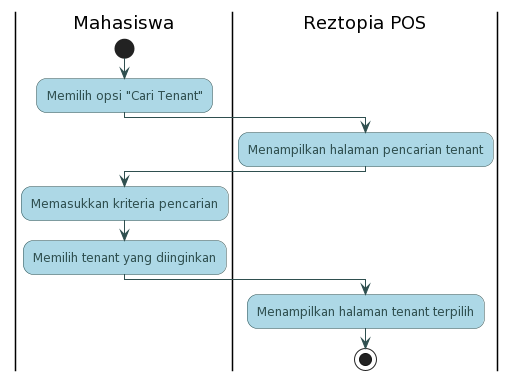
1. Melakukan Login



### Gambar IV.04 Activity Diagram Melakukan Login

Diagram ini secara umum menggambarkan aktivitas "Melakukan Login" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memilih opsi "Login" dan kemudian memasukkan data email dan password pada halaman login. Aplikasi "Reztopia POS" memvalidasi data yang dimasukkan dan jika data valid, mahasiswa diberikan akses ke aplikasi dan halaman utama ditampilkan. Namun, jika data tidak valid, aplikasi akan menampilkan pesan error dan mengarahkan mahasiswa kembali ke halaman login.

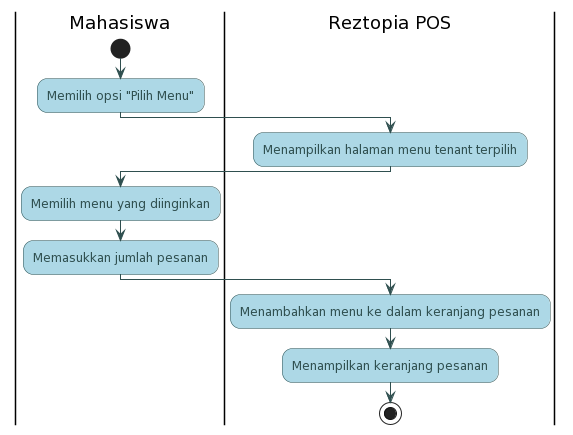
1. Memilih Tenant



### Gambar IV.05 Activity Diagram Memilih Tenant

Diagram ini secara umum menggambarkan aktivitas "Memilih Tenant" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai dengan mencari tenant dan memilih tenant yang diinginkan untuk memesan makanan atau minuman.

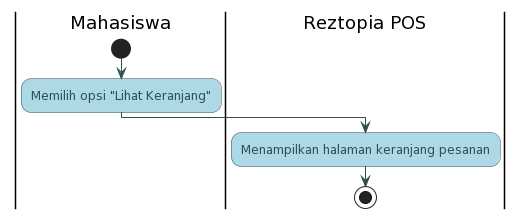
1. Memilih Menu



### Gambar IV.06 Activity Diagram Memilih Menu

Diagram ini secara umum menggambarkan aktivitas "Memilih Menu" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai dengan memilih menu dari tenant yang telah dipilih sebelumnya. Aplikasi "Reztopia POS" menampilkan halaman menu dengan pilihan menu yang tersedia. Mahasiswa kemudian memilih menu yang diinginkan dan menambahkan pesanan ke keranjang.

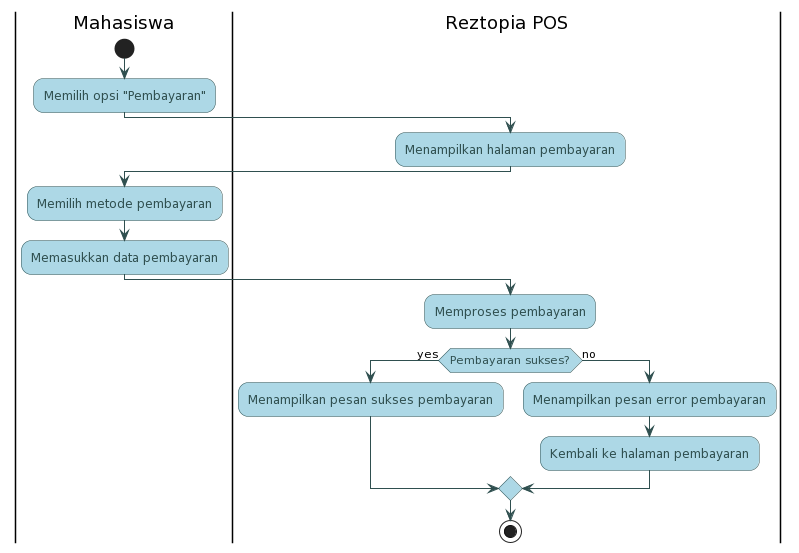
1. Melihat Keranjang



### Gambar IV.07 Activity Diagram Memilih Keranjang

Diagram ini secara umum menggambarkan aktivitas "Melihat Keranjang" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai proses dengan memilih opsi "Lihat Keranjang", dan aplikasi "Reztopia POS" menampilkan halaman keranjang pesanan. Proses berakhir setelah mahasiswa melihat keranjang pesanannya.

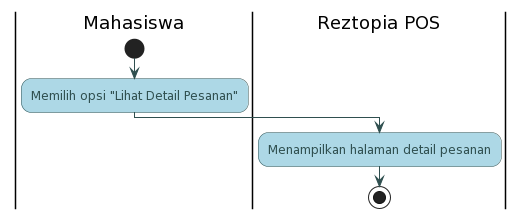
1. Melakukan Pembayaran



### Gambar IV.08 Activity Diagram Melakukan Pembayaran

Diagram ini secara umum menggambarkan aktivitas "Melakukan Pembayaran" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai proses dengan memilih opsi "Pembayaran", dan aplikasi "Reztopia POS" menampilkan halaman pembayaran. Selanjutnya, mahasiswa memilih metode pembayaran dan memasukkan data pembayaran. Aplikasi "Reztopia POS" akan memproses pembayaran tersebut. Jika pembayaran sukses, akan ditampilkan pesan sukses pembayaran. Namun, jika pembayaran gagal, akan ditampilkan pesan error pembayaran dan kembali ke halaman pembayaran. Proses berakhir setelah pembayaran selesai (stop).

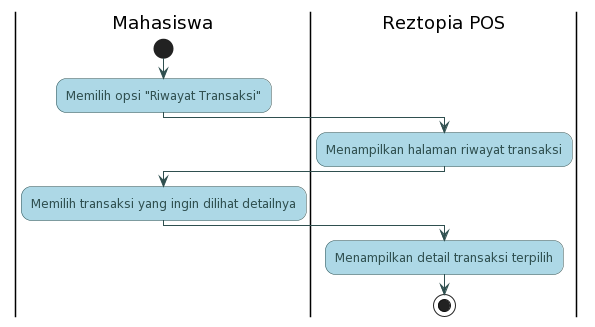
1. Melihat Detail Pesanan



### Gambar IV.09 Activity Diagram Detail Pesanan

Diagram ini secara umum menggambarkan aktivitas "Melihat Detail Pesanan" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai proses dengan memilih opsi "Lihat Detail Pesanan", dan aplikasi "Reztopia POS" menampilkan halaman dengan informasi detail mengenai pesanan yang telah dibuat. Proses berakhir setelah mahasiswa melihat detail pesanannya.

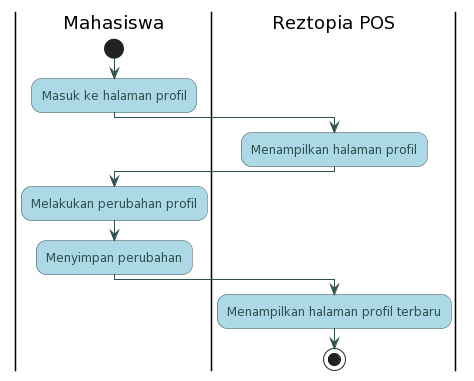
1. Mengelola Riwayat Transaksi



### Gambar IV.10 Activity Diagram Riwayat Transaksi

Diagram ini secara umum menggambarkan aktivitas "Mengelola Riwayat Transaksi" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai proses dengan memilih opsi "Riwayat Transaksi", dan aplikasi "Reztopia POS" menampilkan halaman riwayat transaksi yang telah dilakukan sebelumnya. Selanjutnya, mahasiswa memilih transaksi yang ingin dilihat detailnya, dan aplikasi menampilkan detail transaksi tersebut. Proses berakhir setelah mahasiswa melihat detail transaksi yang dipilih dari riwayat transaksi.

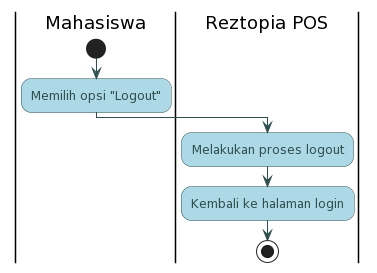
1. Mengelola Profil



### Gambar IV.11 Activity Diagram Mengelola Profil

Diagram ini secara umum menggambarkan aktivitas "Mengelola Profil" yang dilakukan oleh seorang mahasiswa di aplikasi "Reztopia POS". Mahasiswa memulai dengan masuk ke halaman profil, lalu aplikasi menampilkan halaman profil yang berisi informasi profil mahasiswa. Mahasiswa dapat melakukan perubahan pada profil dan menyimpan perubahan tersebut. Diagram ini membantu memvisualisasikan langkah-langkah yang terjadi dalam proses mengelola profil di aplikasi "Reztopia POS" oleh seorang mahasiswa.

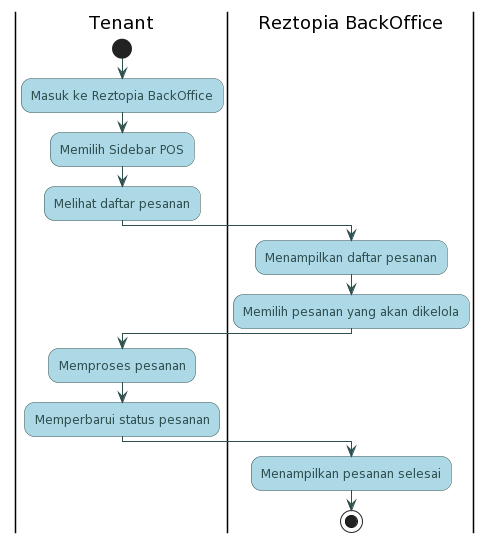
1. Melakukan Logout



### Gambar IV.12 Activity Diagram Melakukan Logout

Diagram ini secara umum menggambarkan aktivitas "Mengelola Pesanan" yang dilakukan oleh seorang tenant di aplikasi "Reztopia BackOffice". Tenant memulai dengan masuk ke Reztopia BackOffice dan memilih sidebar POS untuk melihat daftar pesanan. Selanjutnya, tenant melihat daftar pesanan dan memperbarui status pesanan yang sedang dikelola. Aplikasi Reztopia BackOffice akan menampilkan pesanan selesai setelah tenant telah memperbarui status pesanan tersebut. Proses berakhir setelah tenant selesai mengelola pesanan dan aplikasi menampilkan pesanan yang telah selesai.

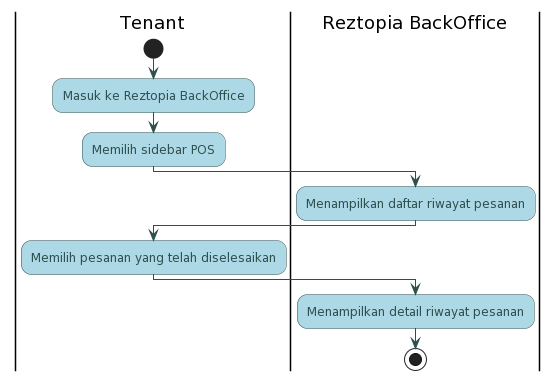
1. Mengelola Pesanan



### Gambar IV.13 Activity Diagram Mengelola Pesanan

Diagram ini secara umum menggambarkan aktivitas "Mengelola Pesanan" yang dilakukan oleh seorang tenant di aplikasi "Reztopia BackOffice". Tenant memulai proses dengan masuk ke Reztopia BackOffice dan memilih sidebar POS untuk melihat daftar pesanan. Selanjutnya, tenant memilih pesanan yang akan dikelola dan melakukan proses pengolahan pesanan serta memperbarui status pesanan tersebut. Aplikasi Reztopia BackOffice akan menampilkan pesanan selesai setelah tenant telah mengelola pesanan tersebut. Proses berakhir setelah tenant selesai mengelola pesanan dan aplikasi menampilkan pesanan yang telah selesai diolah.

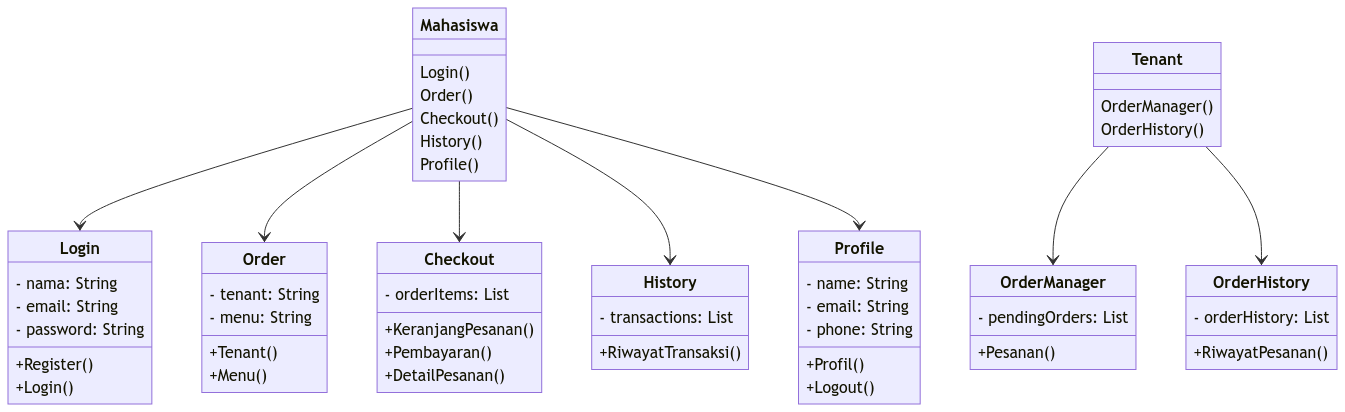
1. Mengelola Riwayat Pesanan



### Gambar IV.14 Activity Diagram Mengelola Riwayat Pesanan

Diagram ini secara umum menggambarkan aktivitas "Mengelola Riwayat Pesanan" yang dilakukan oleh seorang tenant di aplikasi "Reztopia BackOffice". Tenant memulai dengan masuk ke Reztopia BackOffice dan memilih sidebar POS. Aplikasi menampilkan daftar riwayat pesanan yang telah dilakukan, dan tenant dapat memilih pesanan yang telah diselesaikan untuk melihat detailnya. Diagram ini membantu memvisualisasikan langkah-langkah yang terjadi dalam proses mengelola riwayat pesanan di aplikasi "Reztopia BackOffice" oleh seorang tenant.

## **Class Diagram**

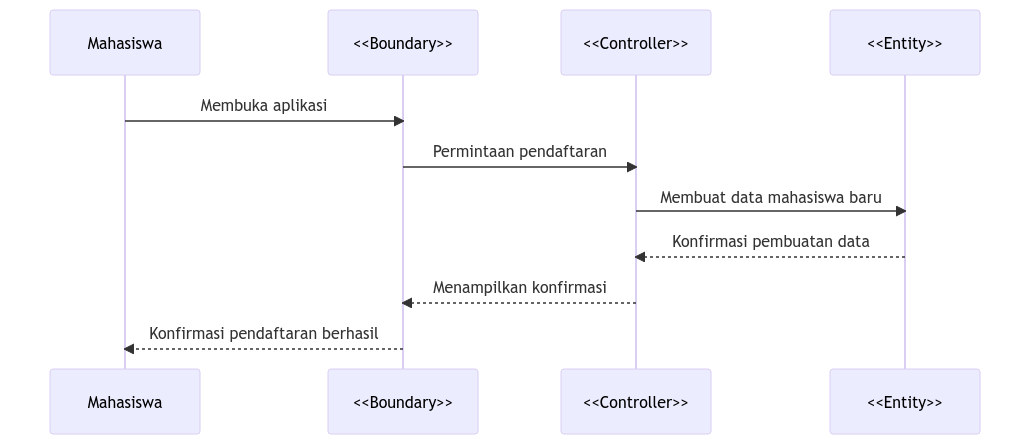


### Gambar IV.15 Class Diagram Reztopia Fitur Front Office

Diagram di atas merupakan representasi visual dari hubungan antara kelas-kelas dalam aplikasi Reztopia POS. Diagram ini menyajikan kelas-kelas seperti Login, Order, Checkout, History, Profile, OrderManager, dan OrderHistory beserta atribut-atribut dan operasi-operasi yang terkait dengan masing-masing kelas. Dengan menggunakan diagram ini, dapat dengan mudah dipahami bagaimana kelas-kelas tersebut saling berinteraksi dan berkontribusi dalam menjalankan fungsionalitas aplikasi Reztopia POS.

## **Sequence Diagram**

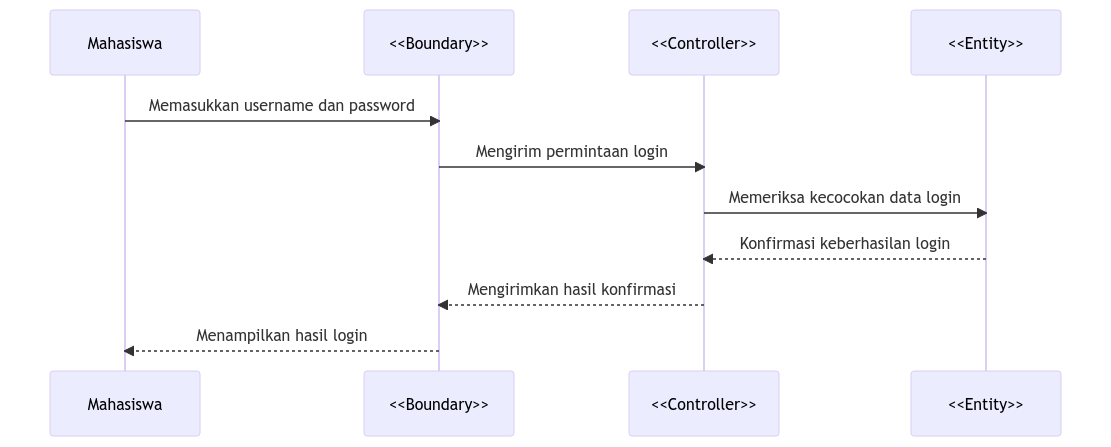
1. Melakukan Register



### Gambar IV.16 Sequence Diagram Melakukan Register

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa mengirim permintaan pendaftaran ke Boundary. Boundary melakukan validasi data dan meneruskannya ke Controller. Controller berkomunikasi dengan Entity untuk membuat akun mahasiswa baru, dan setelah berhasil, konfirmasi pendaftaran ditampilkan kepada Mahasiswa.

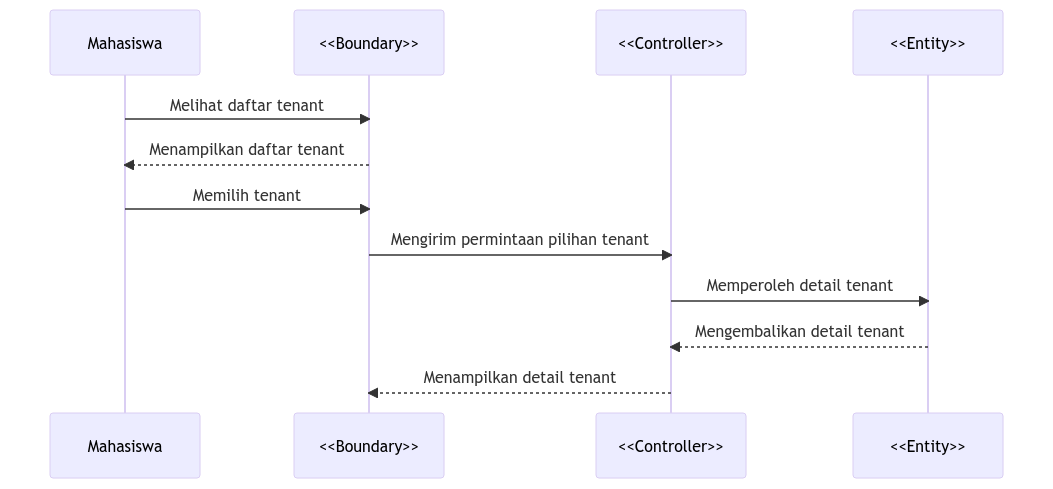
1. Melakukan Login



### Gambar IV.17 Sequence Diagram Melakukan Login

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa memasukkan username dan password ke dalam Boundary. Boundary mengirim permintaan login ke Controller. Controller berkomunikasi dengan Entity untuk memeriksa kecocokan data login, dan setelah berhasil, hasil konfirmasi dikirimkan kembali ke Boundary. Hasil login kemudian ditampilkan kepada Mahasiswa.

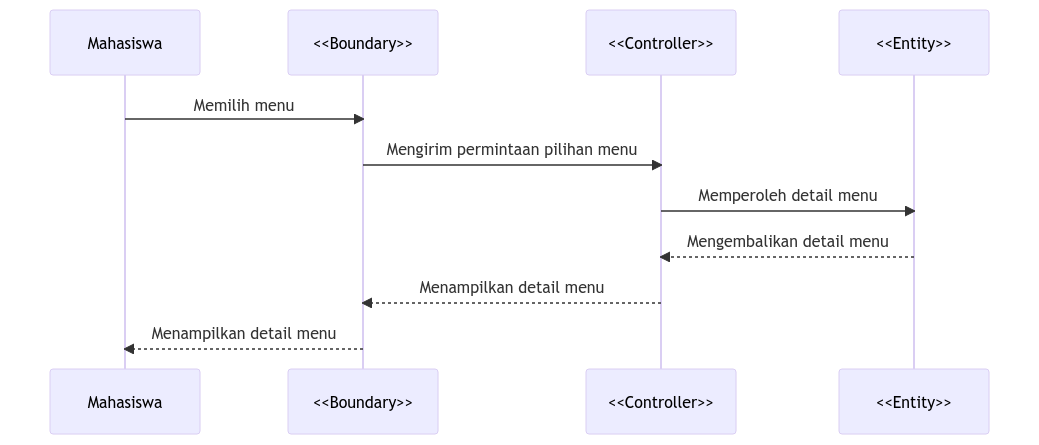
1. Memilih Tenant



### Gambar IV.18 Sequence Diagram Memilih Tenant

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa melihat daftar tenant yang tersedia melalui Boundary, yang kemudian menampilkan daftar tersebut kepada Mahasiswa. Mahasiswa memilih tenant yang diinginkan, dan Boundary mengirim permintaan pilihan tenant ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh detail tenant, yang nantinya ditampilkan oleh Boundary kepada Mahasiswa.

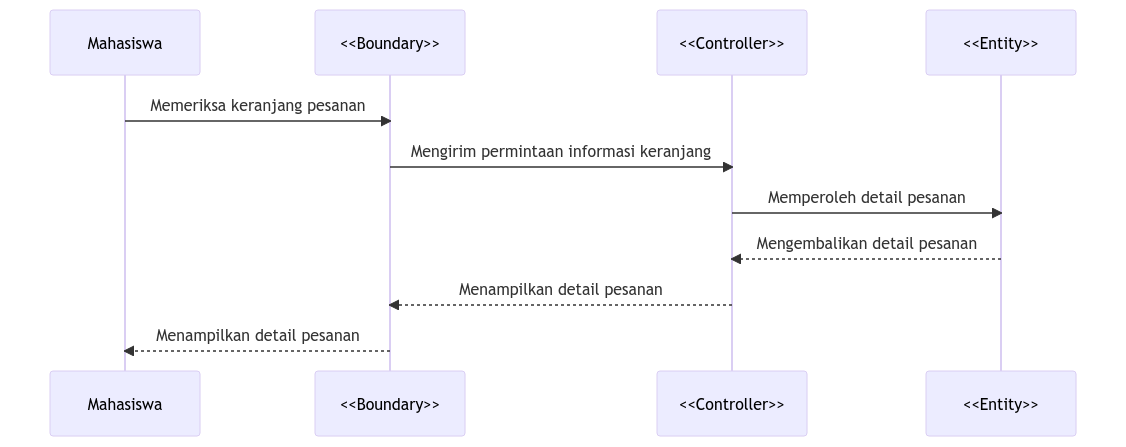
1. Memilih Menu



### Gambar IV.19 Sequence Diagram Memilih Menu

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa memilih menu dari tenant yang telah dipilih melalui Boundary. Boundary mengirim permintaan pilihan menu ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh detail menu, yang nantinya ditampilkan oleh Controller kepada Boundary. Detail menu kemudian ditampilkan oleh Boundary kepada Mahasiswa.

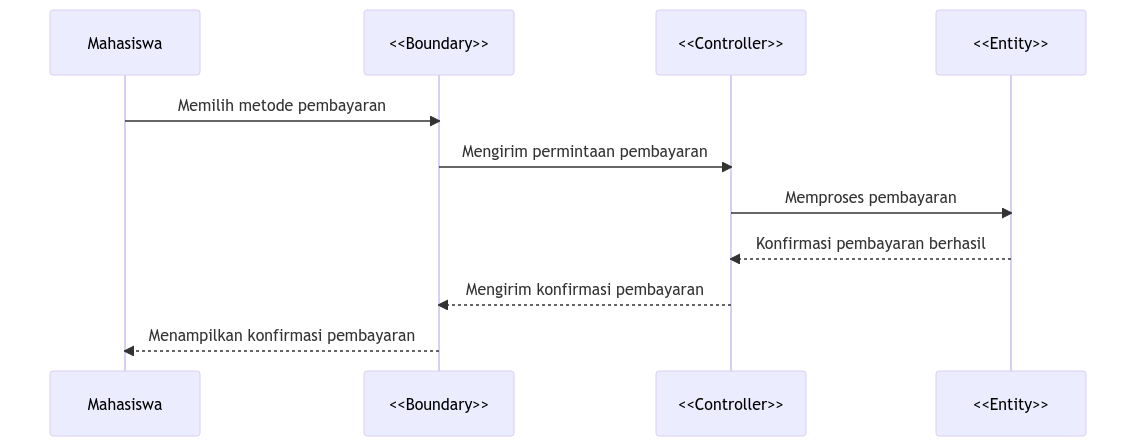
1. Melihat Keranjang



### Gambar IV.20 Sequence Diagram Melihat Keranjang

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa memeriksa keranjang pesanan di aplikasi Reztopia POS melalui Boundary. Boundary mengirim permintaan informasi keranjang ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh detail pesanan, yang nantinya ditampilkan oleh Controller kepada Boundary. Detail pesanan kemudian ditampilkan oleh Boundary kepada Mahasiswa.

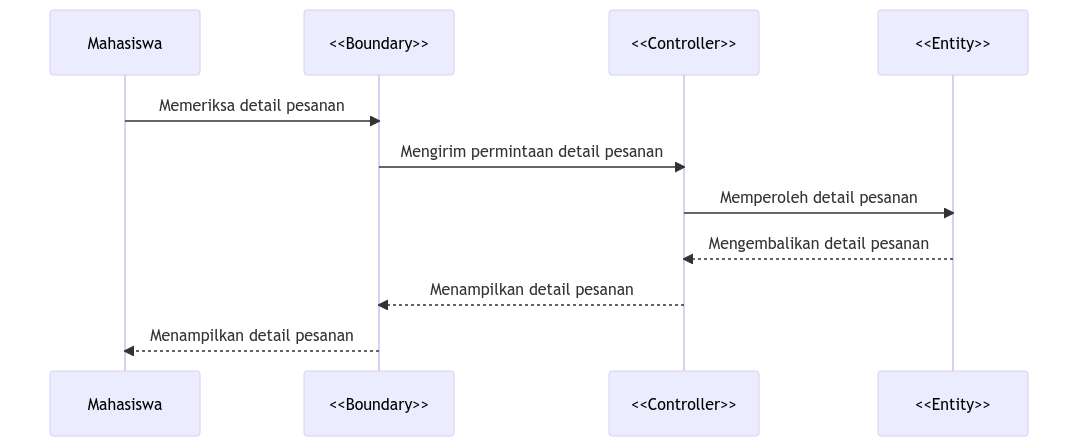
1. Melakukan Pembayaran



### Gambar IV.21 Sequence Diagram Melakukan Pembayaran

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa memilih metode pembayaran di aplikasi Reztopia POS melalui Boundary. Boundary mengirim permintaan pembayaran ke Controller. Controller berkomunikasi dengan Entity untuk memproses pembayaran, dan setelah berhasil, konfirmasi pembayaran dikirimkan oleh Controller kepada Boundary. Konfirmasi pembayaran kemudian ditampilkan oleh Boundary kepada Mahasiswa.

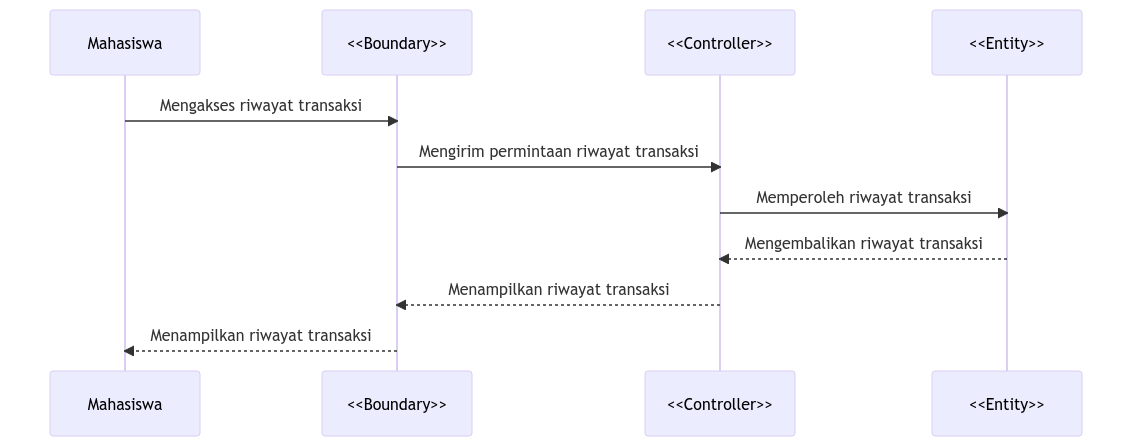
1. Melihat Detail Pesanan



### Gambar IV.22 Sequence Diagram Detail Pesanan

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa memeriksa detail pesanan di aplikasi Reztopia POS melalui Boundary. Boundary mengirim permintaan detail pesanan ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh detail pesanan, yang kemudian ditampilkan oleh Controller kepada Boundary. Detail pesanan ditampilkan oleh Boundary kepada Mahasiswa.

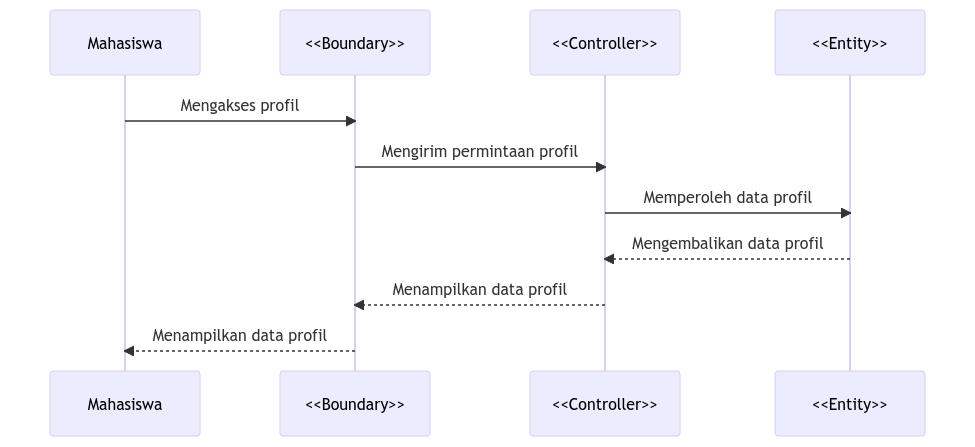
1. Mengelola Riwayat Transaksi



### Gambar IV.23 Sequence Diagram Mengelola Riwayat Transaksi

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa mengakses riwayat transaksi di aplikasi Reztopia POS melalui Boundary. Boundary mengirim permintaan riwayat transaksi ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh riwayat transaksi, yang kemudian ditampilkan oleh Controller kepada Boundary. Riwayat transaksi ditampilkan oleh Boundary kepada Mahasiswa.

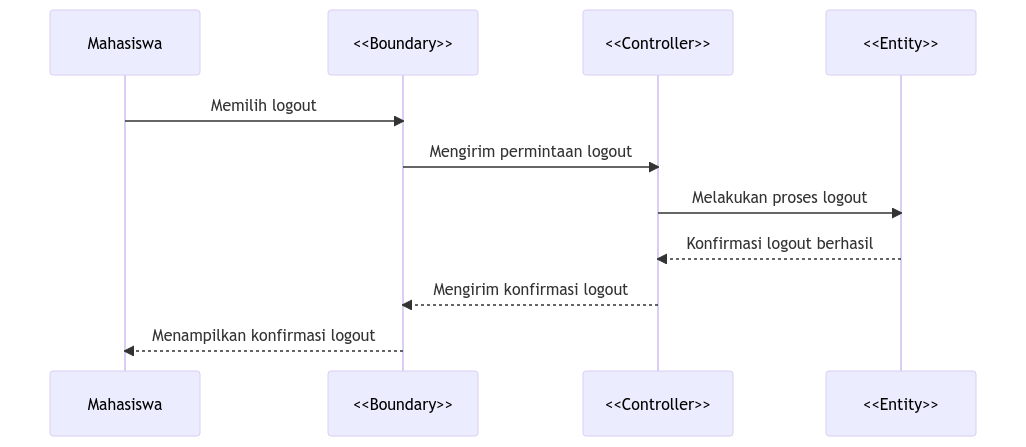
1. Mengelola Profil



### Gambar IV.24 Sequence Diagram Mengelola Profil

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa mengakses profilnya di aplikasi Reztopia POS melalui Boundary. Boundary mengirim permintaan profil ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh data profil, yang kemudian ditampilkan oleh Controller kepada Boundary. Data profil ditampilkan oleh Boundary kepada Mahasiswa.

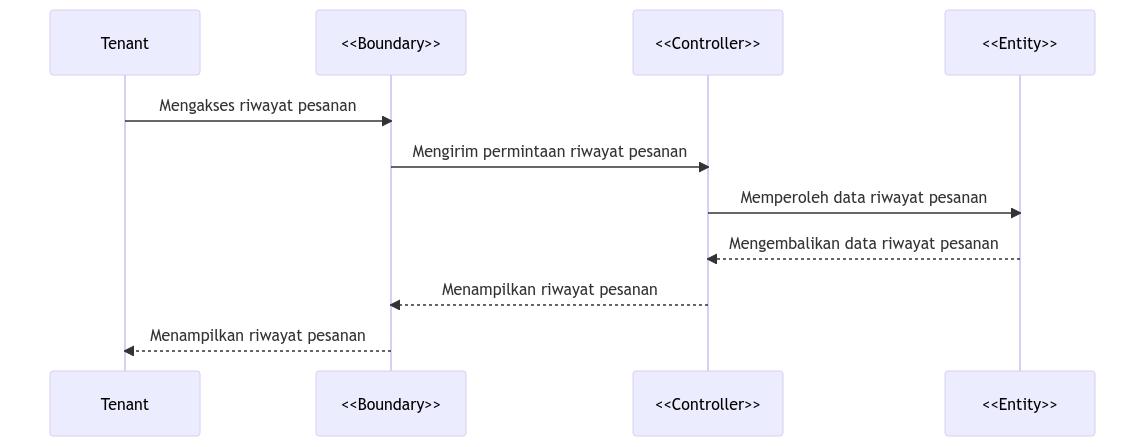
1. Melakukan Logout



### Gambar IV.25 Sequence Diagram Mengelola Logout

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Mahasiswa memilih logout di aplikasi Reztopia POS melalui Boundary. Boundary mengirim permintaan logout ke Controller. Controller berkomunikasi dengan Entity untuk melakukan proses logout, dan setelah berhasil, konfirmasi logout dikirimkan oleh Controller kepada Boundary. Konfirmasi logout kemudian ditampilkan oleh Boundary kepada Mahasiswa.

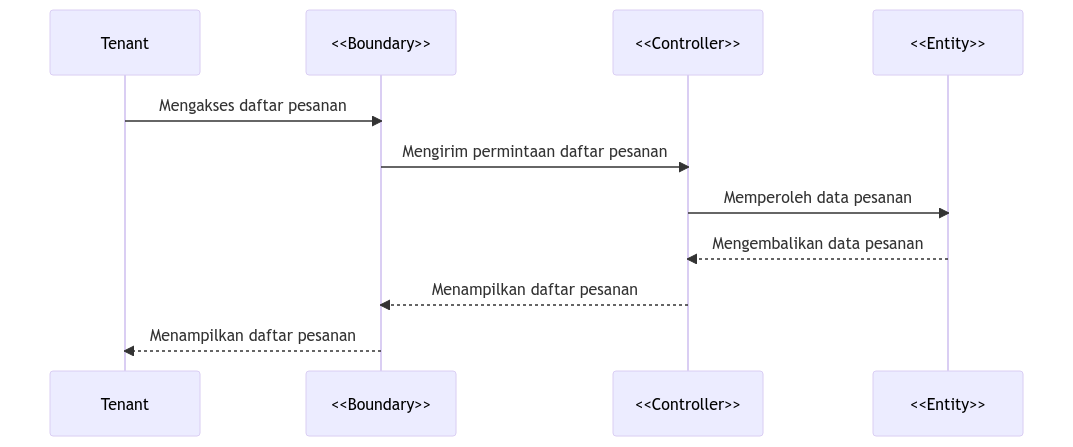
1. Mengelola Pesanan



### Gambar IV.26 Sequence Diagram Mengelola Pesanan

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Tenant mengakses daftar pesanan dari mahasiswa di aplikasi Reztopia BackOffice melalui Boundary. Boundary mengirim permintaan daftar pesanan ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh data pesanan, yang kemudian ditampilkan oleh Controller kepada Boundary. Daftar pesanan ditampilkan oleh Boundary kepada Tenant.

1. Mengelola Riwayat Pesanan



### Gambar IV.27 Sequence Diagram Riwayat Pesanan

Gambaran umum singkat mengenai diagram ini adalah sebagai berikut: Tenant mengakses riwayat pesanan di aplikasi Reztopia BackOffice melalui Boundary. Boundary mengirim permintaan riwayat pesanan ke Controller. Controller berkomunikasi dengan Entity untuk memperoleh data riwayat pesanan, yang kemudian ditampilkan oleh Controller kepada Boundary. Riwayat pesanan ditampilkan oleh Boundary kepada Tenant.

## **Entity Relationship Diagram**

### Gambar IV.28 Entity Relationship Diagram Fitur Front Office

Diagram entitas relasional di atas mencakup enam entitas utama dalam aplikasi Reztopia: Mahasiswa, Tenant, Pesanan, Menu, Transaksi, dan Riwayat. Mahasiswa dapat membuat Pesanan yang mencakup beberapa item Menu. Pesanan ini kemudian dibayar oleh Mahasiswa, yang menjadi Transaksi. Informasi tentang Transaksi ini kemudian disimpan dalam Riwayat, yang dapat dilihat oleh Mahasiswa dan Tenant. Tenant juga memiliki peran dalam menyediakan Menu yang bisa dipilih oleh Mahasiswa. Jadi, diagram ini menunjukkan bagaimana Mahasiswa dan Tenant berinteraksi dengan berbagai komponen dalam aplikasi, dari membuat pesanan hingga melakukan transaksi dan melacaknya dalam riwayat.

## **Relational Table Diagram**

### Gambar IV.29 Relational Table Diagram Fitur Front Office

Diagram diatas merupakan skema relasional untuk aplikasi Reztopia yang merinci struktur data dan hubungan antara entitas penting dalam aplikasi tersebut. Ada enam tabel utama: Users, Tenants, Orders, OrderDetails, OrderHistory, dan Profiles. Tabel Users berisi informasi dasar tentang pengguna aplikasi dan terhubung dengan tabel Profiles untuk informasi tambahan tentang pengguna. Tabel Orders mencatat pesanan yang dilakukan oleh pengguna dan terkait dengan Tenant yang menerima pesanan tersebut. OrderDetails adalah entitas anak dari Orders, mencatat detail masing-masing pesanan. OrderHistory juga terkait dengan Orders, mencatat riwayat status dari setiap pesanan. Diagram ini menunjukkan bagaimana data bergerak dan berinteraksi dalam sistem, memfasilitasi proses pengelolaan pesanan dan transaksi dalam aplikasi Reztopia.

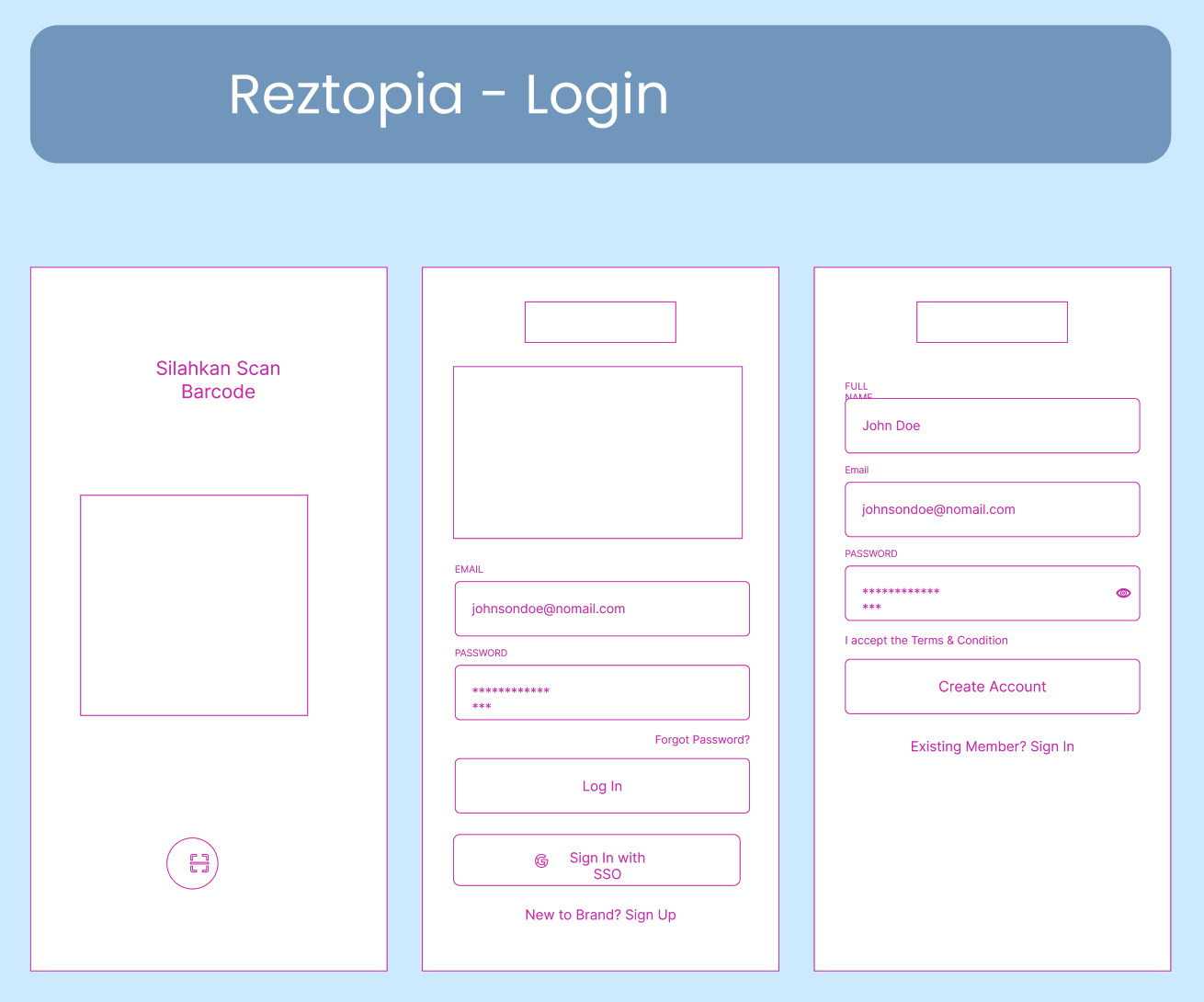
## **Deployment Diagram**

### Gambar IV.30 Deployment Diagram Fitur Back Office

Diagram deployment di atas menggambarkan bagaimana sistem aplikasi web Reztopia hosting dan diakses. Pengguna mengakses dua aplikasi yang berbeda, Reztopia POS dan Reztopia BackOffice, melalui peramban web di perangkat mereka. Kedua aplikasi ini di hosting pada server Reztopia dan berkomunikasi melalui RESTful yang menggunakan Postman untuk menguji dan mengelola layanan API. Semua data yang dihasilkan dan dikelola oleh aplikasi disimpan dalam sebuah database yang dioperasikan melalui phpMyAdmin. Ini menciptakan ekosistem terintegrasi dimana pengguna dapat mengakses dan berinteraksi dengan aplikasi melalui web, sementara semua data dan fungsionalitas dikelola dan di hosting pada server Reztopia.

## **Low Fidelity**

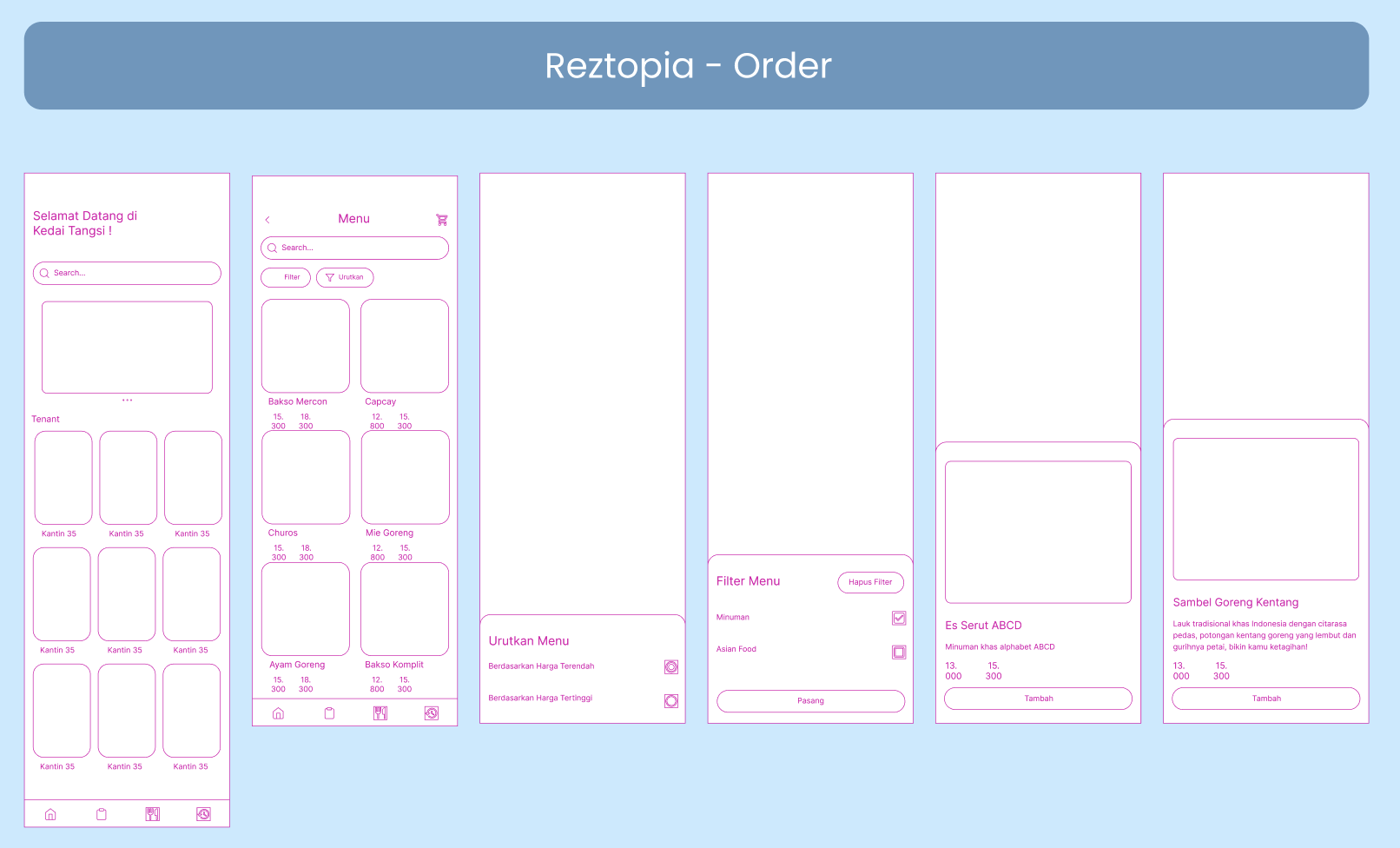
1. Reztopia POS - Login



### Gambar IV.31 Low Fidelity: Reztopia POS - Login

Merupakan tahapan awal dimana mahasiswa sebagai pengguna melakukan registrasi dan login ke aplikasi Reztopia POS. Ini merupakan langkah fundamental sebelum mahasiswa bisa memanfaatkan fungsi lain dalam aplikasi.

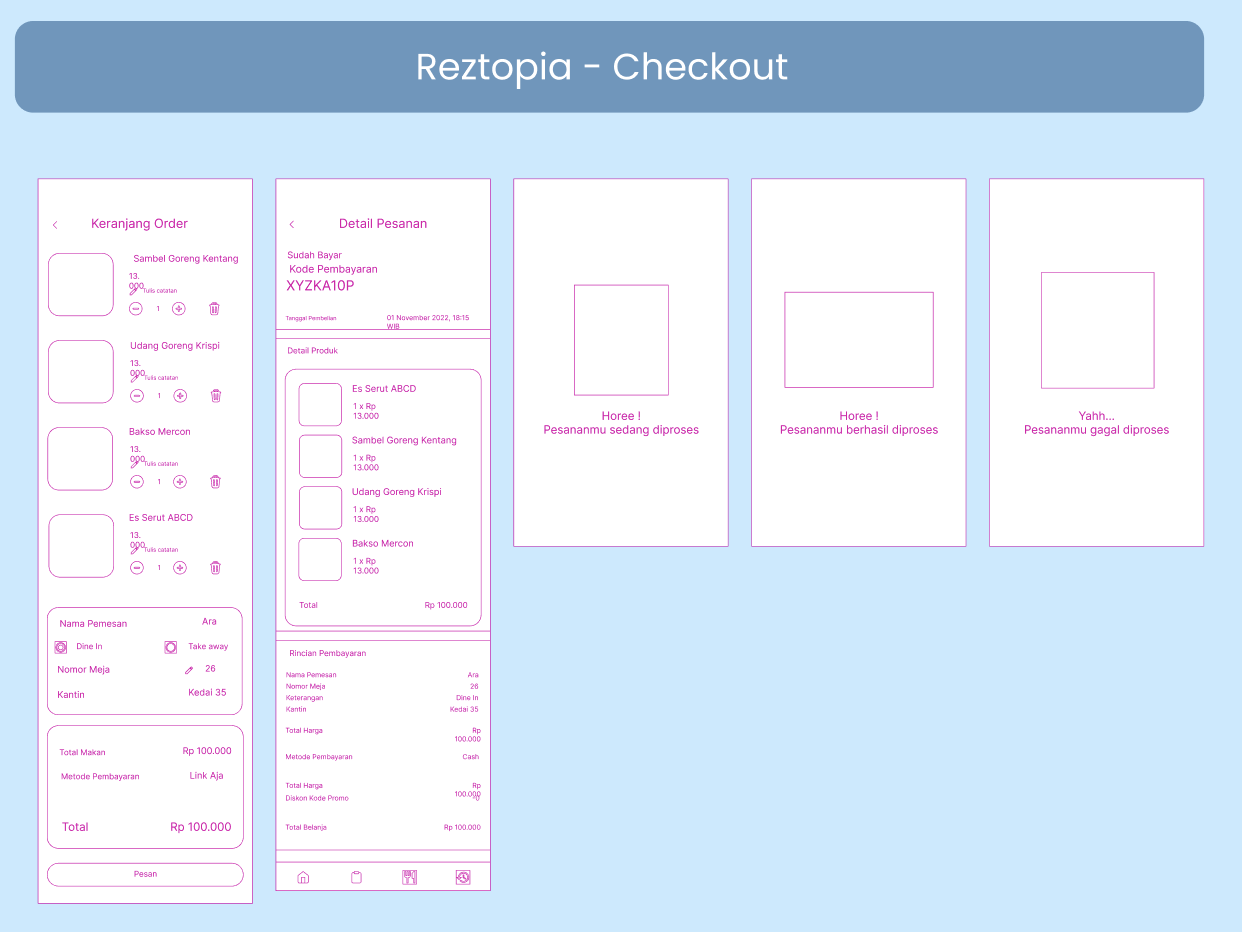
1. Reztopia POS - Order



### Gambar IV.32 Low Fidelity: Reztopia POS - Order

Setelah login, mahasiswa bisa memilih tenant dan menu yang diinginkan untuk membuat pesanan. Setiap tahapan dari proses ini dirancang untuk memudahkan mahasiswa dalam bertransaksi.

1. Reztopia POS - Checkout



### Gambar IV.33 Low Fidelity: Reztopia POS - Checkout

Setelah melakukan pemesanan, mahasiswa dapat melihat keranjang pesanan, melakukan pembayaran, dan memastikan detail pesanan sesuai dengan keinginan. Ini penting untuk meminimalkan kesalahan dan memastikan kepuasan pengguna.

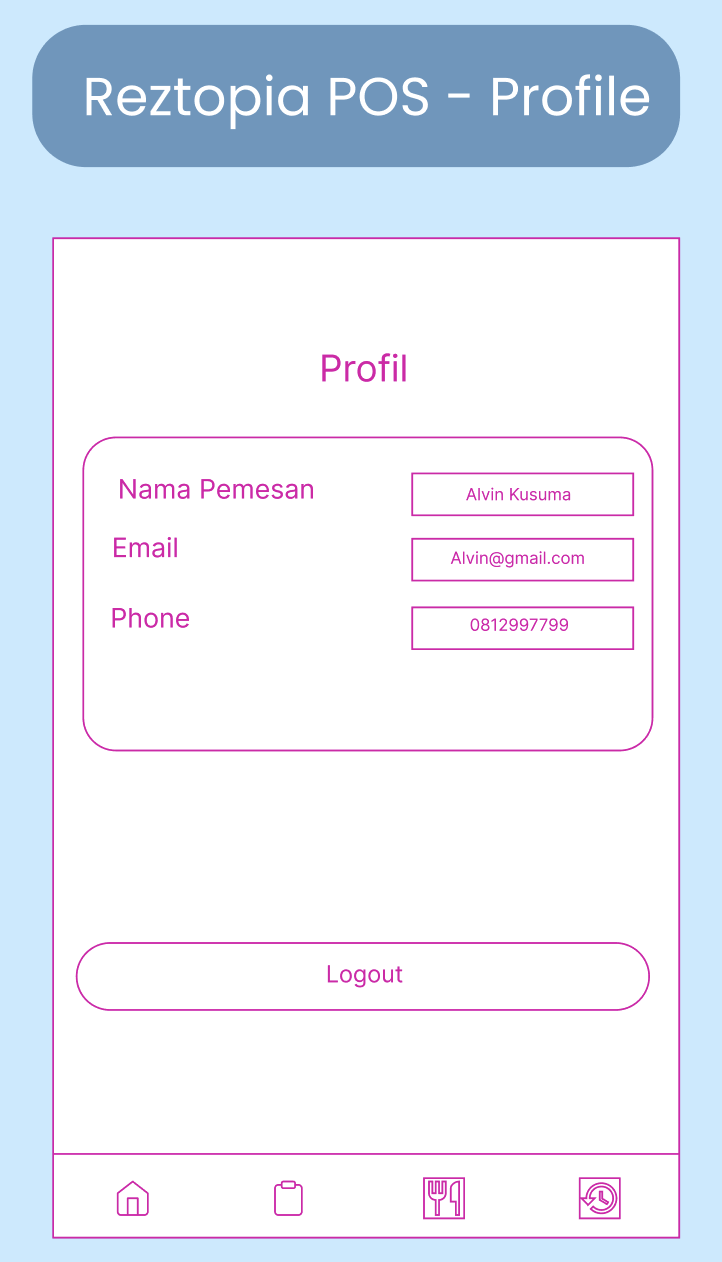
1. Reztopia POS - History



### Gambar IV.34 Low Fidelity: Reztopia POS - History

Mahasiswa dapat melacak riwayat transaksi mereka melalui fitur ini. Hal ini membantu mereka untuk memeriksa pesanan sebelumnya dan membantu dalam proses pengambilan keputusan untuk pesanan berikutnya.

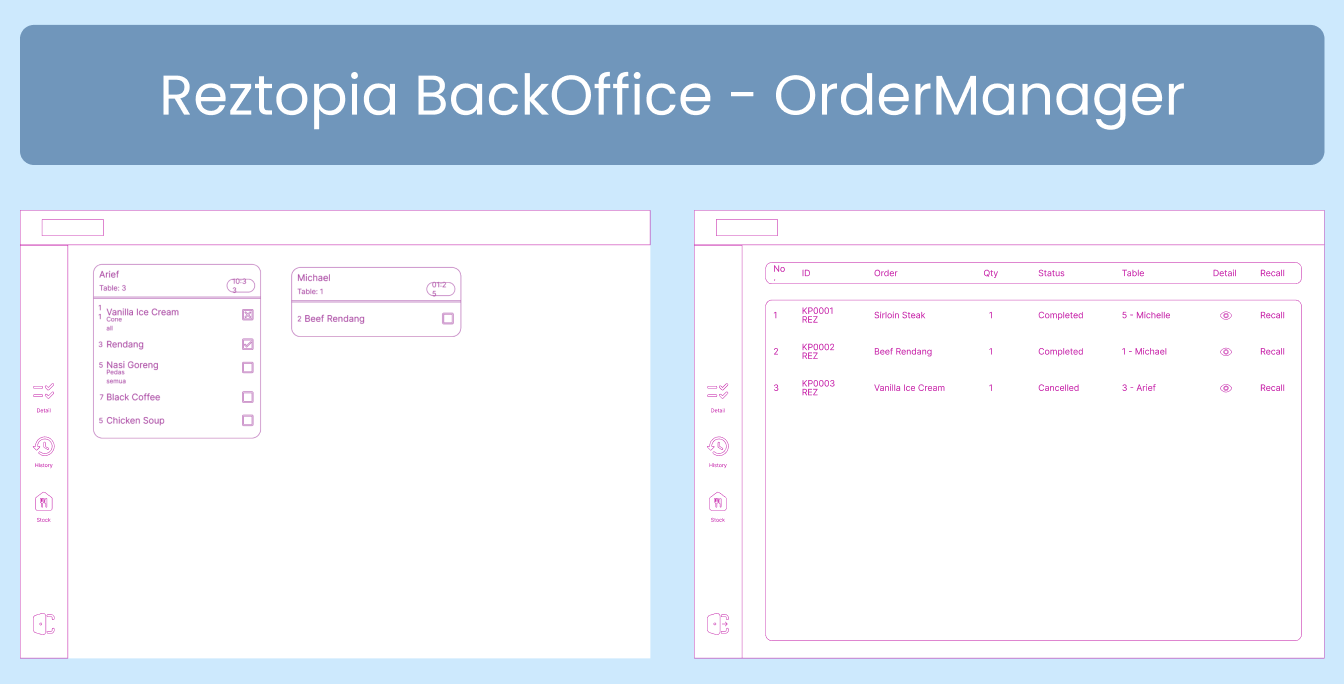
1. Reztopia POS - Profile



### Gambar IV.35 Low Fidelity: Reztopia POS - Profile

Mahasiswa memiliki akses untuk mengelola profil mereka, memastikan bahwa data pribadi mereka selalu up to date dan sesuai.

1. Reztopia BackOffice - OrderManager



### Gambar IV.36 Low Fidelity: Reztopia POS - OrderManager

Bagian ini dirancang khusus untuk tenant. Dengan fitur ini, tenant dapat mengelola pesanan dari mahasiswa dan melihat riwayat pesanan yang telah diselesaikan. Ini membantu tenant untuk bekerja lebih efisien dan memantau perkembangan usaha mereka.

# DAFTAR PUSTAKA

Putri, A. D., & Wulandari, A. (2020). Analisis Kepuasan Mahasiswa Terhadap Kualitas Pelayanan Kantin Asrama Universitas Telkom dengan Metode Importance Performance Analysis (IPA). Jurnal Ilmu Manajemen Dan Akuntansi Terapan, 11(2), 173-186.

Rizqi, A., & Wibisono, Y. (2018). Perancangan Aplikasi Point of Sales di Restoran untuk Menunjang Proses Transaksi. NeO-BIS (Ners and Electronics Oriented Business Information System), 2(1), 60-67.

Munawar, A. (2018). Analisis faktor-faktor yang mempengaruhi minat penggunaan aplikasi food delivery pada pengguna smartphone. Jurnal Pemikiran Sosiologi, 7(2), 123-132.

Arifianto, A. (2021). Implementasi Metode Scrum pada Proses Pengembangan Perangkat Lunak. Jurnal Sistem Informasi Bisnis, 11(1), 39-45.

Firdaus, M. F., & Taufik, A. (2020). Pemanfaatan Teknologi JavaScript dalam Pengembangan Aplikasi Web Interaktif. Jurnal Informatika Mulawarman, 15(1), 65-72.

Wibisono, A., & Setiawan, R. (2021). Penerapan React dalam Pengembangan Antarmuka Pengguna Aplikasi Web. Jurnal Ilmiah Teknologi Informasi Asia, 15(1), 15-21.

Ahmad Ramadhan. (2019). "Pengujian RESTful API Menggunakan Postman pada Aplikasi Mobile." Jurnal Informatika, 4(2), 69-74.

Fannani, Ilham. 2020. Perancangan Aplikasi POINT OF SALES Berbasis Website Togoide Studi Kasus UKM Laundry Menggunakan Metode Iterative dan Incremental.

Marom, N. (2020). Perancangan User Interface Pada Aplikasi Customer Layanan Pemesanan Makanan “EatAja” Menggunakan Metode User Centered Design.

Siddik, M. N. A. (2020). Perancangan User Interface Mitra Sebagai Penerima Pesanan Pada Aplikasi EATAJA Mitra Dengan Metode Goal-Directed Design

Pramono, R., & Utami, A. P. (2021). Pengembangan Model Konseptual pada Penelitian Sistem Informasi Berbasis Design Science. Jurnal Ilmiah Teknologi Informasi Terapan (JITTER), 3(1), 24-31.

Utami, A. P., & Pramono, R. (2020). Analisis Perencanaan Awal dalam Metode Scrum pada Pengembangan Perangkat Lunak. Jurnal Sistem Informasi Bisnis, 10(2), 101-108.

Septiandani, D., & Fauziah, N. (2019). Analisis Kelebihan Metode Scrum dalam Pengembangan Perangkat Lunak. Jurnal Teknik Informatika, 12(2), 176-184.

Daryanto, A., & Rahardjo, S. (2021). Pemanfaatan Teknik Pengumpulan Data dalam Riset Kualitatif. Jurnal Ilmiah Ilmu Administrasi Publik, 10(1), 35-47.

Darmawan, I., & Nugroho, A. (2021). Pengolahan Data dan Analisis Kuantitatif dalam Penelitian Sosial. Jurnal Sosiologi dan Pendidikan, 17(1), 51-64.

Maulidia, E., & Santoso, H. B. (2021). Evaluasi Aplikasi Berbasis Website menggunakan Metode User Acceptance Testing. Jurnal Sistem Informasi, 17(1), 22-29.