



**UNIVERSITAS TELKOM
FAKULTAS INFORMATIKA
PROGRAM STUDI REKAYASA PERANGKAT LUNAK**

TUGAS PROJECT MAHASISWA	
MATA KULIAH	Dasar Pemrograman Berorientasi Objek
DOSEN PENGAMPU	Gita Fadila Fitriana, S.Kom., M. Kom
BENTUK TUGAS	Final Project
TANGGAL PENGUMPULAN	
JUDUL TUGAS	
Program CRUD dalam Sistem Manajemen Perkantoran	
NAMA KELOMPOK	
ANGGOTA KELOMPOK	<ol style="list-style-type: none">1. Novita syahwa tri hapsari - 23111040072. Viona aziz syahputri - 23111040083. Mei sari mantiantini - 23111040124. Prajna paramitha w - 23111040165. Isabelle putri ardini - 2311104030
JOB DESK	
Novita syahwa tri hapsari	<ol style="list-style-type: none">1. Membuat class diagram beserta dengan source code untuk class karyawan, proyek, dan departemen induk.2. Membuat laporan progress report code program class diagram.3. Membuat laporan tugas besar.4. Membuat ppt untuk pengumpulan tugas besar.
viona aziz syahputri	<ol style="list-style-type: none">1. Membuat class diagram dan source code departemen content creator , departemen HRD, dan gaji content creator.2. Membuat laporan tugas besar.3. Membuat ppt untuk pengumpulan tugas besar.
mei sari mantiantini	<ol style="list-style-type: none">1. Membuat class diagram dan source code departemen Accounting, Gaji accounting, dan gaji HRD.2. Membuat laporan tugas besar3. Membuat ppt untuk pengumpulan tugas besar.
prajna paramitha wardhany	<ol style="list-style-type: none">1. Membuat laporan progress report class diagram.2. Membuat class diagram dan source code untuk class main, gaji induk, dan departemen marketing.3. Membuat laporan tugas besar.

	4. Membuat PPT untuk pengumpulan tugas besar.
isabelle putri ardini	1. Membuat class diagram dan source code departemen IT, Gaji IT, dan gaji Marketing. 2. Membuat laporan tugas besar. 3. Membuat ppt untuk pengumpulan tugas besar.
LINK GITHUB PROJECT	https://github.com/270505Nana/Sikumbanqantikwarnawarni_SE0701

PENDAHULUAN
<p>A. Latar belakang</p> <p>Pengelolaan data karyawan, proyek, dan gaji merupakan aspek penting dalam operasional perusahaan. Sistem manajemen karyawan yang efektif dapat membantu perusahaan dalam meningkatkan efisiensi, mengurangi kesalahan manual, dan mendukung pengambilan keputusan yang lebih baik. Sistem manajemen karyawan dirancang untuk menangani berbagai kebutuhan operasional perusahaan, mulai dari pencatatan informasi pribadi karyawan, penjadwalan proyek, hingga perhitungan gaji yang akurat sesuai kebijakan perusahaan. Dengan teknologi modern, sistem ini juga dapat meningkatkan transparansi, akurasi, dan kepuasan karyawan melalui akses yang lebih mudah dan cepat ke informasi penting.</p> <p>B. Tujuan</p> <p>Tujuan yang ingin dicapai melalui program yang dikembangkan, yaitu:</p> <ol style="list-style-type: none"> 1. Meningkatkan Efisiensi Operasional 2. Mengurangi Kesalahan 3. Memperbaiki Transparansi 4. Mengelola Data Karyawan Secara Terpusat 5. Mempermudah Pengelolaan Proyek 6. Meningkatkan Kepuasan Karyawan
DESKRIPSI APLIKASI
<p>Penjelasan meliputi poin-poin sebagai berikut !</p> <p>A. Penjelasan Program</p> <p>Program ini bertujuan untuk mempermudah pengelolaan sumber daya manusia di perusahaan. Fungsi utama dari program ini adalah mencatat dan menghapus data karyawan (nama, status, email, nomor telepon, jabatan, dan departemen) serta mengelompokkan berdasarkan departemen, dan mencatat dan menghapus data proyek</p>

(nama proyek, durasi, departemen, jumlah karyawan yang terlibat, beserta ID karyawan).

B. Fungsi utama dan kegunaan program.

Sistem manajemen karyawan bertujuan untuk membantu mengelola data karyawan, proyek, dan gaji dalam berbagai departemen secara efisien. Beberapa fungsi utama dari program:

- Pengelolaan Data Karyawan: Menambah, menampilkan, dan menghapus karyawan berdasarkan departemen.
- Pengelolaan Proyek: Menambah proyek baru, memperbarui durasi proyek, dan menampilkan daftar proyek sesuai departemen.
- Manajemen Gaji: Menghitung gaji berdasarkan kategori karyawan dan departemen, termasuk pengelompokan untuk karyawan tetap dan magang.

Kegunaan utama program ini untuk memberikan solusi untuk mengelola administrasi karyawan secara efisien, mengurangi kesalahan input yang dilakukan secara manual, dan memudahkan pendataan informasi terkait karyawan dan proyek.

C. Spesifikasi teknis program

Program ini dikembangkan menggunakan bahasa pemrograman java dengan JDK version 20. Pengembangan ini dilakukan menggunakan Integrated Development Environment (IDE) NetBeans versi 22. Untuk dokumentasi dan pengelolaan dependensi, program ini memanfaatkan Maven sebagai build automation tool. Dan untuk pengelolaan code pemrograman, sistem version control menggunakan Git yang terintegrasi dengan GitHub sebagai platform repository.

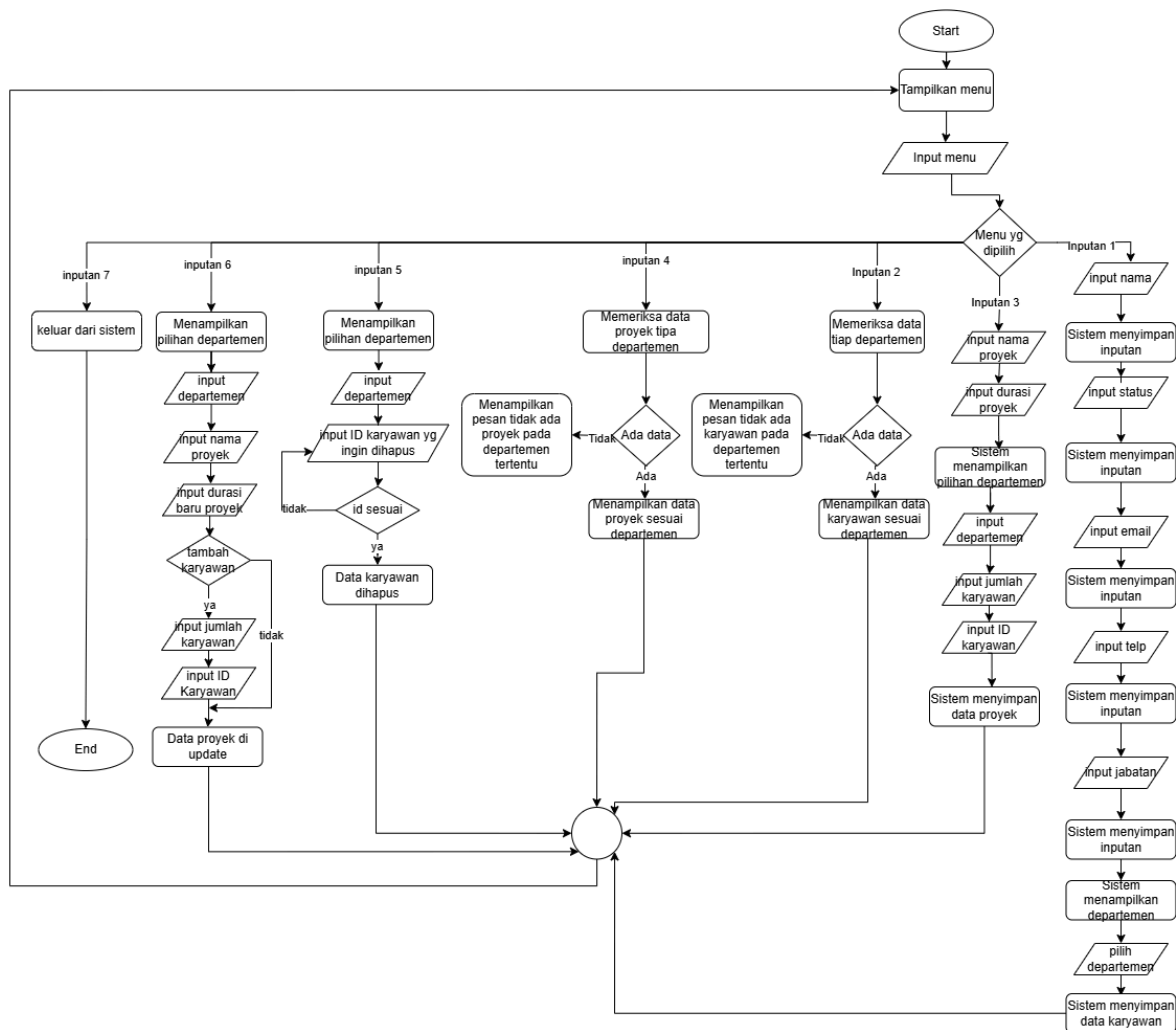
FITUR DAN KEGUNAAN

Daftar fitur beserta penjelasan dari setiap fitur - **nana**

1. Tambah Karyawan: Merupakan fitur untuk menambahkan data karyawan
2. Tampilkan Karyawan: Merupakan fitur untuk menampilkan seluruh data karyawan sesuai dengan departemennya
3. Tambah Proyek: Merupakan fitur untuk menambah proyek sesuai dengan departemennya, dan menambahkan durasi pengerjaan proyek
4. Tampilkan Proyek: Merupakan fitur untuk menampilkan keseluruhan data proyek sesuai dengan departemennya
5. Update proyek: Merupakan fitur untuk mengubah data durasi proyek yang sudah ada, jadi untuk durasi bisa diperpanjang ataupun diperpendek.
6. Hapus karyawan: Merupakan fitur untuk menghapus data karyawan berdasarkan ID Pegawai.

ANALISIS DAN DESAIN PROGRAM

A. Diagram alur (flowchart) dan penjelasan logika dan alur kerja program.



Penjelasan logika :

Program ini merupakan sistem informasi untuk pengelolaan data karyawan dan proyek departemen. Logika program dirancang dengan pendekatan berbasis menu, sehingga setiap fungsi program dapat diakses melalui input menu oleh pengguna.

1. Menu utama:

- Program dimulai dengan menampilkan menu utama, yang memberikan opsi untuk mengelola karyawan, proyek, atau keluar dari sistem.
- User memilih menu yang diinginkan.

2. Validasi input :

- Sistem memastikan input dari user valid.
- Jika tidak valid program akan memberikan pesan error.

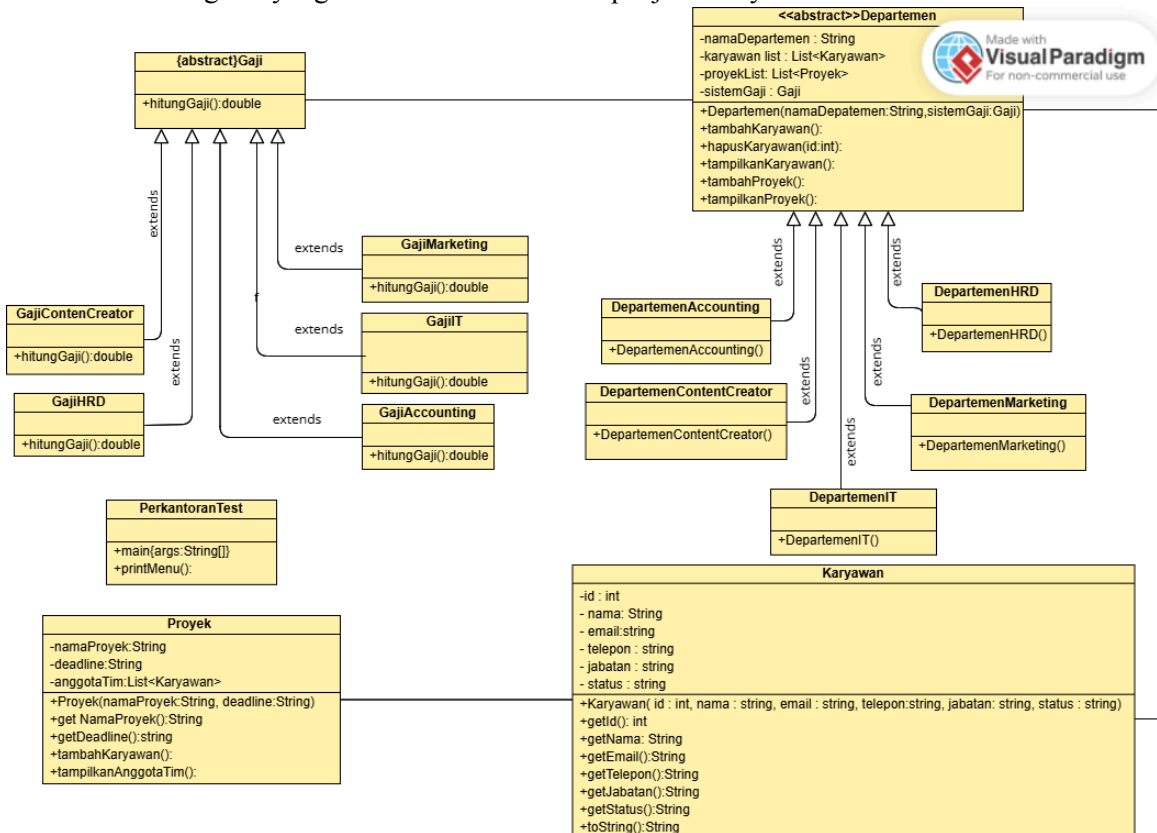
3. Manipulasi data :

- Data karyawan : input, pengeditan atau penghapusan data karyawan.
- Data proyek : input, pengeditan atau penghapusan data proyek.
- 4. Looping :
 - Setelah setiap tindakan selesai, program akan kembali ke menu utama untuk memilih opsi berikutnya hingga user memilih opsi keluar dari sistem.
- 5. Sistem keluar
 - program hanya akan berhenti apabila user memilih input 7.

Alur kerja program :

1. Start dan Tampilan Menu
 - Program dimulai dengan tampilan menu utama.
 - User diberikan pilihan untuk memilih input (1–7).
2. Input Data Karyawan (Input 1)
 - a. User diminta memasukkan data karyawan:
 - Nama, email, telepon, dan jabatan.
 - b. Data tersebut disimpan oleh sistem ke data karyawan.
 - c. Sistem memberikan konfirmasi bahwa data telah berhasil disimpan.
3. Pemeriksaan Data Karyawan (Input 2)
 - a. User memilih departemen tertentu.
 - b. Sistem memeriksa apakah terdapat karyawan di departemen tersebut:
 - Jika ada: Menampilkan daftar data karyawan.
 - Jika tidak ada: Menampilkan pesan bahwa data tidak ditemukan.
4. Pemeriksaan Data Proyek (Input 3)
 - a. Sistem memeriksa data proyek pada departemen tertentu.
 - Jika proyek ada: Menampilkan data proyek yang sesuai.
 - Jika tidak ada: Memberikan pesan bahwa proyek tidak ditemukan.
5. Memanipulasi Data Proyek (Input 4)
 - a. User menginput nama proyek dan memilih untuk memperbarui data.
 - b. Sistem memvalidasi apakah proyek tersebut ada dalam database:
 - Jika valid: Data proyek diperbarui.
 - Jika tidak valid: Pesan error ditampilkan.
6. Menghapus Data Karyawan (Input 5)
 - a. Pengguna memasukkan departemen dan ID karyawan untuk dihapus.
 - b. Sistem memvalidasi ID tersebut:
 - Jika valid: Data karyawan akan dihapus dari database.
 - Jika tidak valid: Pesan error ditampilkan.
7. Manipulasi Departemen (Input 6)
 - a. User dapat menambah atau menghapus departemen dari database.
 - b. Setelah operasi selesai, sistem memberikan konfirmasi perubahan.
8. Keluar dari Sistem (Input 7)
 - a. Sistem akan berhenti menjalankan program dan menampilkan pesan akhir.

B. Class diagram yang telah anda buat beserta penjelasannya



Class Gaji : Merupakan class yang berfungsi untuk mengelola gaji karyawan, yang kemudian dipisahkan berdasarkan 5 departemen yang ada, yaitu : Content creator, marketing, it, hrd dan accounting. Dimana memiliki 1 method yaitu `hitungGaji` dengan tipe data double.

Class Content Creator : Merupakan sub class dari gaji dimana didalamnya akan menggunakan method yang sama dengan paretnnya yaitu Gaji dan melakukan set gaji untuk karyawan departemen content creator baik karyawan tetap dan intern.

Class marketing : Merupakan sub class dari gaji dimana didalamnya akan menggunakan method yang sama dengan paretnnya yaitu Gaji dan melakukan set gaji untuk karyawan departemen marketing baik karyawan tetap dan intern.

Class HRD: Merupakan sub class dari gaji dimana didalamnya akan menggunakan method yang sama dengan paretnnya yaitu Gaji dan melakukan set gaji untuk karyawan departemen HRD baik karyawan tetap dan intern.

Class IT: Merupakan sub class dari gaji dimana didalamnya akan menggunakan method yang sama dengan parentnya yaitu Gaji dan melakukan set gaji untuk karyawan departemen IT baik karyawan tetap dan intern.

Class Accounting: Merupakan sub class dari gaji dimana didalamnya akan menggunakan method yang sama dengan parentnya yaitu Gaji dan melakukan set gaji untuk karyawan departemen accounting baik karyawan tetap dan intern.

Class departemen : Merupakan parent class departemen dimana didalamnya memiliki 4 atribut yaitu : namaDepartemen, Array list karyawan list, Array list Proyek, sistemgaji gaji. Lalu memiliki beberapa method yaitu :

1. tambahKaryawan : untuk menambahkan karyawan kedalam array lalu menampilkan pesan bahwa karyawan berhasil di tambahkan
2. hapusKaryawan : untuk menghapus data karyawan berdasarkan dengan id karyawan dan juga menghapus datanya dari departemen tersebut.
3. TampilkanKaryawan : untuk menampilkan keseluruhan karyawan, dimana disini dia hanya memanggil arrayList karyawanList lalu melakukan looping untuk menampilkan seluruh data karyawan.
4. tambahProyek : untuk menambahkan proyek kedalam arrayList proyekList lalu menampilkan message bahwa proyek berhasil ditambahkan.
5. tampilkanProyek : untuk menampilkan keseluruhan proyek berdasarkan dengan departemennya. Memanggil arrayList proyekList.

Class Departemen Accounting : Merupakan sub class dari departemen dimana didalamnya hanya melakukan set nilai untuk nama departemennya dan gajinya.

Class Departemen HRD: Merupakan sub class dari departemen dimana didalamnya hanya melakukan set nilai untuk nama departemennya dan gajinya.

Class Departemen IT: Merupakan sub class dari departemen dimana didalamnya hanya melakukan set nilai untuk nama departemennya dan gajinya.

Class Departemen Marketing: Merupakan sub class dari departemen dimana didalamnya hanya melakukan set nilai untuk nama departemennya dan gajinya.

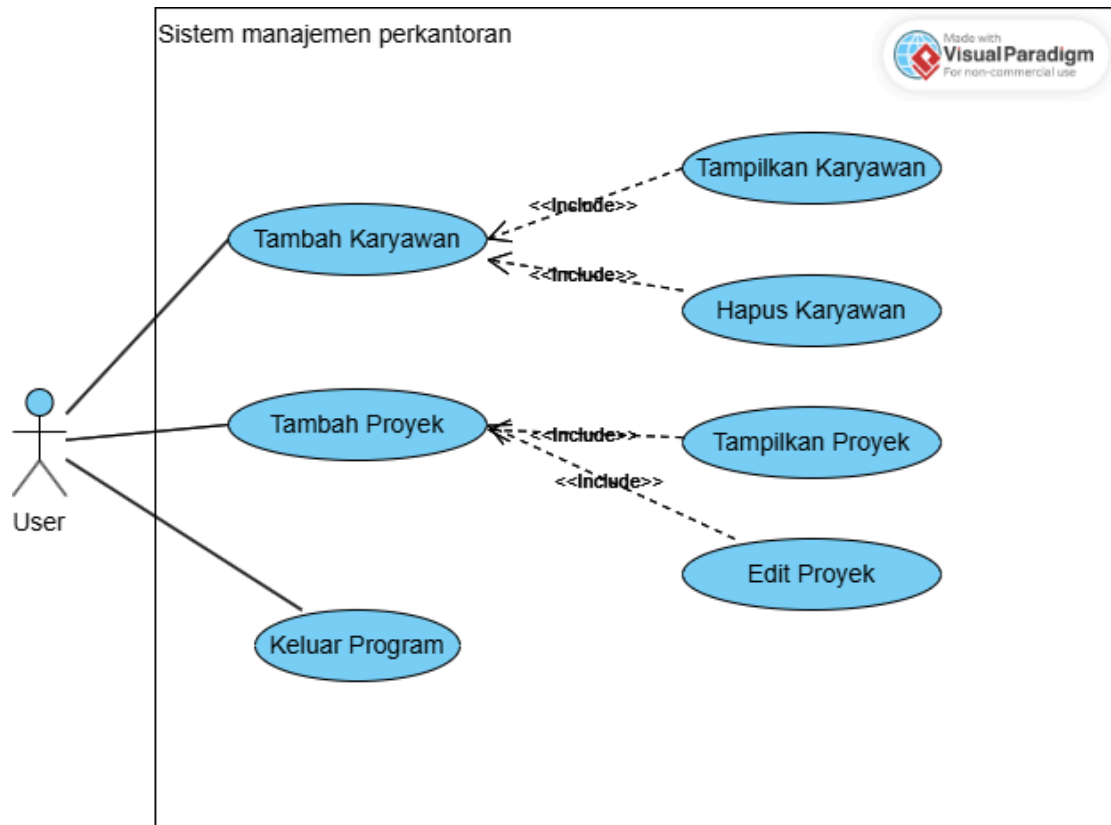
Class Departemen Content Creator: Merupakan sub class dari departemen dimana didalamnya hanya melakukan set nilai untuk nama departemennya dan gajinya

Class Proyek : Merupakan class proyek yang berfungsi untuk menampung data proyek yang ada dalam masing” departemen. Dimana memiliki atribut namaProyek, durasi dan memiliki List karyawanList yang nantinya digunakan untuk list karyawan di masing-masing proyek. Memiliki method getter untuk meng get nilai-nilai yang diinputkan. Kemudian memiliki method addKaryawan untuk menambahkan karyawan kedalam karyawanList

Class Karyawan : Dimana memiliki atribut id, nama, email, telepon, jabatan dan status. Dan memiliki getter untuk masing-masing atribut untuk meng get value dari masing-masing atribut dari inputan user.

Class perkantoran test : Merupakan class utama dimana program dipanggil dan dijalankan, yang nantinya akan menampilkan menu untuk tambah karyawan, hapus karyawan, tambah proyek, update proyek, menampilkan data karyawan, dan menampilkan data proyek.

C. Use case



D. Referensi untuk program ini dapat mencakup:

- Sistem Informasi Manajemen yang fokus pada pengelolaan karyawan dan proyek.
- Sistem Pengelolaan Proyek: Aplikasi seperti Trello yang digunakan untuk mengelola tim proyek dan distribusi tugas.
- Pemrograman Berorientasi Objek: Program ini sering kali digunakan untuk mendemonstrasikan konsep inheritance, polymorphism, dan abstraction dalam pengelolaan sistem.
- **Desain berbasis class dan subclass:** Program ini memanfaatkan pewarisan (inheritance) antara class parent (Gaji, Departemen) dan subclass.
- **Pemanfaatan polimorfisme:** Metode seperti hitungGaji menunjukkan penggunaan konsep overriding pada subclass.
- **Analisis dan desain menggunakan UML:** Diagram class dan flowchart mendukung pengembangan sistem dengan pendekatan terstruktur.

Referensi dari Open Source Projects:

- Contoh repositori GitHub yang relevan:
 1. Sistem informasi manajemen berbasis Java.
 2. Template untuk pengelolaan data karyawan dengan OOP.

IMPLEMENTASI PROGRAM

A. Penjelasan anda mengimplementasikan program secara ringkas

Struktur Data:

Karyawan disimpan dalam daftar (ArrayList) pada masing-masing departemen.

Proyek juga dikelola sebagai daftar yang berisi nama proyek, durasi, dan daftar karyawan.

Proses Utama:

1. Manajemen Karyawan:

- Pengguna dapat menambah karyawan dengan detail seperti nama, status, email, jabatan, dan departemen.
- Karyawan disimpan di dalam daftar (ArrayList) milik departemen.

2. Manajemen Proyek:

- Pengguna dapat membuat proyek baru dengan nama, durasi, dan menambahkan karyawan berdasarkan ID mereka.
- Proyek disimpan dalam daftar milik departemen.

3. Update Proyek:

- Proyek dapat diperbarui melalui fungsi untuk mengubah durasi dan menambah anggota baru.

4. Output Informasi:

- Menampilkan daftar karyawan dan proyek pada masing-masing departemen.

Antarmuka Pengguna:

Menggunakan menu berbasis teks untuk interaksi pengguna, dengan input numerik untuk navigasi dan manipulasi data.

B. Contoh potongan kode program yang menerapkan konsep OOP

1) Encapsulation

```
// Getters and Setters
public String getNamaDepartemen() {
    return namaDepartemen;
}

public void setNamaDepartemen(String namaDepartemen) {
    this.namaDepartemen = namaDepartemen;
}
```

Akses ke atribut kelas menggunakan getter dan setter untuk membatasi akses langsung. Variabel seperti karyawanList, namaDepartemen, dan sistemGaji hanya dapat diakses melalui metode yang disediakan.

2) Abstraction

```
abstract class Gaji {
    public abstract double hitungGaji(Karyawan karyawan);
}
```

Kelas abstract berfungsi sebagai template untuk kelas turunannya. Metode hitungGaji() di Gaji disediakan tanpa implementasi langsung, hanya diimplementasikan pada kelas turunannya.

3) Inheritance (pewarisan)

```
public class DepartemenAccounting extends Departemen {  
    public DepartemenAccounting(){  
        super("Accounting", new GajiAccounting());  
    }  
}
```

Pada codingan diatas menunjukkan bahwasannya class departemenAccounting merupakan class turunan dari Departemen.

4) Polymorphism

```
public double hitungGaji(Karyawan Karyawan){  
    if (Karyawan.getStatus().equalsIgnoreCase("Tetap")){  
        return 4_000_000;  
    }  
}
```

Kelas abstract gaji menggunakan metode hitungGaji() yg di override dalam kelas turunannya. Referensi ke kelas abstract digunakan untuk menyimpan objek kelas turunan.

C. Screenshot dari setiap fitur yang anda buat.

```
1. Tambah Karyawan  
2. Tampilkan Karyawan  
3. Tambah Proyek  
4. Tampilkan Proyek  
5. Hapus karyawan  
6. Keluar  
Pilih opsi: 1  
Nama Karyawan: Mei  
Status (Tetap/Intern): tetap  
Email: mei@gmail.com  
Telepon: 0822  
Jabatan: HRD  
1. Accounting  
2. HRD  
3. IT  
4. Marketing  
5. Content creator  
Pilih Departemen: 2  
Karyawan Mei berhasil ditambahkan.
```

```
Pilih opsi: 2  
Tidak ada karyawan.  
ID      Nama      Email      Telepon      Jabatan      Status  
Gaji  
=====
```

629	Mei	Mei	mei@gmail.com	0822	HRD	Rp0,00
-----	-----	-----	---------------	------	-----	--------

```
Tidak ada karyawan.  
Tidak ada karyawan.  
Tidak ada karyawan.
```

Pilih opsi: 3
Nama Proyek: SATELIT
Durasi Proyek: 2 bulan

Pilih Departemen untuk Tambahkan Proyek:
1. Accounting
2. HRD
3. IT
4. Marketing
5. Content Creator
Pilihan: 2
Masukkan jumlah karyawan yang terlibat: 2
Masukkan ID Karyawan yang terlibat: 01
Masukkan ID Karyawan yang terlibat: 02
Proyek SATELIT berhasil ditambahkan.

Pilih opsi: 4
Daftar Proyek di Departemen Accounting:
Tidak ada proyek yang terdaftar.
Daftar Proyek di Departemen HRD:
Proyek@70177ecd
Daftar Proyek di Departemen IT:
Tidak ada proyek yang terdaftar.
Daftar Proyek di Departemen Marketing:
Tidak ada proyek yang terdaftar.
Daftar Proyek di Departemen Content Creator:
Tidak ada proyek yang terdaftar.

Pilih opsi: 5
Pilih Departemen untuk Hapus Karyawan:
1. Accounting
2. HRD
3. IT
4. Marketing
5. Content Creator
Pilihan: 2
Masukkan ID Karyawan yang ingin dihapus: 01
Karyawan dengan ID 1 tidak ditemukan.

IMPROVEMENT PROGRAM SETELAH PRESENTASI TAHAP 1 (DENGAN ASDOS)

Jelaskan fitur-fitur/kode yang di improve sebagai perbaikan dari versi sebelumnya tampilkan. Jika memungkinkan anda dapat menampilkan before after ke dalam bentuk tabel.

Program Sebelumnya	Program setelah di Improve												
Program belum bisa menampilkan data karyawan sesuai dengan departemen	<p>Program telah bisa menampilkan data karyawan</p> <p>Departemen: Accounting Daftar Karyawan: Tidak ada karyawan.</p> <p>Departemen: HRD Daftar Karyawan: Tidak ada karyawan.</p> <p>Departemen: IT Daftar Karyawan:</p> <table><thead><tr><th>ID</th><th>Nama</th><th>Status</th><th>RpGaji</th></tr><tr><th colspan="4">=====</th></tr></thead><tbody><tr><td>479</td><td>Prajna paramitha</td><td>Intern</td><td>Rp6000000,00</td></tr></tbody></table> <p>Departemen: Marketing Daftar Karyawan: Tidak ada karyawan.</p> <p>Departemen: Content Creator Daftar Karyawan: Tidak ada karyawan.</p>	ID	Nama	Status	RpGaji	=====				479	Prajna paramitha	Intern	Rp6000000,00
ID	Nama	Status	RpGaji										
=====													
479	Prajna paramitha	Intern	Rp6000000,00										
Program belum bisa update data proyek sesuai departemen	<p>Program telah bisa melakukan update data proyek sesuai departemen</p> <p>Pilih opsi: 6 Pilih Departemen untuk Update Proyek: 1. Accounting 2. HRD 3. IT 4. Marketing 5. Content Creator Pilihan: 3 Masukkan Nama Proyek yang ingin diupdate: Web perkantoran Masukkan Durasi Baru Proyek: 4 Durasi proyek berhasil diperbarui. Masukkan jumlah karyawan yang ingin ditambahkan ke proyek: 0</p>												
Program belum bisa menggunakan try and catch untuk menangani inputan yang salah dari user	<p>Program sudah menggunakan try and catch untuk menangani kesalahan input dari user</p> <p>Masukkan jumlah karyawan yang terlibat: 1 Masukkan ID Karyawan yang terlibat: nana Inputan ID tidak valid. Harap masukkan angka.</p>												

PENGUJIAN PROGRAM

- A. Test case yang digunakan untuk menguji program (format tabel dapat digunakan).
- B. Hasil **REKAPITULASI** pengujian pada dokumen logbook testing pada lampiran.

Total Test Case	Pass	Fail	Persentase keberhasilan
9	7	2	80%

KESIMPULAN DAN SARAN

Sampaikan kesimpulan hasil akhir dari project apakah sangat baik, baik, cukup, kurang atau sangat kurang berdasarkan kriteria berikut

Kriteria	Persentase Test Case Berhasil	Deskripsi
Sangat Baik (4)	90% - 100%	Semua test case atau hampir semua test case berhasil dijalankan tanpa error.
Baik (3)	75% - 89%	Sebagian besar test case berhasil, ada sedikit error yang minor.
Cukup (2)	50% - 74%	Setengah atau lebih test case berhasil, tetapi banyak error signifikan.
Kurang (1)	25% - 49%	Hanya sebagian kecil test case berhasil, program banyak mengalami error.
Tidak Berhasil (0)	< 25%	Hampir semua test case gagal atau program tidak dapat dijalankan sama sekali.

Berikan hasil penjelasan alasan mendapat kriteria tersebut beserta saran ke depan.

Baik, dapat dilihat dari lampiran Logbook testing dibawah, dimana kami melakukan testing sebanyak 10 kali ditemukan bahwa selama melakukan testing tersebut terdapat 2 fail dan 8 pass. Dimana “*pass*” ini dapat diartikan hasil output aktual dengan hasil output yang diharapkan sesuai oleh karena itu dapat dikatakan “*pass*”, sedangkan untuk “*fail*” dimana ketika output aktual dengan output yang diharapkan tidak sesuai atau terjadi error sehingga dikategorikan “*fail*”. Saran kami untuk pengembangan aplikasi kedepannya adalah dibutuhkannya testing secara menyeluruh agar aplikasi dapat berjalan dengan baik tanpa terjadi error.

DAFTAR RUJUKAN

<https://www.dicoding.com/blog/contoh-use-case-diagram/>

<https://www.jagoanhosting.com/blog/oop-adalah/>

Putra, Fajar Pratama Purwanto (2022). Pengembangan Sistem Presensi untuk Work From Home (WFH) dan Work From Office (WFO) Selama Pandemi Covid-19. Jurnal Sains, Nalar, dan Aplikasi Teknologi Informasi Vol. 1 No. 2

LAMPIRAN 1
LOGBOOK TESTING

No	Test Case	Input	Output yang Diharapkan	Output Aktual	Status (Pass/Fail)
1	Input menu tidak valid (9)	9	Pilihan tidak valid! Silakan coba lagi.	Pilihan tidak valid! Silakan coba lagi.	Pass
2	Input departemen pada menu tambah karyawan tidak valid (7)	7	Departemen tidak tersedia, coba lagi	Error	Fail
3	Input Menu valid (1)	1	Meminta inputan user	Meminta inputan user	Pass
4	Inputan departemen proyek pada menu tambah proyek tidak valid	6	Pilihan departemen tidak valid!	Pilihan departemen tidak valid!	Pass
5	Inputan departemen untuk menu hapus karyawan tidak valid	6	Pilihan departemen tidak valid!	Pilihan departemen tidak valid!	Pass
6	Input ID karyawan pada menu hapus karyawan tidak valid	01	Karyawan dengan ID 1 tidak ditemukan.	Karyawan dengan ID 1 tidak ditemukan.	Pass
7	Input nama proyek pada menu update proyek tidak valid	“asad”	Proyek dengan nama asad tidak ditemukan di departemen Accounting	Proyek dengan nama asad tidak ditemukan di departemen Accounting	Pass
8	Inputan departemen proyek pada menu update proyek tidak valid	7	Pilihan departemen tidak valid!	Pilihan departemen tidak valid!	Pass
9	Input ID karyawan yang terlibat pada menu tambah proyek tidak valid	“baba”	Inputan ID tidak valid. Harap masukkan angka.	Error	Fail

Persentase keberhasilan :

Total Test Case	Pass	Fail	Persentase keberhasilan
9	7	2	80%