

Kelompok : HIM

1. 13521118 / Ahmad Ghulam Ilham
2. 13521152 / Muhammad Naufal Nalendra
3. 13521158 / Muhammad Dhiwaul Akbar
4. 13521159 / Sulthan Dzaky Alfaro
5. 13521166 / Mohammad Rifqi Farhansyah
6. 13521169 / Muhammad Habibi Husni

Asisten Pembimbing : 13519007 / Muhammad Tito Prakasa

Link Github : https://github.com/habibibi/Tubes2_OOP.git

1. Deskripsi Umum Aplikasi

Sistem POS (Point of Sales) adalah sistem yang digunakan oleh bisnis untuk memproses transaksi penjualan secara efisien dan akurat. Sistem ini biasanya terdiri dari perangkat keras dan perangkat lunak yang bekerja bersama untuk mencatat dan melacak penjualan, mengelola stok barang, dan menghasilkan laporan penjualan. Perangkat keras POS umumnya mencakup komputer, monitor, printer struk, dan pemindai barcode, sedangkan perangkat lunak POS dirancang khusus untuk mengelola berbagai aspek operasional bisnis, seperti pencatatan penjualan, pembayaran, pengelolaan inventaris, dan pelaporan penjualan. Sistem POS memungkinkan bisnis untuk mengoptimalkan proses penjualan, mengurangi kesalahan manusia, dan meningkatkan efisiensi dalam mengelola operasi penjualan mereka.

Adapun fitur-fitur yang ada dalam aplikasi. Aplikasi kami terdiri dari yang pertama halaman utama. Halaman utama terdiri dari hari yang sesuai dengan hari pengguna saat digunakan dan juga jam sesuai dengan jam pengguna pada saat digunakan. Selanjutnya, fitur yang kedua adalah menu dan tab. Dalam setiap halaman yang tersedia, dimuat dalam sebuah menu. Dan untuk setiap halaman yang sudah dibuka akan dibuat menjadi tab yang bisa ditutup. Secara umum, bentuknya seperti browser web. Selanjutnya, ada fitur customer, member dan vip. Customer merupakan pelanggan biasa yang tidak memiliki keistimewaan, Member merupakan pelanggan yang sudah tersimpan di sistem dan memiliki keistimewaan yaitu setiap pesanan yang dibuat dapat menghasilkan poin, dan VIP merupakan pelanggan yang mirip dengan Member tetapi memiliki tambahan yaitu mendapatkan diskon sebesar 10%. Lalu ada Sistem Usaha Barang.

2. Kakas GUI: <Java FX>

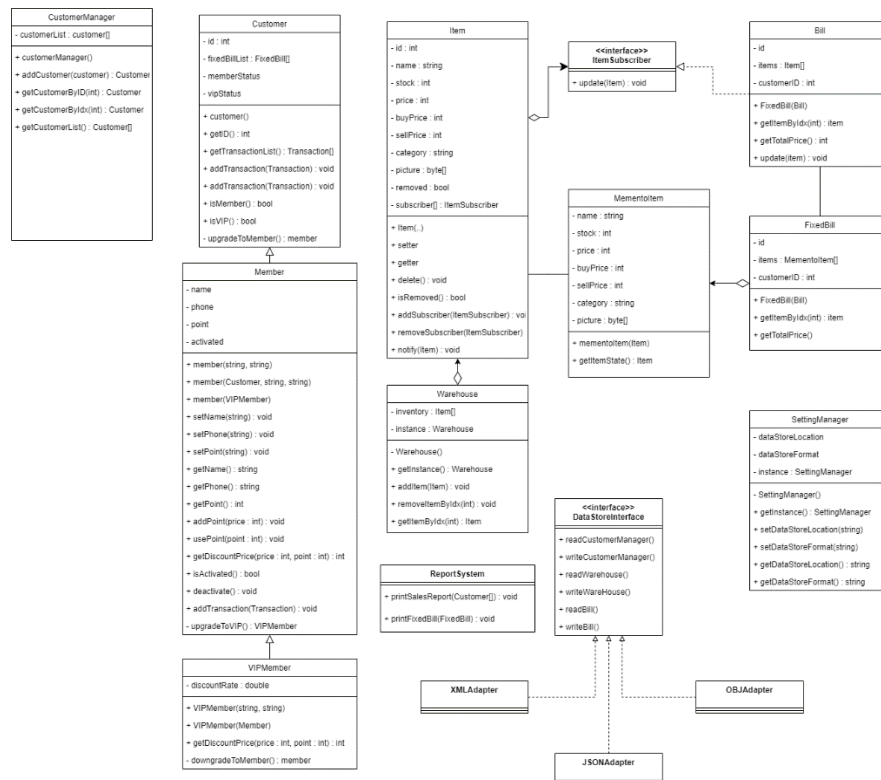
Kakas GUI yang kami pakai adalah Java FX. Java FX itu sendiri adalah platform perangkat lunak untuk membuat dan mengirimkan aplikasi desktop. JavaFX terdiri dari kumpulan package graphic dan media yang memungkinkan developer untuk membuat aplikasi desktop. GUI kami buat dalam package com.himehime.app serta untuk bagian backend kami meletakkannya pada package com.himehime.lib. Kami juga menggunakan beberapa eksternal resources pada folder resources. Lifecycle pada aplikasi kami akan dimulai dengan tampilan main menu yang tidak dapat ditutup. Pengguna dapat berpindah page dengan terlebih dahulu memilih menuBar dan dilanjutkan dengan memilih tabBar yang hendak dituju. Komponen page yang kami sediakan yaitu 9 page, mulai dari main menu, registration member, update membership, Riwayat, add inventory, update inventory, sales report, bill, serta settings. Registration member dan

update membership berkaitan dengan fungsionalitas yang terkait dengan customerManager. Sementara itu add inventory dan update inventory berkaitan dengan fungsionalitas yang terkait dengan warehouse atau inventoryManager.

3. Plugin & Class Loader

Class Plugin dan class Loader tidak diimplementasikan.

4. Class Diagram



Pertama, ada objek customer yang memiliki anak yaitu member dan member juga memiliki anak yaitu VIPMember. Beberapa pelanggan ini dapat disimpan dalam satu arraylist yaitu dalam customer manager. Customer manager ini dapat menyimpan customer, member, maupun VIP member. Selanjutnya ada Warehouse

dan Item. Pertama ada Item, yaitu objek benda pada aplikasi ini yang nanti disimpan pada inventory pada Warehouse. Ada juga class Bill untuk menyimpan item item yang sedang ditambahkan(dibeli) pada aplikasi. Data ini masih sementara dan dapat dihapus maupun ditambahkan. Dan ada juga class FixBill yang merupakan Bill yang sudah fix dan tidak dapat diubah lagi. Dan ada juga adaptor yang mana data data yang ingin disimpan dapat menggunakan adaptor ini ke dalam sebuah file. Adaptor ini merupakan interface dari DataStoreInterface.

5. Konsep OOP

5.1. Inheritance

- Kelas Member (Tubes2_OOP\src\main\java\com\himehime\lib\Member.java) merupakan anak dari kelas Customer (Tubes2_OOP\src\main\java\com\himehime\lib\Customer.java)
- Kelas VIP Member (Tubes2_OOP\src\main\java\com\himehime\lib\VIPMember.java) merupakan anak dari kelas Member (Tubes2_OOP\src\main\java\com\himehime\lib\Member.java)

5.2. Composition

- Kelas Customer (Tubes2_OOP\src\main\java\com\himehime\lib\Customer.java) memiliki banyak kelas FixedBill
- Kelas Bill (Tubes2_OOP\src\main\java\com\himehime\lib\Bill.java) memiliki banyak kelas Item
- Kelas FixedBill (Tubes2_OOP\src\main\java\com\himehime\lib\FixedBill.java) memiliki banyak kelas MementoItem
- Kelas Warehouse (Tubes2_OOP\src\main\java\com\himehime\lib\Warehouse.java) memiliki banyak kelas Item

5.3. Interface

- DataStoreInterface (Tubes2_OOP\src\main\java\com\himehime\lib\DataStoreInterface.java)
- ItemSubscriber (Tubes2_OOP\src\main\java\com\himehime\lib\ItemSubscriber.java)

5.4. Method Overriding dan Method Overloading

- Method addTransaction pada kelas Member (Tubes2_OOP\src\main\java\com\himehime\lib\Member.java)
- Method getDiscount pada kelas VIPMember (Tubes2_OOP\src\main\java\com\himehime\lib\VIPMember.java)

- Method update pada kelas Bill (Tubes2_OOP\src\main\java\com\himehime\lib\Bill.java)

5.5. Polymorphism

- Array list pada CustomerManager (Tubes2_OOP\src\main\java\com\himehime\lib\CustomerManager.java) dapat menyimpan Class Member, Customer dan VIPMember

5.6. Java API Collection

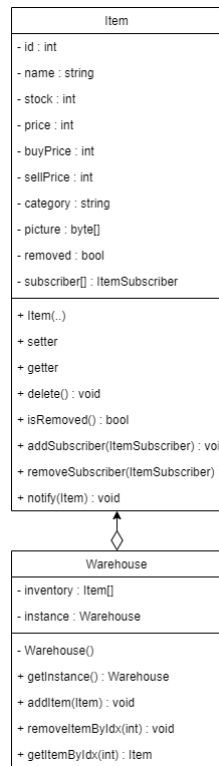
- Kelas Item (Tubes2_OOP\src\main\java\com\himehime\lib\Item.java) memiliki Array of Byte
- Kelas Customer (Tubes2_OOP\src\main\java\com\himehime\lib\Customer.java) memiliki ArrayList of FixedBill
- Kelas FixedBill (Tubes2_OOP\src\main\java\com\himehime\lib\FixedBill.java) memiliki ArrayList of MementoItem

5.7. SOLID

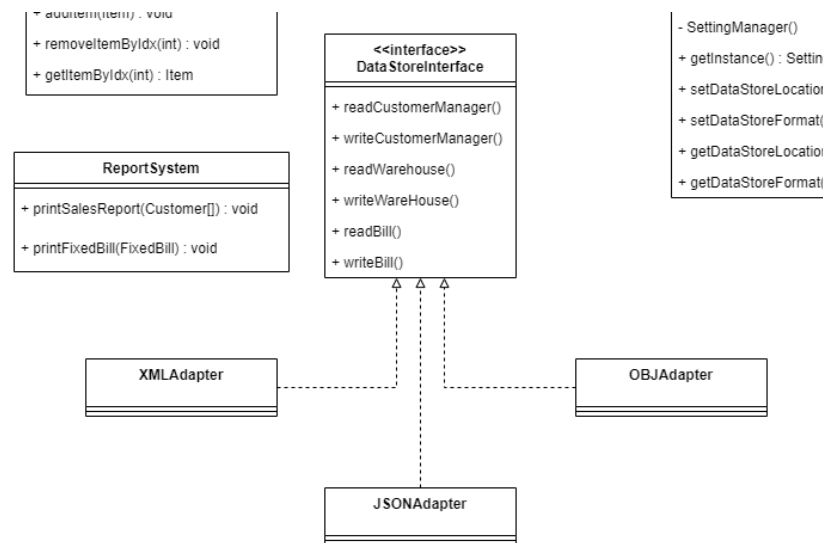
- S: Kelas Warehouse (Tubes2_OOP\src\main\java\com\himehime\lib\Warehouse.java) hanya bertugas untuk manage Inventory, penyimpanan Item, sedangkan Customer diatur oleh Kelas CustomerManager (Tubes2_OOP\src\main\java\com\himehime\lib\CustomerManager.java)
- O: Kelas Customer (Tubes2_OOP\src\main\java\com\himehime\lib\Customer.java) memiliki child yaitu Member (Tubes2_OOP\src\main\java\com\himehime\lib\Member.java) dan Member memiliki child berupa VIPMember (Tubes2_OOP\src\main\java\com\himehime\lib\Warehouse.java)
- L: Pada class Customer Member dan VIP Member, maka kelas Member atau VIP Member dapat di passing ke variable bertipe Customer tanpa mengubah behavior dari programnya.
- I: Pada kelas DataStoreInterface, dibagi menjadi 3 adapter yaitu XML, JSON, dan OBJ. Contohnya pada adapter JSON yang kami buat dapat bekerja untuk memenuhi interface yang tidak terlalu besar.
- D: Pada class Member bergantung pada adanya Customer dan VIP Member juga bergantung adanya class Member. Apabila tidak ada Customer, maka tidak dapat membangun

5.8. Design Pattern

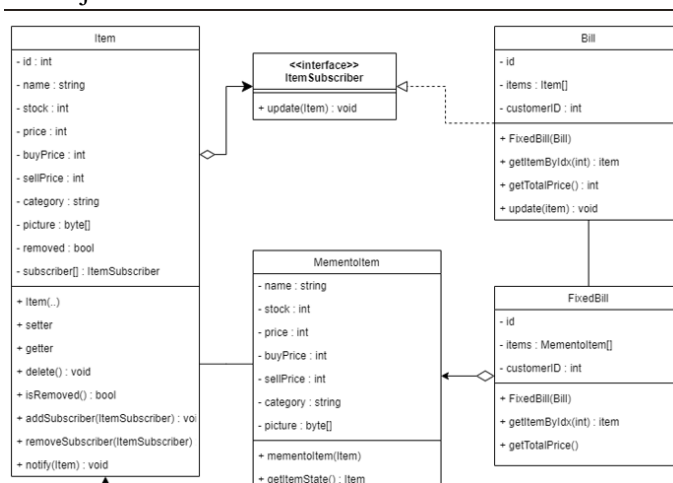
- Singleton: Warehouse (Tubes2_OOP\src\main\java\com\himehime\lib\Warehouse.java) sebagai singleton untuk manage Inventory, penyimpanan Kelas Item (Tubes2_OOP\src\main\java\com\himehime\lib\Item.java) tujuannya agar semua Item hanya perlu dimanage oleh satu objek saja, yaitu kelas Warehouse tersebut tanpa perlu membuat instansiasi baru setiap kali ingin mengubah Item.



- Adaptor: Adaptor yang kami buat yaitu JSONAdapter (Tubes2_OOP\src\main\java\com\himehime\lib\JSONAdapter.java). Adaptor ini merupakan implements dari interface DataStoreInterface yang terdiri dari method pengolahan data. Untuk pengolahan data, kami menggunakan ekstension file json.



- **Builder:** Builder disini digunakan untuk membangun sebuah objek dalam sebuah program secara bertahap. Dalam program kami, Builder diimplementasikan pada class Item. Item disini sangat penting dalam aplikasi karena aplikasi ini berkaitan dengan POS(Point of Sales) yang mana harus ada objek item.



5.9. Reflection

- Tidak diimplementasikan

5.10. Threading

- MainMenu (TubesOOPAsli\Tubes2_OOP\src\main\java\com\himehime\app\MainMenu.java) untuk membuat tampilan tanggal dan jam dibuat threading agar tidak mengganggu program utama dan bisa berjalan secara concurrent.

6. Pembagian Tugas

Pada bagian ini, tuliskan peran terbesar untuk masing-masing anggota kelompok. Tuliskan sesuai kenyataan. Misalnya:

- 13521118 / Ahmad Ghulam Ilham
 - ItemSubscriber
 - Bill
 - FixedBill
 - Setting Manager
 - GUI (Riwayat transaksi, Menu)
- 13521152 / Muhammad Naufal Nalendra
 - Customer
 - Member
 - VIP Member
 - Report System
 - GUI (Member Registration, Add Inventory, Update Inventory, Setting Menu, Sale Report)
- 13521159 / Sulthan Dzaky Alfaro
 - Item
 - Memento Item
 - Warehouse

- DataStoreInterface
- JSON Adapter
- Laporan
- 13521166 / Mohammad Rifqi Farhansyah
 - Update Membership
 - GUI (Styling semua page)
- 13521169 / Muhammad Habibi Husni
 - OBJ Adapter
 - GUI (Build page)
 - Desain Class
- 13521158 / Muhammad Dhiwaul Akbar
 - AFK

7. Foto Kelompok



8. Asistensi

1. Konten Diskusi

Link class diagram: [Diagram Kelas HIM \(Habibi IMba\)](#)

- Implementasi member dan VIP bagaimana?

Jawab: Di program atau database? Kalau di program bisa memanfaatkan ctor, objek customer dicopy ke objek member baru jadi dianggap sebagai objek baru yang berbeda.

- Untuk sistem item memakai design pattern singleton dan observer. Warehouse singleton supaya jadi source, tapi katanya sebaiknya singleton dihindari?

Jawab: Singleton sebaiknya dihindari karena best practice OOP garis keras tidak suka singleton, ibaratnya tidak memenuhi kaidah OOP yang baik dan benar. Untuk kasus ini singleton sudah benar, yang mengatur inventory cukup satu instance saja.

- Baru implement 4 design pattern (memento, observer, adapter, singleton) kira-kira satu lagi bisa pakai apa?

Jawab: Builder pattern. Misal ketika membuat satu objek kosong kemudian diset satu-satu atau constructor yang semua parameter terisi. Kalau pakai builder pattern menggunakan setter yang mengembalikan pointer this dan chaining operation.

- Konsep plugin bagaimana?

Jawab: Ide dari plugin sebetulnya main program dapat di-extend oleh external program dan external program disebut plugin. Main program menyediakan API (interface) yang bisa dipakai plugin. Misal, plugin ingin menggambarkan data penjualan dalam bentuk chart, bisa dibuat interface yang mendapatkan data transaksi atau data stok item yang nantinya digunakan oleh plugin. External program harus tahu method yang ada di main program. *Saran explore Maven child project, isinya interface yang bisa dipakai external program yang akan import child project.*

- Kalau tabel bentuk ISA bagaimana implementasi penyimpanan database?

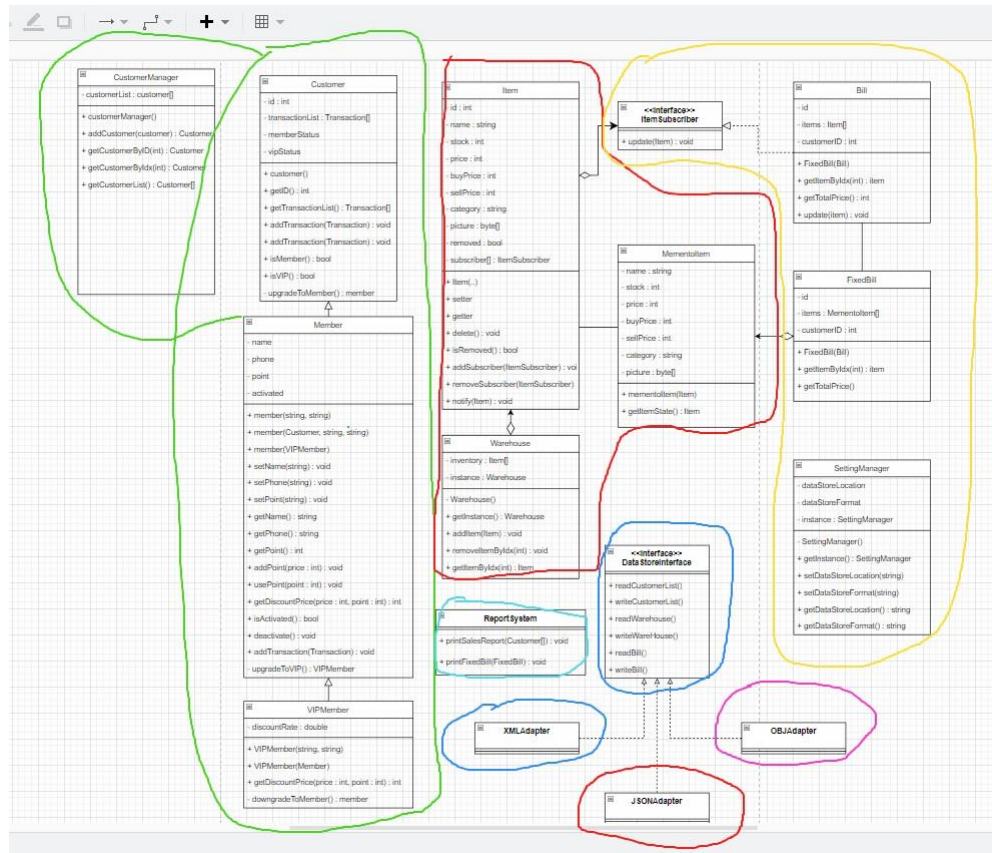
Jawab: Sebetulnya penggunaan ISA lebih ke menghilangkan field boolean member status dan VIP status, jadi data dilihat dari instance objek apa. ***Pesan untuk menyimpan data pakai relational database pakai one-to-one saja karena member mewakili satu customer, begitu juga VIP.***

- Bagaimana instansiasi objek Adapter?

Jawab: Sebaiknya datastore customer, warehouse, bill dipisah (terutama untuk poin bonus). Ketika mengerjakan bonus, ibaratnya table-table customer, warehouse, dan bill berbeda sehingga objek memiliki banyak tanggung jawab (menyalahi SOLID). Komunikasi DB akan terus hidup selama program berjalan sehingga method perlu ditambah data spesifik (e.g. get customer by id, get item by id), menambah data baru. Setiap update data, database juga di-update (untuk poin bonus).

2. Tindak Lanjut

- Pembagian tugas : Diagram ada di link diatas



Muhammad Naufal Nalendra : hijau

Ahmad Ghulam Ilham: kuning

Sulthan Dzaky Alfaro : merah

Muhammad Habibi Husni : pink

Mohammad Rifqi Farhansyah : biru

Muhammad Dhiwaul Akbar : biru muda (Tapi ngga kerjain)

3. Foto Dokumentasi

