

PROGRESS TUGAS BESAR : SISTEM KASIR RESTAURANT

Diajukan Untuk Memenuhi Tugas Mata Kuliah Pemrograman Berorientasi Objek

Dosen Pengampu : Agus Hartoyo, Ph.D



Disusun Oleh :

Afrizal Syahruluddin Yusuf	(1301194288)
Ni Made Dwipadini Puspitarini	(1301194141)
Fadhlurrahman Akbar Nasution	(1301194258)

INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY

2021/2022

Progress Tugas Besar

Dalam rangka memenuhi penilaian UTS pada mata kuliah Pemrograman Berorientasi Objek, berikut akan kami jabarkan progress tugas besar yang telah kelompok kami lakukan :

1. Topik Desain Sistem

Kami membangun desain sistem kasir di sebuah restaurant (*cashier management system*). Sistem ini nantinya akan membantu kegiatan transaksi berupa makanan dan minuman di suatu restaurant yang memiliki toko fisik atau *offline store*. Perlu menjadi catatan bahwa, desain sistem yang kami bangun tidak sama dengan pemesanan makanan secara online. Sistem kasir ini hanya digunakan oleh kasir saja dan pelanggan tidak terlibat secara langsung di dalamnya. Berikut merupakan spesifikasi yang kelompok kami tetapkan untuk desain sistem ini :

- a. Kasir dapat mengelola item yang akan dimasukkan ke sistem, baik itu menambah item, mengedit item, menghapus item, atau menampilkan item.
- b. Kasir dapat mengatur orderan yang dipesan secara langsung oleh pelanggan, baik itu membuat orderan, menampilkan orderan, hingga menghitung total harga yang harus dibayar oleh pelanggan.
- c. Item terdiri dari makanan (*food*) dan minuman (*beverage*).
- d. Pilihan metode pembayaran berupa kartu debit (bank) atau cash.
- e. Kasir mengatur metode pembayaran yang telah dipilih oleh pelanggan.
- f. Sistem dapat menghitung total harga belanja yang harus dibayar pelanggan dan ditambah dengan pajaknya.

2. Class Diagram

Diagram Kelas atau class diagram merupakan diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Class diagram disebut jenis diagram struktur karena menggambarkan apa yang harus ada dalam sistem yang dimodelkan dengan berbagai komponen. Berbagai komponen tersebut dapat mewakili class yang akan di program, objek utama, atau interaksi antara class dan objek.

Class sendiri merupakan istilah yang mendeskripsikan sekelompok objek yang semuanya memiliki peran serupa dalam sistem. Sekelompok objek ini terdiri atas fitur

struktural yang mendefinisikan apa yang diketahui class dan fitur operasional yang mendefinisikan apa yang bisa dilakukan oleh class. Fungsi dan Keunggulan Class Diagram :

1. Fungsi

Fungsi utama dari class diagram adalah menggambarkan struktur sebuah sistem pemrograman. Meski demikian, terdapat beberapa fungsi lainnya dari class diagram.

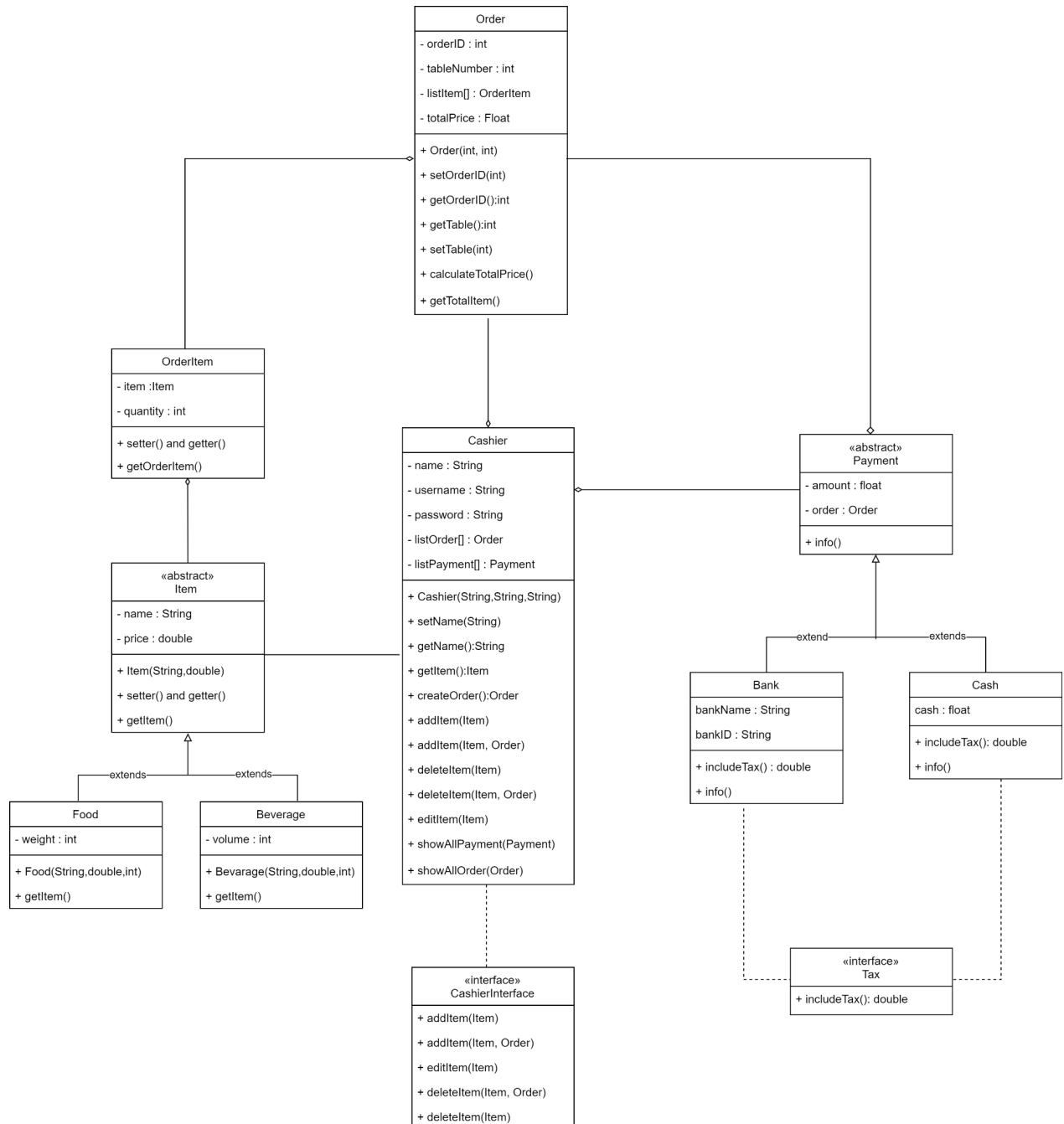
- Menunjukkan struktur statis pengklasifikasi dalam suatu sistem.
- Memberikan notasi dasar untuk diagram struktur lain yang ditentukan oleh UML.
- Dapat digunakan business analyst untuk membuat model sistem dari perspektif bisnis.

2. Keunggulan Class Diagram

Menurut LucidChart, class diagram memberikan beberapa keunggulan bagi pemrograman, terutama dalam bisnis. Berikut adalah beberapa keunggulan class diagram.

- Mampu mengilustrasikan model data untuk sistem informasi, terlepas dari apakah model data tersebut rumit atau sederhana.
- Memberikan gambaran umum mengenai skema aplikasi dengan lebih baik.
- Membantumu menyampaikan secara visual kebutuhan spesifik apa pun dari suatu sistem dan menyebarkan informasi tersebut ke bisnis.
- Terdapat bagan terperinci yang menyoroti kode spesifik yang perlu diprogram dan diterapkan ke struktur yang sesuai.
- Menyediakan deskripsi implementasi independen dari tipe yang digunakan dalam sistem untuk kemudian diteruskan di antara komponen-komponennya.

Dibawah ini merupakan class diagram dari Sistem Kasir Restaurant yang telah kami buat.



Deskripsi Class Diagram

a. Class Cashier

Class Cashier merupakan kelas untuk kasir dengan atribut yang dimiliki adalah name, username, password, array of listOrder, dan array of listPayment. Semua atribut tersebut memiliki hak akses private. Sedangkan, method yang dimiliki pada class

Cashier semuanya memiliki hak akses public, diantaranya adalah setter, getter, createOrder() : Order, showAllPayment(Payment) dan showAllOrder(Order).

b. Abstract Class Item

Class Item merupakan kelas yang akan menyimpan data-data mengenai item. Dimana item disini memiliki turunan yaitu Food atau makanan dan Beverage atau minuman. Atribut pada class ini memiliki hak akses private yang terdiri dari name dan price. Sedangkan untuk methodnya semua berakses public yang terdiri dari setter, getter, dan getItem().

c. Class OrderItem

Class OrderItem merupakan kelas yang akan menyimpan data-data mengenai item yang diorder oleh pelanggan.. Atribut pada class ini memiliki hak akses private yang terdiri dari item bertipe data Item dan quantity yang bertipe data integer. Sedangkan untuk methodnya semua berakses public yang terdiri dari setter, getter, dan getOrderItem().

d. Class Order

Class Order merupakan kelas yang akan menyimpan data-data mengenai orderan pelanggan secara lanjutan. Atribut pada class ini memiliki hak akses private yang terdiri dari orderID, tableNumber, totalPrice, dan array of listItem yang bertipe OrderItem. Sedangkan untuk methodnya semua berakses public yang terdiri dari setter, getter, calculateTotalPrice(), dan getTotalItem().

e. Abstract Class Payment

Class Payment merupakan kelas yang akan mengatur metode pembayaran yang dipilih oleh pelanggan. Dimana payment disini memiliki turunan yaitu Bank dan Cash. Atribut pada class ini memiliki hak akses private yang terdiri dari amount dan order yang bertipe Order. Sedangkan untuk methodnya adalah info() yang berakses public.

f. Class Interface CashierInterface

Class Interface CashierInterface merupakan interface yang dimiliki oleh class Cashier yang berisi method addItem(Item), addItem(Item, Order), deleteItem(Item), deleteItem(Item, Order), editItem(Item).

g. Class Interface Tax

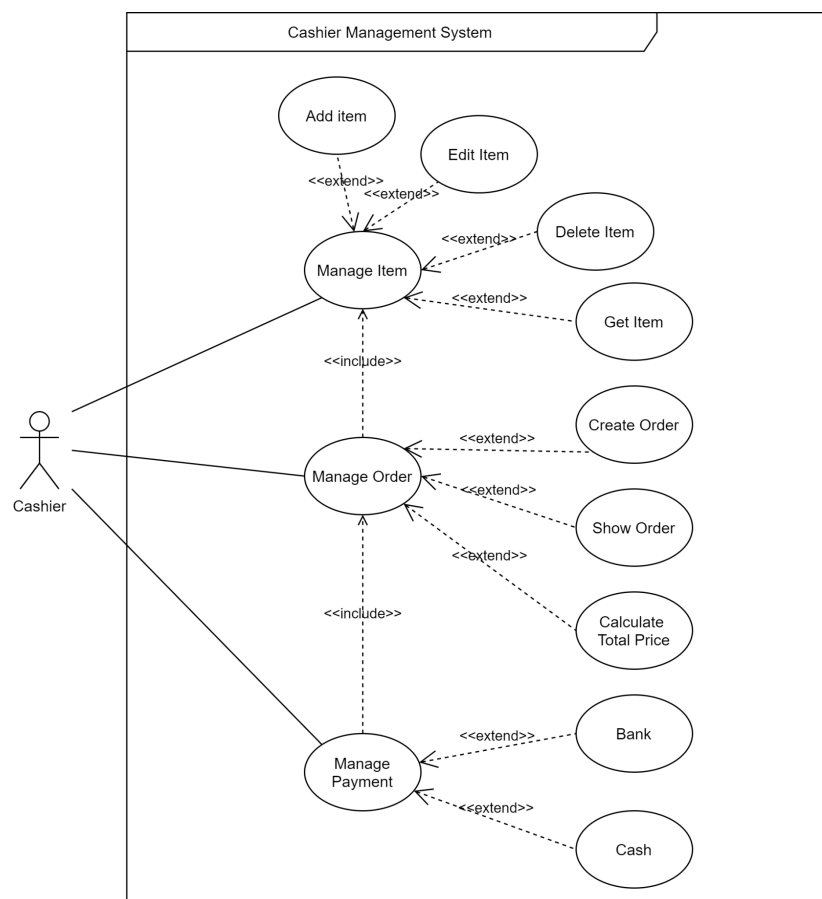
Class Interface Tax merupakan interface yang dimiliki oleh class Payment yang berisi method includeTax() : double untuk mengkalkulasikan total harga belanja yang sudah ditambah dengan pajak.

Relasi Antar Kelas

- a. Asosiasi
 - Class Cashier dengan Abstract Class Item
- b. Agregasi
 - Abstract Class Item dengan Class OrderItem
 - Class OrderItem dengan Class Order
 - Class Order dengan Class Cashier
 - Class Order dengan Abstract Class Payment
 - Abstract Class Payment dengan Class Cashier

3. Use Case Diagram

Berikut merupakan use case yang kami rancang sesuai dengan spesifikasi desain sistem yang telah kami jabarkan sebelumnya.



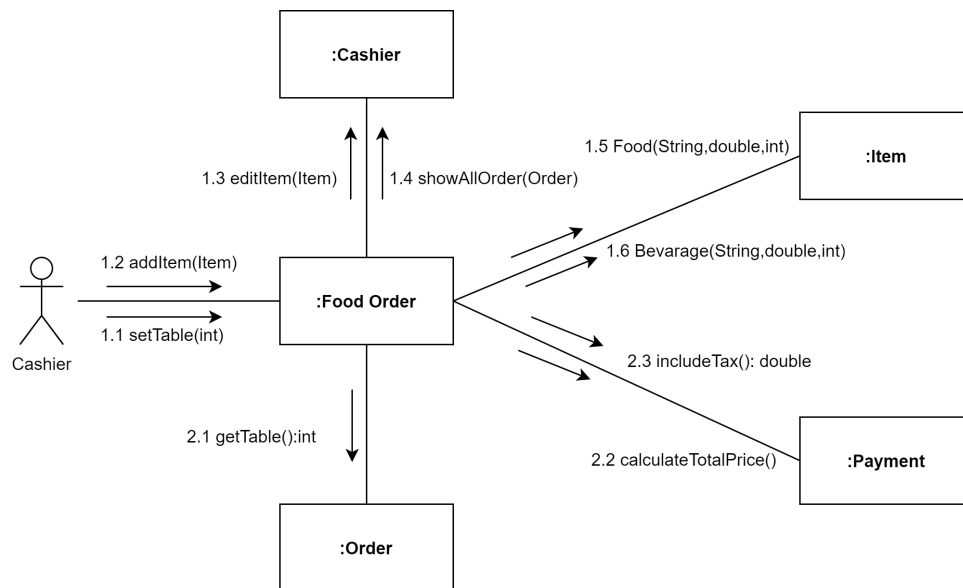
- a. Aktor, merupakan orang yang menggunakan sistem. Karena, sistem yang akan kami bangun diterapkan pada restaurant atau rumah makan yang memiliki toko fisik

(*offline store*) dan pelanggan tidak terlibat secara langsung di dalamnya. Maka, kasir akan menjadi satu-satunya aktor dalam sistem ini.

- b. Use case, merupakan interaksi yang terjadi antara sistem dengan aktor. Dalam sistem kasir ini, use case yang ada ialah :
- Manage Item : merupakan proses untuk mengelola item atau barang.
 - Manage Order : merupakan proses untuk mengatur orderan.
 - Manage Payment : merupakan proses untuk mengatur pembayaran yang dipilih.
- c. Include, menunjukkan bahwa suatu use case merupakan syarat yang harus dilakukan terlebih dahulu sebelum use case lainnya dilakukan. Misalnya pada sistem kasir ini :
- Sebelum kasir manage order, kasir harus melakukan manage item terlebih dahulu.
 - Sebelum kasir manage payment, kasir harus melakukan manage order terlebih dahulu.
- d. Extend, menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya. Misalnya pada sistem kasir ini :
- add item (menambah item), edit item (mengedit item), delete item (menghapus item), get item (menampilkan item) merupakan interaksi atau use case tambahan dari manage item.
 - create order (membuat pesanan), show order (menampilkan pesanan), calculate total price (menghitung total belanja yang harus dibayar pelanggan) merupakan interaksi atau use case tambahan dari manage order.
 - bank dan cash merupakan interaksi atau use case tambahan dari manage payment.

4. Collaboration Diagram

Berikut Collaboration Diagram dari Sistem Kasir Restaurant kelompok kami.



Collaboration Diagram pada umumnya juga dikenal sebagai Communication Diagram atau Interaction Diagram, merupakan ilustrasi dari relasi dan interaksi antara objek software pada Unified Modeling Language (UML). Collaboration Diagram mengelompokkan "message" pada kumpulan diagram sekuen menjadi sebuah diagram. Dalam diagram kolaborasi yang dituliskan adalah operasi atau metode yang dijalankan antara objek yang satu dengan objek yang lainnya secara keseluruhan. Oleh karena itu dapat diambil dari jalannya interaksi pada semua diagram sekuen. Untuk menggambarkan objek dari sebuah diagram kolaborasi, dapat menggunakan pilihan Object. Dan untuk menghubungkan antar objek yang satu dengan objek yang lain digunakan Link.

5. LAMPIRAN

Link Video Presentasi : <https://youtu.be/eQPF5DniWas>

Link Google Docs Laporan : [PBO](#)

Link Pembuatan Diagram :

https://drive.google.com/file/d/1JtQ_rVvbsn6XOrcrb6dL3Q2KmlDEuWO/view?usp=sharing