

Nama : M. Armand Giovani
NRP : 5025211054
Kelas : PPL A

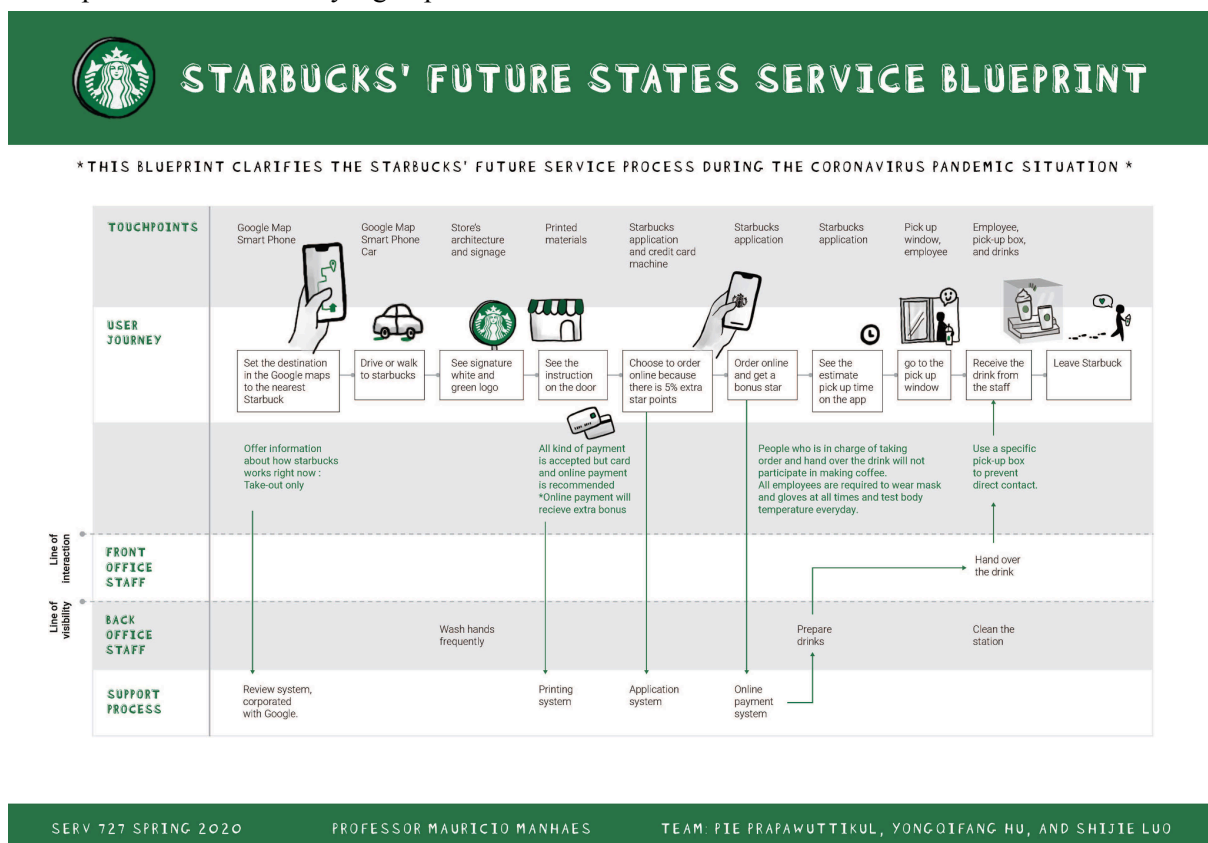
Evaluasi Akhir Semester

Soal EAS :

1. Deskripsikan model bisnis yang dipilih?
2. Identifikasi perangkat lunak apa saja yang diperlukan, kemudian jelaskan alasannya?
3. Buat rancangan perangkat lunaknya ?
4. Lengkapi dengan desain database dan desain UI
5. Buat Presentasi dan demo hasil rancangan Perangkat Lunak, kemudian upload di Youtube, dan isi lembar monitoring?

Jawaban EAS :

1. Deskripsikan Model Bisnis yang Dipilih



Model bisnis yang dipilih adalah model bisnis pelayanan pelanggan di masa pandemi COVID-19 oleh Starbucks. Model ini menekankan pada pelayanan take-out dan pengurangan kontak fisik untuk meminimalisir penyebaran virus. Berikut adalah beberapa poin utama dari model bisnis ini:

- Penggunaan Teknologi: Penggunaan aplikasi ponsel pintar dan Google Maps untuk memandu pelanggan ke lokasi Starbucks terdekat.

- Pemesanan Online: Pelanggan dianjurkan untuk memesan secara online dan melakukan pembayaran melalui aplikasi untuk mendapatkan poin tambahan.
- Pengambilan Pesanan: Pesanan diambil di jendela pick-up khusus tanpa harus memasuki toko, mengurangi interaksi langsung antara pelanggan dan staf.
- Protokol Kesehatan: Semua staf diharuskan menggunakan masker, sarung tangan, dan memeriksa suhu tubuh setiap hari untuk memastikan kesehatan dan keselamatan.

2. Identifikasi Perangkat Lunak yang Diperlukan dan Alasan

Berikut adalah perangkat lunak yang diperlukan untuk mendukung model bisnis tersebut:

- Google Maps API: Untuk membantu pelanggan menemukan lokasi Starbucks terdekat.
 - Alasan: Memudahkan pelanggan dalam menemukan lokasi terdekat tanpa harus mencari secara manual.
- Aplikasi Starbucks: Untuk pemesanan online dan pembayaran.
 - Alasan: Mengurangi kontak fisik dengan memungkinkan pelanggan memesan dan membayar secara digital. Juga memberikan insentif berupa poin tambahan.
- Sistem Pembayaran Online: Terintegrasi dengan aplikasi untuk memproses pembayaran secara aman.
 - Alasan: Memastikan transaksi keuangan berjalan lancar dan aman tanpa harus menggunakan uang tunai atau kartu fisik.
- Sistem Manajemen Informasi: Untuk mengelola informasi pesanan dan status pengiriman.
 - Alasan: Memastikan pesanan dikelola dengan baik dan pelanggan mendapatkan informasi yang akurat mengenai status pesanan mereka.

3. Rancangan Perangkat Lunak

Kebutuhan Fungsional

- Pencarian Lokasi Terdekat:
Sistem harus dapat menampilkan lokasi Starbucks terdekat menggunakan Google Maps. Pengguna dapat memasukkan alamat atau menggunakan lokasi saat ini untuk menemukan Starbucks terdekat.
- Pemesanan Online:
Aplikasi harus menyediakan fitur untuk memesan minuman dan makanan secara online. Pengguna dapat memilih item dari menu, menambahkan ke keranjang, dan menyelesaikan pemesanan.
- Pembayaran Digital:
Sistem harus mendukung pembayaran melalui berbagai metode pembayaran digital (kartu kredit, e-wallet). Pembayaran harus diproses dengan aman dan cepat.
- Informasi Status Pesanan:
Aplikasi harus memberikan informasi real-time mengenai status pesanan (sedang diproses, siap diambil). Pengguna dapat melihat perkiraan waktu pengambilan pesanan.
- Poin dan Bonus:
Sistem harus memberikan poin tambahan untuk setiap pemesanan online. Pengguna dapat melihat dan menukarkan poin yang mereka miliki.
- Protokol Kesehatan:
Sistem harus mencatat dan melaporkan suhu tubuh dan penggunaan masker oleh staf setiap hari. Informasi ini harus tersedia bagi manajemen untuk memastikan kepatuhan terhadap protokol kesehatan.

Kebutuhan Nonfungsional

1. Keamanan:
Sistem harus memastikan semua transaksi pembayaran dienkripsi dan dilindungi dari ancaman keamanan. Data pelanggan harus disimpan dengan aman dan sesuai dengan peraturan privasi.
2. Kinerja:
Aplikasi harus dapat menangani sejumlah besar pengguna secara simultan tanpa penurunan kinerja. Sistem harus memberikan respon cepat untuk setiap permintaan pengguna.
3. Skalabilitas:
Sistem harus dapat dengan mudah diskalakan untuk mendukung penambahan pengguna dan pesanan. Infrastruktur harus mendukung peningkatan beban selama periode puncak (misalnya, liburan).
4. Ketersediaan:
Sistem harus memiliki uptime tinggi dengan minimal downtime. Aplikasi harus tersedia 24/7 untuk memungkinkan pengguna melakukan pemesanan kapan saja.
5. Usability:
Antarmuka pengguna harus intuitif dan mudah digunakan oleh semua pengguna. Pengguna baru harus dapat dengan cepat memahami cara menggunakan aplikasi.
6. Kompatibilitas:
Aplikasi harus kompatibel dengan berbagai perangkat dan sistem operasi (iOS, Android). Sistem harus dapat terintegrasi dengan layanan pihak ketiga seperti Google Maps dan payment gateway.

High-Level Design (HLD)

HLD berfokus pada gambaran umum dari sistem dan arsitektur perangkat lunak. Berikut adalah komponen utama dari HLD:

- a. User Interface (UI) Layer:
Komponen:
 - Aplikasi mobile (iOS, Android)
 - WebsiteFungsi:
 - Menyediakan antarmuka bagi pengguna untuk melakukan login/register, memesan produk, melakukan pembayaran, melihat status pesanan, dan mengelola profil.
- b. Application Layer:
Komponen:
 - Server aplikasi
 - APIFungsi:
 - Menangani proses autentikasi pengguna.
 - Mengelola alur pemesanan, pembayaran, dan manajemen pesanan.
 - Mengintegrasikan dengan sistem manajemen produk, protokol kesehatan, dan layanan pihak ketiga.
- c. Data Layer:
Komponen:
 - Database management system (DBMS)Fungsi:
 - Menyimpan dan mengelola data pelanggan, pesanan, produk, dan protokol kesehatan.

d. Integration Layer:

Komponen:

- Google Maps API
- Payment gateway API (Stripe, PayPal)

Fungsi:

- Mengintegrasikan layanan lokasi dari Google Maps.
- Mengintegrasikan layanan pembayaran digital dari payment gateway.

Low-Level Design (LLD)

LLD memberikan detail implementasi dari setiap komponen dalam HLD. Berikut adalah beberapa komponen LLD:

a. User Interface (UI) Layer:

Teknologi:

- Aplikasi mobile: React Native atau Flutter
- Website: React.js atau Angular

Struktur dan Alur:

- Rancang struktur navigasi dan alur antar layar (login, menu, keranjang, status pesanan, profil).
- Tentukan komponen UI yang akan digunakan untuk setiap layar (button, input field, card, dll.).

Desain:

- Buat wireframe dan mockup untuk setiap layar.
- Tentukan gaya visual, warna, dan tipografi yang sesuai dengan branding Starbucks.

b. Application Layer:

Teknologi Server-side:

- Node.js dengan framework Express.js
- Java dengan framework Spring Boot

Struktur API:

- Rancang endpoint API untuk operasi CRUD (Create, Read, Update, Delete) pada entitas (pelanggan, pesanan, produk, protokol kesehatan).
- Implementasikan logika bisnis pada API, seperti autentikasi, manajemen pesanan, dan manajemen poin.

Pola Desain:

- Terapkan pola desain MVC (Model-View-Controller) atau MVVM (Model-View-ViewModel) untuk memisahkan concerns.
- Gunakan prinsip Clean Architecture untuk memastikan modularitas dan testabilitas.

c. Data Layer:

DBMS:

- Gunakan MySQL atau PostgreSQL sebagai database relasional.

Skema Database:

- Berdasarkan entitas yang telah didefinisikan sebelumnya (pelanggan, pesanan, produk, protokol kesehatan).
- Tentukan relasi antar tabel dan constraints yang sesuai.

ORM:

- Gunakan Hibernate atau Sequelize sebagai ORM untuk mempermudah interaksi dengan database.

d. Integration Layer:

Google Maps API:

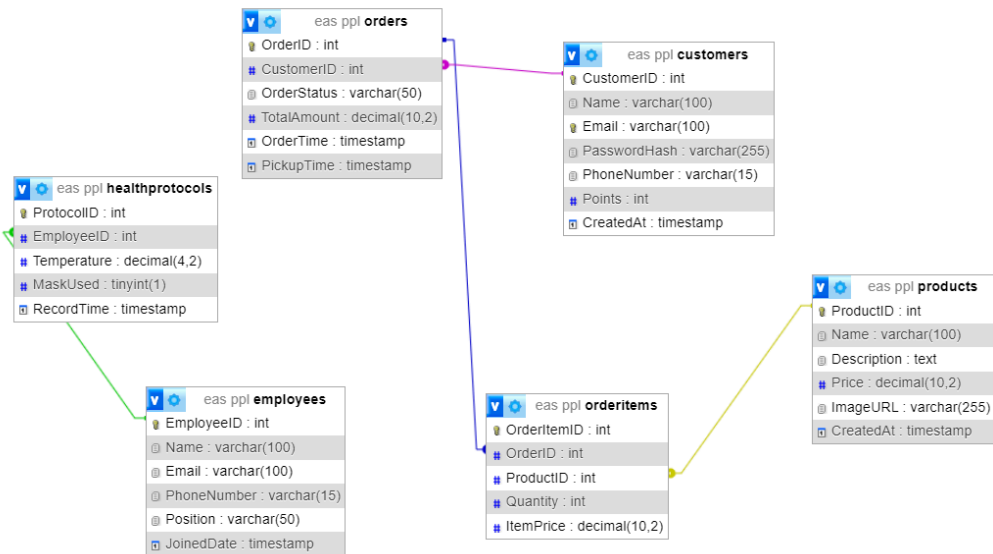
- Integrasikan API Google Maps untuk menampilkan lokasi Starbucks terdekat.
- Gunakan fitur geocoding untuk mendapatkan koordinat berdasarkan alamat.

Payment Gateway API:

- Integrasikan dengan Stripe atau PayPal, mobile banking untuk memproses pembayaran secara aman.

4. Lengkapi dengan desain database dan desain UI

Desain Database :



Tabel yang ada pada desain database untuk aplikasi pemesanan Starbucks:

1. Tabel Customers (Pelanggan)

- 'CustomerID': Primary key, auto-increment integer untuk mengidentifikasi setiap pelanggan secara unik.
- 'Name': Nama lengkap pelanggan.
- 'Email': Alamat email pelanggan, digunakan sebagai identifikasi unik.
- 'PasswordHash': Hash dari password yang digunakan pelanggan untuk login.
- 'PhoneNumber': Nomor telepon pelanggan.
- 'Points': Jumlah poin yang dimiliki oleh pelanggan.
- 'CreatedAt': Timestamp saat akun pelanggan dibuat.

2. Tabel Orders (Pesanan)

- 'OrderID': Primary key, auto-increment integer untuk mengidentifikasi setiap pesanan secara unik.
- 'CustomerID': Foreign key yang menghubungkan pesanan dengan pelanggan yang membuat pesanan.
- 'OrderStatus': Status terkini dari pesanan (misalnya: "Diproses", "Siap Diambil", "Selesai").
- 'TotalAmount': Total harga dari semua item yang dipesan.
- 'OrderTime': Timestamp saat pesanan dibuat.

- `PickupTime`: Timestamp saat pesanan siap untuk diambil.

3. Tabel OrderItems (Item Pesanan)

- `OrderItemID`: Primary key, auto-increment integer untuk mengidentifikasi setiap item pesanan secara unik.
- `OrderID`: Foreign key yang menghubungkan item pesanan dengan pesanan yang terkait.
- `ProductID`: Foreign key yang menghubungkan item pesanan dengan produk yang dipesan.
- `Quantity`: Jumlah produk yang dipesan.
- `ItemPrice`: Harga per item produk yang dipesan.

4. Tabel Products (Produk)

- `ProductID`: Primary key, auto-increment integer untuk mengidentifikasi setiap produk secara unik.
- `Name`: Nama produk.
- `Description`: Deskripsi produk.
- `Price`: Harga per unit produk.
- `ImageURL`: URL gambar produk.
- `CreatedAt`: Timestamp saat produk ditambahkan ke sistem.

5. Tabel HealthProtocols (Protokol Kesehatan)

- `ProtocolID`: Primary key, auto-increment integer untuk mengidentifikasi setiap catatan protokol kesehatan secara unik.
- `EmployeeID`: Foreign key yang menghubungkan catatan protokol dengan karyawan yang bersangkutan.
- `Temperature`: Suhu tubuh karyawan pada saat pencatatan.
- `MaskUsed`: Indikator apakah karyawan menggunakan masker atau tidak.
- `RecordTime`: Timestamp saat catatan protokol kesehatan dibuat.

6. Tabel Employees (Karyawan)

- `EmployeeID`: Primary key, auto-increment integer untuk mengidentifikasi setiap karyawan secara unik.
- `Name`: Nama lengkap karyawan.
- `Email`: Alamat email karyawan.
- `PhoneNumber`: Nomor telepon karyawan.
- `Position`: Posisi atau jabatan karyawan.
- `JoinedDate`: Timestamp saat karyawan bergabung dengan perusahaan.

Berikut kode sqlnya :

```
-- Tabel Pelanggan
CREATE TABLE Customers (
    CustomerID INT AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(100),
    Email VARCHAR(100) UNIQUE,
    PasswordHash VARCHAR(255),
    PhoneNumber VARCHAR(15),
    Points INT DEFAULT 0,
```

```

        CreatedAt TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    );

-- Tabel Pesanan
CREATE TABLE Orders (
    OrderID INT AUTO_INCREMENT PRIMARY KEY,
    CustomerID INT,
    OrderStatus VARCHAR(50),
    TotalAmount DECIMAL(10, 2),
    OrderTime TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    PickupTime TIMESTAMP,
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);

-- Tabel Item Pesanan
CREATE TABLE OrderItems (
    OrderItemID INT AUTO_INCREMENT PRIMARY KEY,
    OrderID INT,
    ProductID INT,
    Quantity INT,
    ItemPrice DECIMAL(10, 2),
    FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
    FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
);

-- Tabel Produk
CREATE TABLE Products (
    ProductID INT AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(100),
    Description TEXT,
    Price DECIMAL(10, 2),
    ImageURL VARCHAR(255),
    CreatedAt TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Tabel Protokol Kesehatan
CREATE TABLE HealthProtocols (
    ProtocolID INT AUTO_INCREMENT PRIMARY KEY,
    EmployeeID INT,
    Temperature DECIMAL(4, 2),
    MaskUsed BOOLEAN,
    RecordTime TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID)
);

```

```
);  
  
-- Tabel Karyawan  
CREATE TABLE Employees (  
    EmployeeID INT AUTO_INCREMENT PRIMARY KEY,  
    Name VARCHAR(100),  
    Email VARCHAR(100),  
    PhoneNumber VARCHAR(15),  
    Position VARCHAR(50),  
    JoinedDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

Desain UI:

1. Splash Screen



2. Log In



STARBUCKS COFFEE

Welcome to Starbucks official application
Enter your information to get started

[Login](#) [Sign Up](#)

Enter email or username

Password 

[Forgot Password ?](#)

Login

3. Sign Up



STARBUCKS COFFEE

Welcome to Starbucks official application
Enter your information to get started

[Login](#) [Sign Up](#)

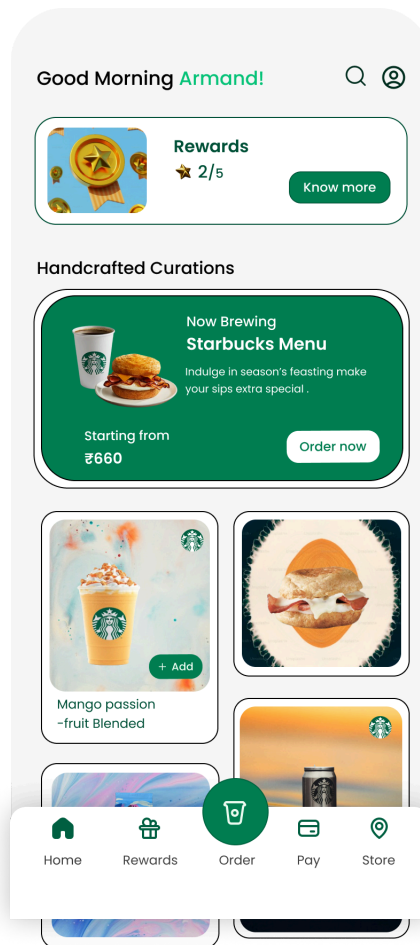
Enter email or username

Password 

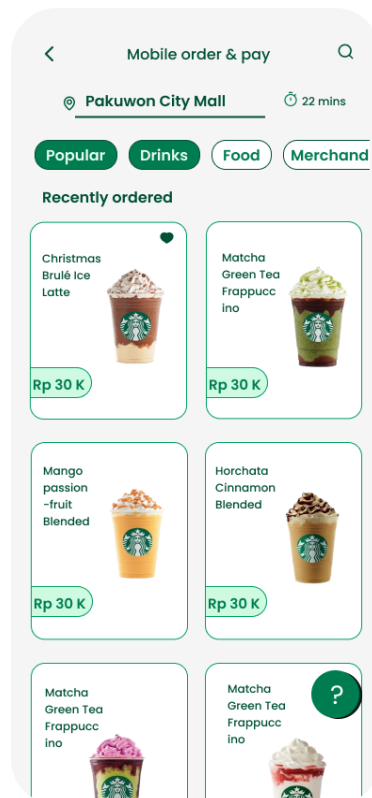
Confirm Password

Sign Up

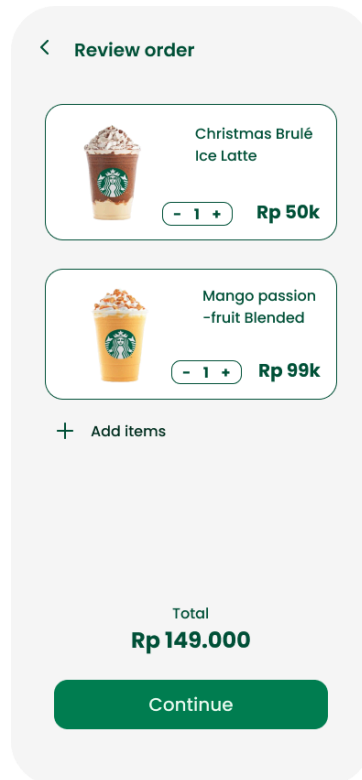
4. HomePage



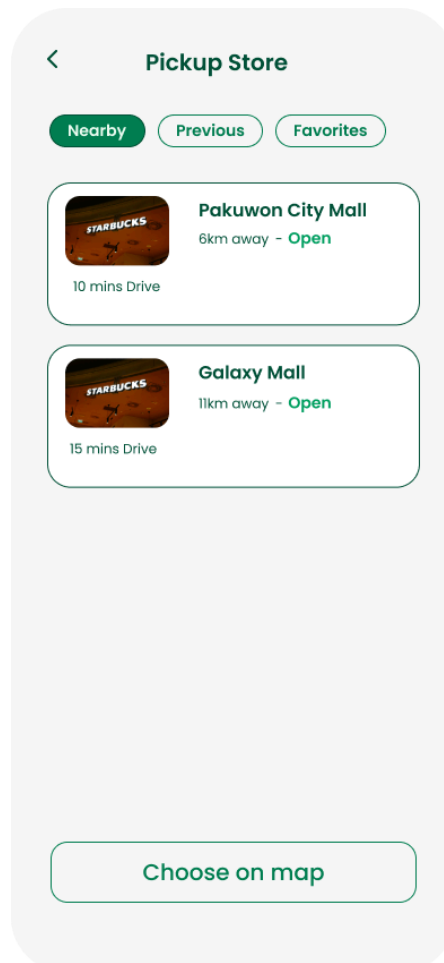
5. Menu Page



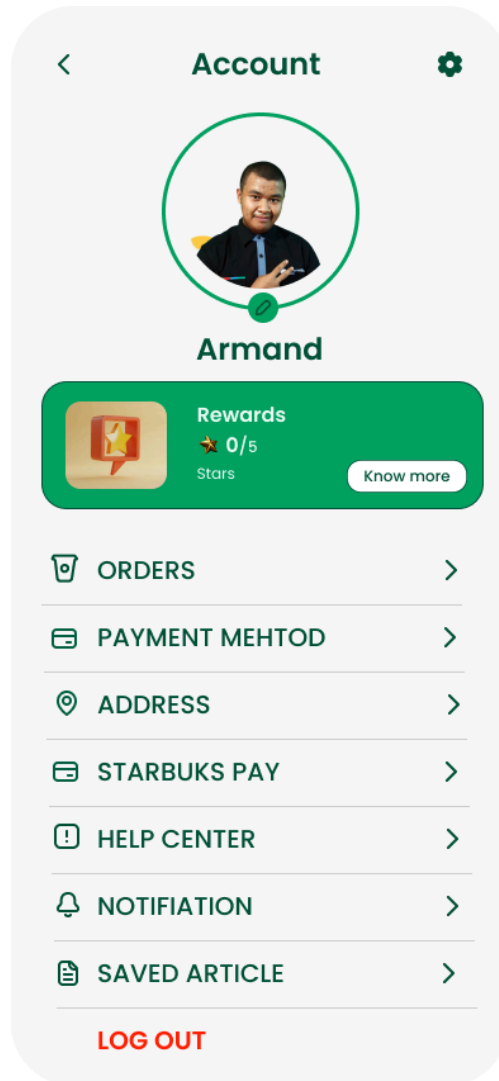
6. Order Page



7. Order Status



8. Profile



5. Buat Presentasi dan demo hasil rancangan Perangkat Lunak, kemudian upload di Youtube, dan isi lembar monitoring?

https://youtu.be/e_PDb7sL3Q0