IF2010 Pemrograman Berorientasi Objek

TUGAS BESAR

GOSONG!

Kelompok A - K03

Leticia Aldina Trulykinanti 18223108 Laras Hati Mahendra 18223118 Firanti Naulia Fasya 18224119
Ibrahim Ferizarizqi Permana 18224120 Tyara Penelope Lumban Gaol 18224122

USER MANUAL

Getting Started

GETTiNG STARTED

1. CLONE REPOSiTORY DARi GiTHUB KE KOMPUTER KALiAN.
2. BUKA PROJECT Di VS CODE.
3. BUKA TERMiNAL Di VS CODE, LALU RUN
4. SETELAH iTU, PERMAiNAN AKAN SiAP DiJALANKAN.

OVERViEW PEMAiN & CHEF

DALAM PERMAiNAN iNi, PEMAiN MENGENDALiKAN DUA CHEF YANG BEKERJA SAMA Di DAPUR UNTUK MENYiAPKAN HiDANGAN.
SETiAP CHEF MEMiLiKi KEMAMPUAN DASAR YANG SAMA.

note: gunakan tab untuk switch chef

USER MANUAL

KONTROL GERAKAN

ATAS KiRi BAWAH KANAN

DUA TOMBOL UTAMA

TERDAPAT DUA JENiS TOMBOL AKSi:

V → INTERECT /AMBiL/GUNAKAN

> MENGAMBiL iNGREDiENT
> MENGAMBiL PLATE BERSiH,
> MENGAMBiL/MENGELUARKAN iTEM DARi OVEN,
> MEMULAi AKSi WASHiNG,
> MEMBUANG iTEM KE TRASH,
> MEMOTONG (CUTTiNG ACTiON),
> MENG-SERVE DiSH.

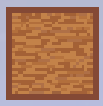
C → TARUH / PLACE / ASSEMBLY

> MENARUH iTEM,
> MENARUH BAHAN Di CUTTiNG,
> MENARUH BAHAN Di ASSEMBLY,
> MENARUH iTEM UNTUK DiMASAK,
> MENARUH PLATE / iTEM KE WASHiNG STATiON,
> MENGAMBiL PLATE DARi MEJA,
> ASSEMBLY DiSH.

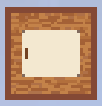

FINAL MAP
Active: Chef 1 (WASD)
TAB: Switch | C: Drop / Serve / Trash | V: Use Station
Score: 0
5

FINAL MAP

ASSET


Assembly Station


Cooking Station


Cutting Station


Washing Station


Trash Station

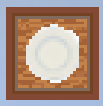
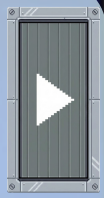


Plate Storage


Serving Counter


Ingredient Station - Cheese


Ingredient Station - Chicken


Ingredient Station - Sausage


Ingredient Station - Tomato


Ingredient Station - Dough

STRUKTUR FINAL

STRUKTUR FINAL

Game Summary
Untuk PASS

Stage Cleared

DESIGN PATTERN

```java
package controller;

import model.chef.Chef;
import model.station.Station;

public interface Action {
    boolean execute(Chef chef, Station station);
}
```

Strategy Pattern

→ Memisahkan algoritma interaksi dengan station yang berbeda-beda.

```java
public static Dish createPizzaDish(PizzaType type, List<Preparable> components) {
    if (type == null) {
        return null;
    }

    Dish dish;

    switch (type) {
        case MARGHERITA -> dish = new PizzaMargherita(ItemLocation.PLATING);
        case AYAM       -> dish = new PizzaAyam(ItemLocation.PLATING);
        case SOSIS      -> dish = new PizzaSosis(ItemLocation.PLATING);
        default -> {
            return null;
        }
    }
```

Factory Pattern

→ Membuat object tanpa menyebutkan class konkritnya secara eksplisit.

DESIGN PATTERN

```java
public static OrderManager getInstance() {
    if (instance == null) {
        instance = new OrderManager();
    }
    return instance;
}
```

Singleton Pattern

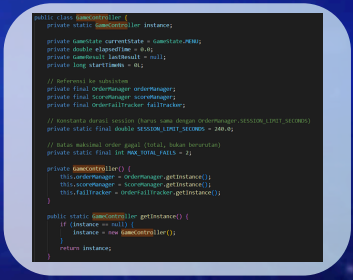
→ Memastikan hanya ada satu object dipakai disemua pihak.

```java
package model.enums;

public enum IngredientState {
    RAW,          // Mentah / belum diproses
    CHOPPED,      // Sudah dipotong
    COOKING,      // Sedang dimasak
    COOKED,       //Sudah matang
    BURNED        //hangus
}
```

State Pattern

→ Mengubah behavior object ketika internal state berubah.



Observer Pattern

→ "mengobservasi" events dari berbagai subsistem.

SOLID PRINCIPLES

```java
public class ScoreManager {

    private static ScoreManager instance;

    //Konstanta Skor
    public static final int POINTS_SUCCESS = 120; //kalo berhasil masukin ke serving
    public static final int PENALTY_FAIL = -50;  //kalo waktu order udah habis

    //State
    private int score = 0;
    private int successCount = 0;
    private int failCount = 0;

    private ScoreManager() {}

    public static ScoreManager getInstance() {
        if (instance == null) {
            instance = new ScoreManager();
        }
        return instance;
    }

    public static void resetInstance() {
        if (instance != null) {
            instance.score = 0;
            instance.successCount = 0;
            instance.failCount = 0;
        }
        instance = null;
    }
```

Single Responsibility Principle

→ Pada kode Class tersebut fokus pada satu tugas spesifik agar mudah di-maintain dan di-test

```java
public abstract boolean interact(Chef chef);
public abstract String getSymbol();
```

Open/Closed Principle (OCP)

→ memungkinkan penambahan jenis station baru melalui subclass tanpa perlu memodifikasi kelas Station.

```java
public abstract class Item { // abstract b
    private String name; //nama benda
    private ItemType type; //jenis: ingred
    private ItemLocation location; // loka
    private boolean isEdible;
    private boolean isClean;
```

Liskov Substitution Principle (LSP)

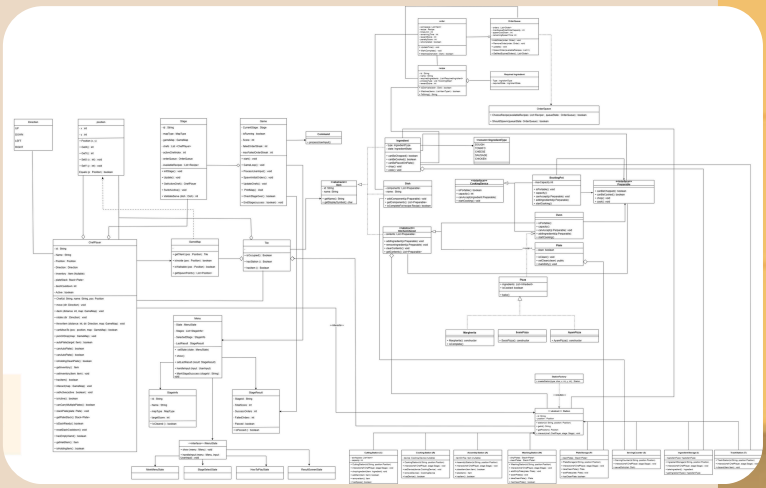
→ karena seluruh subclass memiliki struktur state yang sama dan dapat digunakan sebagai Item tanpa merusak asumsi tentang atribut dasar seperti nama, lokasi, dan status kebersihan.

Pembagian Tugas

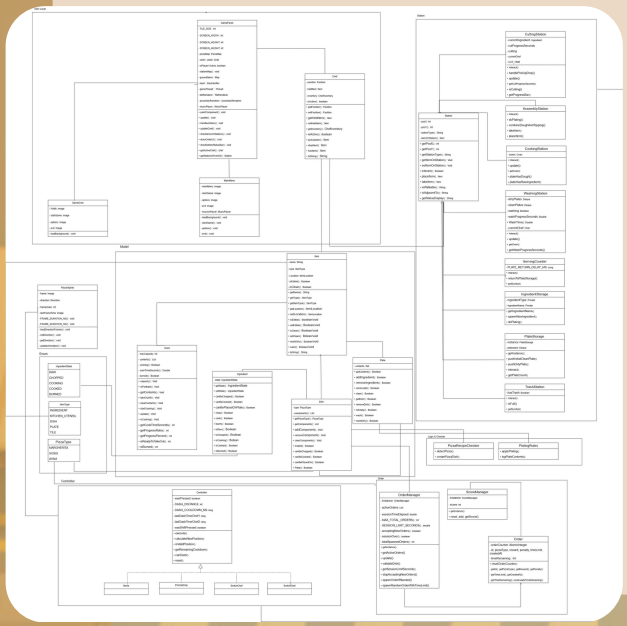
Nama	NIM	Tugas
Leticia Aldina Trulykinanti	18223108	• Membuat seluruh Stasiun, Game Over, Game Summary dan Stage Cleared dan implementasikan ke gamePanel • Mendesign aset Wall Tile, Floor Tile, Chef Merah & Biru, Ingredient Storage, Washing Station, Cooking Station, Serving Counter, dan, Trash Station • Mendesign Booklet
Laras Hati Mahendra	18223118	• Membuat seluruh Item, Dish, Preparable. • Mendesign asset Game, Order, seluruh aset item(ingredient, inventory, dish) • Mendesign Booklet dan menentukan design pattern dan SOLID • Membuat Class Diagram versi GUI
Firanti Naulia Fasya	18224119	• Mendesign aset Spawn,, Assembly Station, Cutting Station, PlateStorage • Membuat seluruh Actions, Controller, dan membuat model chef • Membuat dash dan throw (bonus) dan implementasi ke gamepanel
Ibrahim Ferizarizqi Permana	18224120	• Membuat Repository • Mengatur Timeline pengerjaan kelompok • Mengimplementasikan UI, Music, dan GUI untuk Game utama • Mengerjakan Order, OrderManager, ScoreManager.
Tyara Penelope Lumban Gaol	18224122	• Fixing pembuatan seluruh stasiun, item, actions, order, plating integrasikan semua ke dalam Game Panel • Mengerjakan PizzaMap dan nambahin sfx di setiap station • Integrasikan semua aset ke dalam gamePanel (GUI)

Class Diagram

Milestone I



Milestone 2

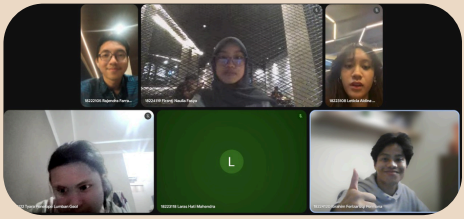


Pada class diagram kedua terlihat beberapa perubahan karena sudah disesuaikan dengan GUI. Karena pada diagram 1 hanya mencakup sesuai dengan spesifikasi.

full: https://bit.ly/3XKRTL5

Log Act

1. Asistensi 1 - 19 November 2025



Di asistensi pertama ini, kami sudah membuat layout utama class diagram dan diberikan beberapa *feedback* oleh asisten kami yaitu Kak Jendra. Beberapa poinnya adalah:

- Tidak perlu menggunakan frying pan
- Class Diagram bukanlah Diagram Relasional
- Tambahkan alasan kenapa harus ada method dan attribute tersebut
- GUI dapat diambil dari internet atau dibuat sendiri
- Di map pizza ada 3 plate. Beberapa piring kotor bisa langsung diangkut semua di plate storage. Tapi kalo piring bersih diambilnya satu per satu.
- Kalo kasusnya cuma 1 piring yg dipake masih ada 2 piring yg mengganggur di plate storage.

2. Pertemuan Offline - 25 November 2025

Merupakan pertemuan offline pertama yang *fullteam*. Disini kami membahas tentang pembagian tugas secara lebih detail dan mengerjakan *coding* tahap pertama.

3. Pertemuan Offline - 29 November 2025

Merupakan pertemuan offline kedua yang fullteam. Kami melanjutkan progress dari pertemuan sebelumnya termasuk menyelesaikan kode inti. Hasil dari pertemuan ini adalah layout final game di mana chef sudah bisa bergerak dan *switching*.

Log Act

4. Pertemuan Offline - 8 Desember 2025

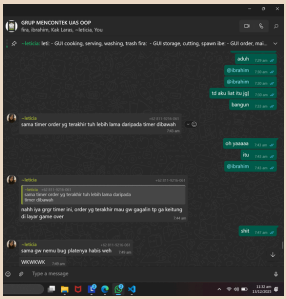
Kami lupa dokumentasi disini 😭 (sangking keosnya) namun disini kami berkumpul untuk kembali mengerjakan progress kami yang sudah sampai tahap implementasi GUI. Sejauh ini *floortile* dan *walltile* sudah terimplementasi.

5. Asistensi 2 - 9 Desember 2025

Asistensi terakhir dengan Kak Jendra. Disini kami konsultasi progress dan *address* beberapa hal yang sekiranya belum jelas di M2 ini.

6. Kerja Kelompok Last - 12 Desember 2025

Kerja kelompok terakhir sebelum pengumpulan (terus deadlinenya diundur) Harusnya ada Ibra juga cuma dia pulang duluan. SELEESAAAIIIIII. paling chaos



Masa masa panik karena diextend tapi masih ada yang error ahhah

GOSONG!

Ibrahim Ferizarizqi Permana
18224120

Firanti Naulia Fasya
18224119

Tyara Penelope Lumban Gaol
18224122

Leticia Aldina Trulykinanti
18223108

Laras Hati Mahendra
18223118

SELAMAT BERMAIN!