

**UJIAN TENGAH SEMESTER (UTS)  
GANJIL 2025/2026**

**Mata Kuliah:**  
**PEMROGRAMAN BERORIENTASI OBJEK**

Kelas D3TI2.A

**Studi Kasus:** Aplikasi Launcher Web dan Aplikasi Berbasis Console



Oleh:  
Akram nail dalimartha  
2403005

**D3 TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
POLITEKNIK NEGERI INDRAMAYU  
OKTOBER 2025**

## Daftar Isi

<b>Deskripsi Permasalahan .....</b>	<b>1</b>
<b>Analisis Kebutuhan.....</b>	<b>1</b>
<b>Analisis Fitur.....</b>	<b>1</b>
<b>Rancangan UML Use Case Diagram .....</b>	<b>3</b>
<b>Rancangan Tampilan Program Aplikasi Console .....</b>	<b>4</b>
<b>Analisis Class .....</b>	<b>5</b>
Identifikasi Class dan Attribute .....	5
Identifikasi Method .....	5
Jenis Relasi dan Alasan .....	5
Multiplicity.....	6
<b>Rancangan UML Class Diagram.....</b>	<b>7</b>
<b>Rancangan UML Sequence Diagram .....</b>	<b>8</b>
Sequence Diagram: main menu .....	8
Sequence Diagram: menjalankan group .....	9
Sequence Diagram: menjalankan aplikasi .....	10
Sequence Diagram: menambahkan group .....	11
Sequence Diagram: menambahkan aplikasi.....	12
Sequence Diagram: menghapus group .....	13
Sequence Diagram: menghapus aplikasi .....	14
<b>Pra-Kode Program .....</b>	<b>16</b>
<b>Kode Program .....</b>	<b>16</b>
Main.java .....	16
MainMenu.java .....	16
Aplikasi.java .....	20
Group.java.....	22
<b>Compile &amp; Run .....</b>	<b>25</b>
Compile .....	25
Run .....	25
<b>Testing.....</b>	<b>25</b>
Skenario menjalankan group (Gagal) .....	25
Skenario menjalankan group (Berhasil) .....	26
Skenario menjalankan group (Berhasil) .....	26
Skenario menjalankan aplikasi (Berhasil) .....	26
Skenario menjalankan aplikasi (Gagal) .....	27
Skenario menambah group (Gagal) .....	27
Skenario menambah group (Berhasil) .....	27
Skenario menghapus group (Gagal) .....	27
Skenario menghapus group (Berhasil) .....	28
Skenario menambah aplikasi (Berhasil) .....	28

Skenario menghapus aplikasi (Gagal) .....	28
Skenario menghapus aplikasi (Berhasil) .....	29
<b>Build (Deploy) .....</b>	<b>29</b>
<b>URL Repository .....</b>	<b>29</b>

## Deskripsi Permasalahan

Perusahaan game disukabumi memiliki masalah di pekerja . Ketika karyawan bekerja mereka tidak bekerja di satu aplikasi saja namun bekerja di banyak aplikasi dan website secara bersamaan. Sebagai contoh karyawan di divisi 3d model, karyawan tersebut membuat model di dalam aplikasi blender namun dia juga membutuhkan texture di web texturehaven.com dan beberapa website lain hal ini menyebabkan 2 hal, satu karyawan tersebut terkadang lupa aplikasi mana dan web mana yang karyawan tersebut butuhkan untuk bekerja, dua waktu kerja dipakai hanya untuk mencari website dan aplikasi yang dibutuhkan.

## Analisis Kebutuhan

Dibutuhkan suatu program aplikasi console yang dapat membuka website dan aplikasi yang dibutuhkan untuk setiap masing-masing karyawan.

Program dirancang untuk:

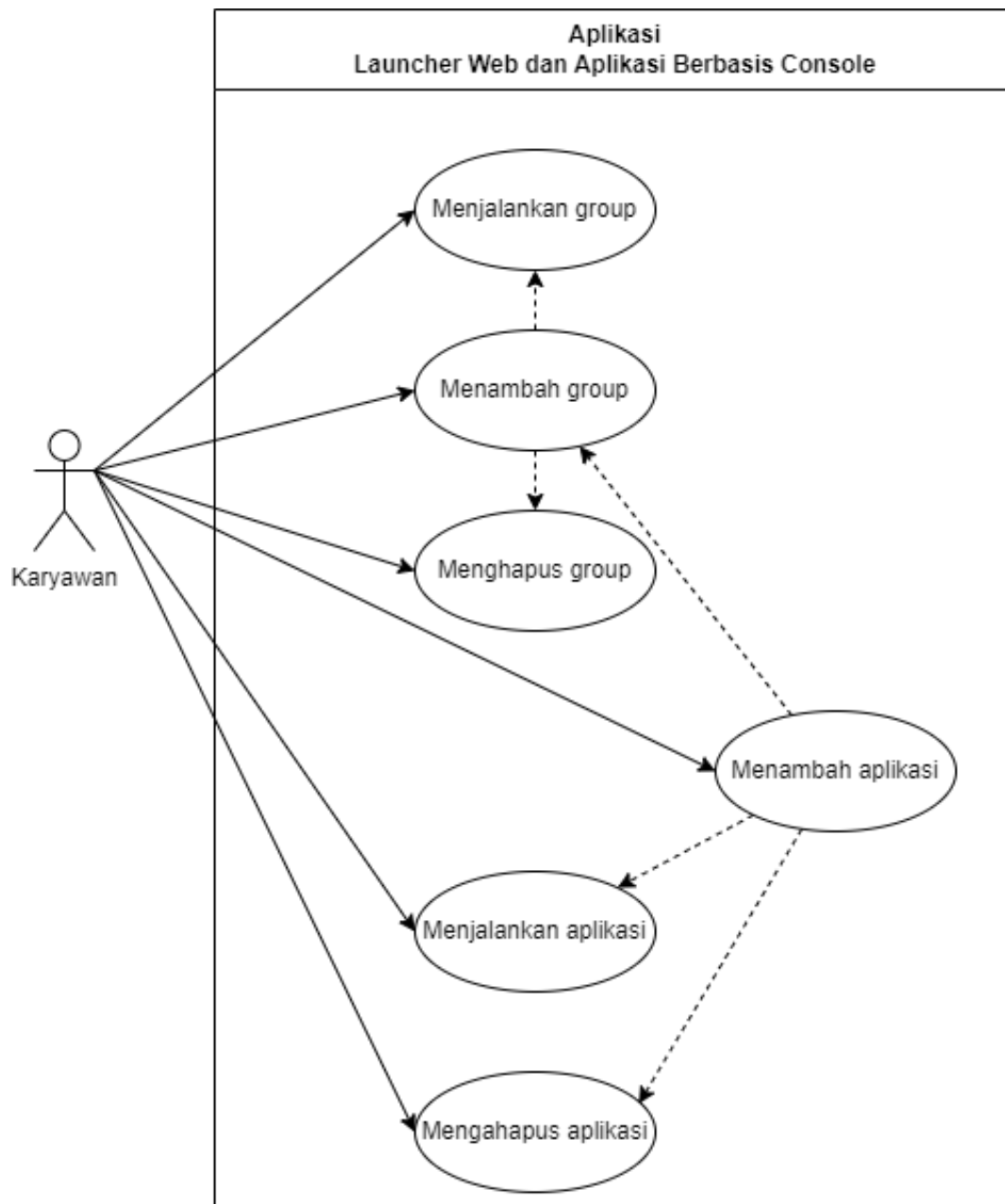
1. Dapat membuka aplikasi dan website;
2. Dapat menambah dan menyimpan link aplikasi dan website yang dibutuhkan karyawan;
3. Dapat membuat sebuah group yang terdiri dari berbagai macam link aplikasi dan website yang jika grup tersebut di panggil maka aplikasi dan website di grup tersebut akan terbuka.
4. Dapat menghapus grup atau link aplikasi dan website.

## Analisis Fitur

No	Fitur	Kebutuhan	Alur
1	Menampilkan menu utama	Menampilkan opsi: 1. Jalankan group 2. Jalankan aplikasi/ website 3. Mengatur group 4. Mengatur aplikasi/ website 5. Keluar	User memilih opsi, dan program mengarahkan ke fungsi terkait. Menu berulang hingga user memilih keluar.
2	Menjalankan group	Untuk membuka group yang berisi kumpulan aplikasi dan software	<b>Pertama:</b> Menampilkan group yang ada  <b>Kedua:</b> jika tidak ada group: pesan “belum ada group yang dibuat ketik b untuk membuat” jika ada group: pesan “ketik nomor group yang ada inginkan”  <b>Ketiga:</b> Karyawan menginput: nomor  <b>Keempat:</b> Aplikasi membuka aplikasi dan website yang ada di group
3	Menjalankan aplikasi	Untuk membuka aplikasi/ website	<b>Pertama:</b> Menampilkan aplikasi yang ada  <b>Kedua:</b> jika tidak ada aplikasi: pesan “belum ada aplikasi yang dibuat ketik b untuk membuat” jika ada aplikasi: pesan “ketik nomor aplikasi yang ada inginkan”  <b>Ketiga:</b> Karyawan menginput: nomor  <b>Keempat:</b> Aplikasi membuka aplikasi

No	Fitur	Kebutuhan	Alur
4	Mengatur group/ website	Terdapat opsi untuk tambah dan hapus group	<p><b>Tambah:</b></p> <p><b>Pertama:</b> Input: nama</p> <p><b>Kedua:</b> Menampilkan semua data aplikasi yang tersimpan Menampilkan pesan “Ketik nomer aplikasi yang ingin anda tambah”</p> <p><b>Ketiga:</b> Karyawan menginput: nomor</p> <p><b>Keempat:</b> Mengirim pesan: “Apakah anda mau menambah aplikasi lain? Jika tidak ketik N jika ya ketik Y”</p> <p><b>Kelima:</b> Karyawan menginput: N</p> <p><b>Keenam:</b> Menampilkan semua data website yang tersimpan Menampilkan pesan “Ketik nomer website yang ingin anda tambah”</p> <p><b>Ketujuh:</b> Karyawan menginput: nomor</p> <p><b>Kedelapan:</b> Mengirim pesan: “Apakah anda mau menambah website lain? Jika tidak ketik N jika ya ketik Y”</p> <p><b>Kesembilan:</b> Karyawan menginput: N</p> <p><b>Hapus:</b></p> <p><b>Pertama:</b> Menampilkan semua data group yang tersimpan Menampilkan pesan “Ketik nomer group yang ingin anda hapus”</p> <p><b>Kedua:</b> Karyawan menginput: nomor</p> <p><b>Ketiga:</b> Mengirim pesan: “Apakah anda apakah anda yakin mau menghapus group ini? Jika tidak ketik N jika ya ketik Y”</p> <p><b>Keempat:</b> Karyawan menginput: Y</p>
5	Mengatur aplikasi	Terdapat opsi untuk tambah dan hapus aplikasi	<p><b>Tambah:</b></p> <p><b>Pertama:</b> Input: nama aplikasi</p> <p><b>Kedua:</b> Input: link aplikasi</p> <p><b>Hapus:</b></p> <p><b>Pertama:</b> Menampilkan semua data aplikasi yang ada Menampilkan pesan “Ketik nomer aplikasi yang ingin anda hapus”</p> <p><b>Kedua:</b> Karyawan menginput: nomor</p> <p><b>Ketiga:</b> Mengirim pesan: “Apakah anda apakah anda yakin mau menghapus aplikasi ini? Jika tidak ketik N jika ya ketik Y”</p> <p><b>Keempat:</b> Karyawan menginput: Y</p>

## Rancangan UML Use Case Diagram



## Rancangan Tampilan Program Aplikasi Console

Tampilan menu utama
Berikut fitur aplikasi yang dapat anda gunakan: 1. Jalankan group 2. Jalankan aplikasi/website 3. Mengatur group 4. Mengatur aplikasi/website 5. Keluar ketik nomor fitur yang ada inginkan: _

Tampilan daftar group(berhasil)	Tampilan daftar group(gagal)
Daftar group: 1. Buat kerja 2. Buat istirahat Ketik nomor group yang anda ingin jalankan: _	Mohon maaf belum ada group yang tersedia

Tampilan daftar group(berhasil)	Tampilan daftar group(gagal)
Daftar group: 1. Discord 2. Unity Ketik nomor group yang anda ingin jalankan: _	Mohon maaf belum ada aplikasi/web yang tersedia

Tampilan menjalankan group
Daftar group: 1. Buat kerja 2. Buat istirahat Ketik nomor group yang anda ingin jalankan: 1 Group berhasil di jalankan.

Tampilan menjalankan aplikasi/website(berhasil)
Daftar aplikasi: 1. Discord 2. Unity Ketik nomor aplikasi/fitur yang anda ingin jalankan: 1 Aplikasi berhasil di jalankan.

Tampilan tambah group
-----------------------

Masukan nama group: Buat kerja  
 Berikut aplikasi yang dapat ditambahkan  
 1. Discord  
 2. Unity  
 Ketik nomer aplikasi yang ingin anda tambah: 1  
 Apakah anda mau menambah aplikasi lain? Jika tidak ketik N jika ya ketik Y: N  
 Berikut website yang dapat ditambahkan  
 1. Texture Haven  
 2. Unity store  
 Apakah mau menambahkan website lain? Jika tidak ketik N jika ya ketik Y: N  
 Group berhasil dibuat

#### Tampilan tambah aplikasi/website

Masukan nama aplikasi: Discord  
 Masukan link path aplikasi axe/lnk : " C:\Users\MSI THIN\Desktop\Discord.lnk"  
 Aplikasi berhasil ditambah

## Analisis Class

### Identifikasi Class dan Attribute

No	Class (Kata Benda)	Attribute (Kata Benda)
1	Group	namaGroup: String
		daftarLinkAplikasi: List<String>
2	Aplikasi	namaAplikasi: String
		linkAplikasi: String
3	MainMenu	fileName: String
		daftarAplikasi: List<Aplikasi>
		aplikasi: Aplikasi
		daftarGroup: List<Group>
		group: Group

### Identifikasi Method

No	Class (Kata Benda)	Method (Kata Kerja)	Parameter (Kata Benda)
1	Group	menambahkanGroup(): Group	daftarAplikasi: List<Aplikasi>
		menjalankanGroup(): void	daftarGroup: List<Group>
		menghapusGroup(): int	daftarGroup: List<Group>
2	Aplikasi	jalankanAplikasi(): void	daftarAplikasi: List<Aplikasi>
		tambahAplikasi(): Aplikasi	-
		menghapusAplikasi(): void	daftarAplikasi: List<Aplikasi>
3	MainMenu	menjalankanMainMenu(): void	-
		inisialisasi(): void	-
		membacaSaveFile(): void	daftarAplikasi: List<Aplikasi>, daftarGroup: List<Group>
		membukaMenu(): void	daftarAplikasi: List<Aplikasi>, daftarGroup: List<Group>, aplikasi: Aplikasi, group: Group
		mengSaveData: (): void	daftarAplikasi: List<Aplikasi>, daftarGroup: List<Group>

### Jenis Relasi dan Alasan

#### a. Dependency

No	Class Awal	Class Tujuan	Alasan
1	Group	Aplikasi	Group hanya menggunakan class Aplikasi sementara di method menambahGroup(), untuk membaca data namaLink. Tidak menyimpan objek Aplikasi secara langsung —



			hanya mengambil linkAplikasi dari daftar dan disimpan dalam List<String> daftarLinkAplikasi.
--	--	--	--

b. Unidirectional Association

No	Class Awal	Class Tujuan	Alasan
1	MainMenu	Group	Karena MainMenu memiliki variabel Aplikasi aplikasi, List<Aplikasi> daftarAplikasi. Artinya MainMenu menyimpan referensi langsung ke objek-objek Aplikasi. Sementara Aplikasi tidak tahu apa itu MainMenu.
	MainMenu	Aplikasi	Karena MainMenu memiliki variabel Group group, List<Group> daftar Group. Artinya MainMenu menyimpan referensi langsung ke objek-objek Group. Sementara Group tidak tahu apa itu MainMenu.

c. Bidirectional Association

Tidak terdapat relasi dua arah antar class. Setiap hubungan antar class hanya berjalan satu arah, misalnya class MainMenu mengenal Aplikasi dan Group, tetapi Aplikasi dan Group tidak mengenal kembali MainMenu.

d. Aggregation

Tidak ada class yang memiliki atribut berupa referensi ke class lain yang dapat eksis secara independen. Semua objek dikelola langsung dalam konteks metode, bukan sebagai bagian dari hubungan “memiliki tetapi independen”.

e. Composition

Tidak terdapat hubungan “bagian dari” yang kuat antara class. Tidak ada class yang bertanggung jawab penuh atas siklus hidup class lain (misalnya, menghapus objek ketika induknya dihapus). Objek dibuat dan dikelola secara terpisah.

f. Generalization

Tidak ada pewarisan antar class dalam program. Semua class berdiri sendiri tanpa menggunakan konsep extends atau superclass-subclass.

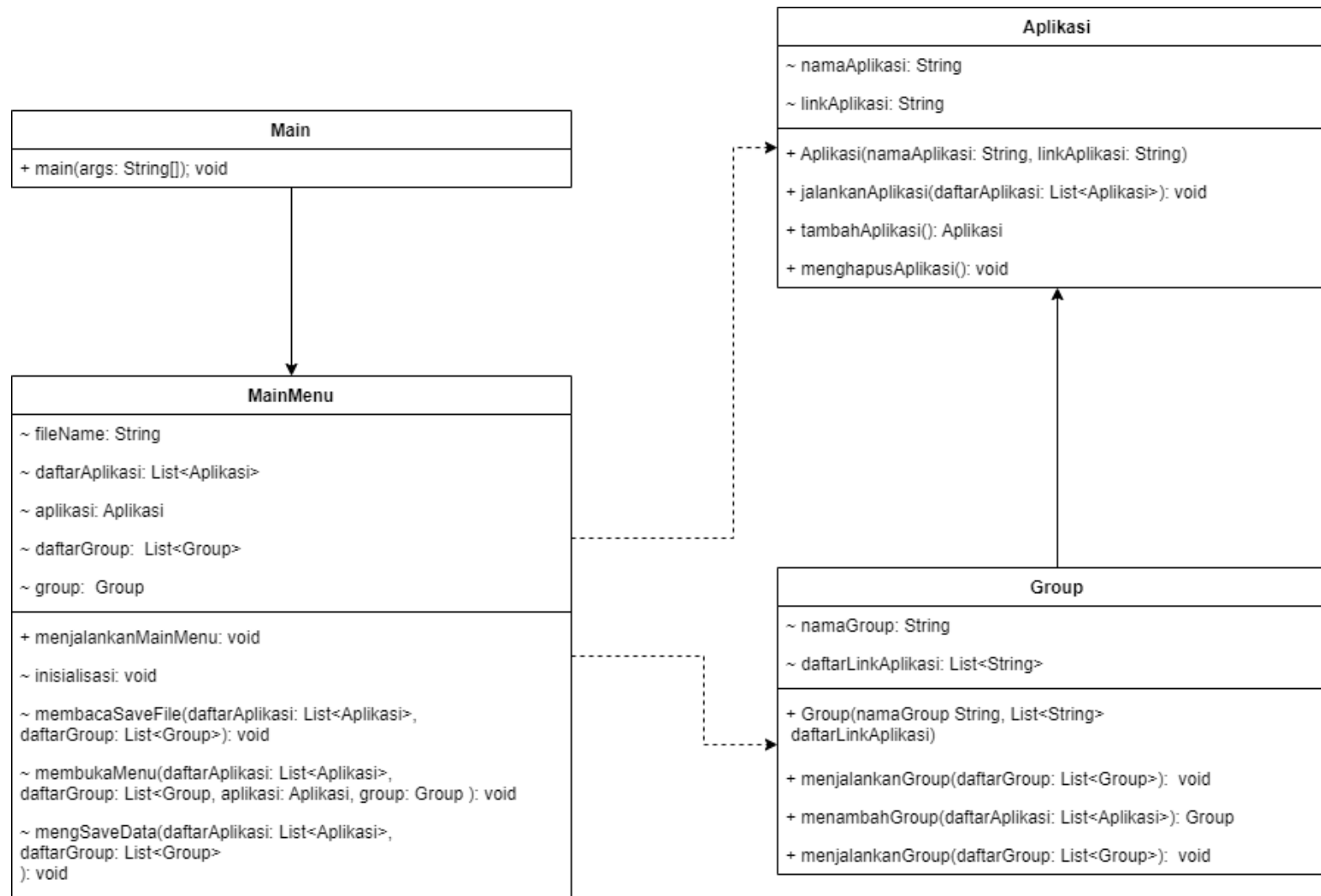
g. Realization

Tidak ada class yang **mengimplementasi** suatu interface, walaupun di class Aplikasi dan Group ada implements tapi itu merupakan bawaan dari pack java, jadi menurut saya tidak termasuk konsep realization.

## Multiplicity

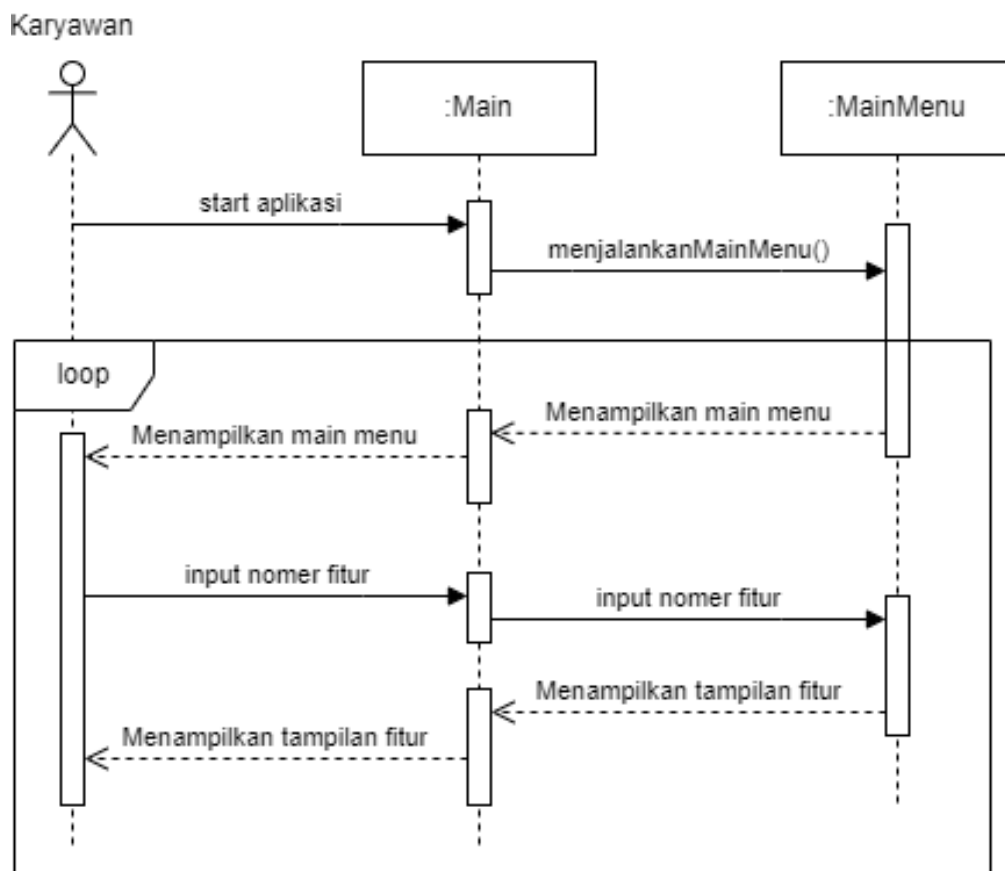
No	Class Awal	Class Tujuan	Relasi	Multiplicity	Keterangan
1	MainMenu	Aplikasi		1 & 1...*	Satu MainMenu <b>mengetahu</b> satu atau banyak Aplikasi
2	MainMenu	Group		1 & 1...*	Satu MainMenu <b>mengetahu</b> satu atau banyak Group
3	Group	Aplikasi		1 & 1...*	Satu Group <b>mengetahu</b> satu atau banyak Aplikasi

## Rancangan UML Class Diagram



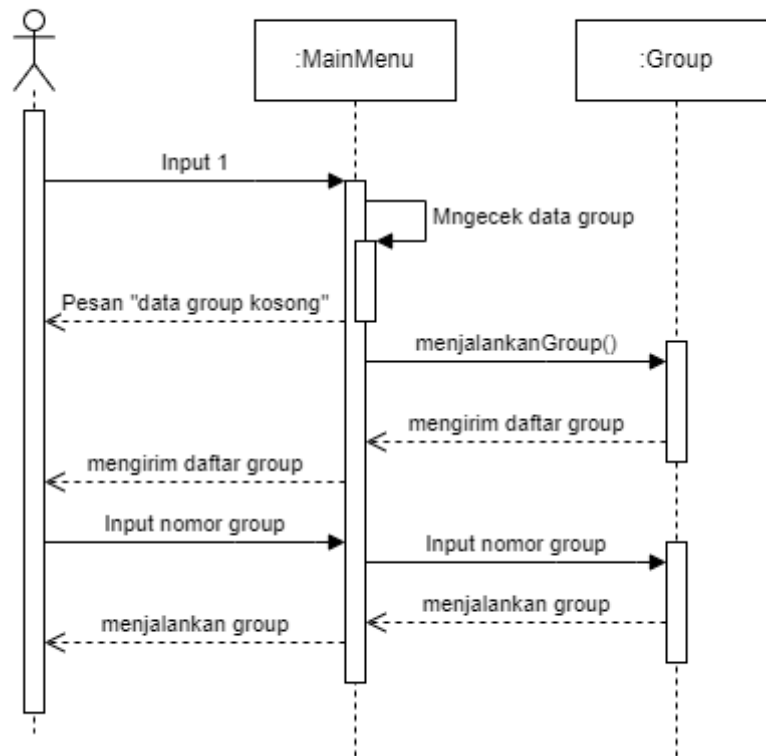
## Rancangan UML Sequence Diagram

Sequence Diagram: main menu

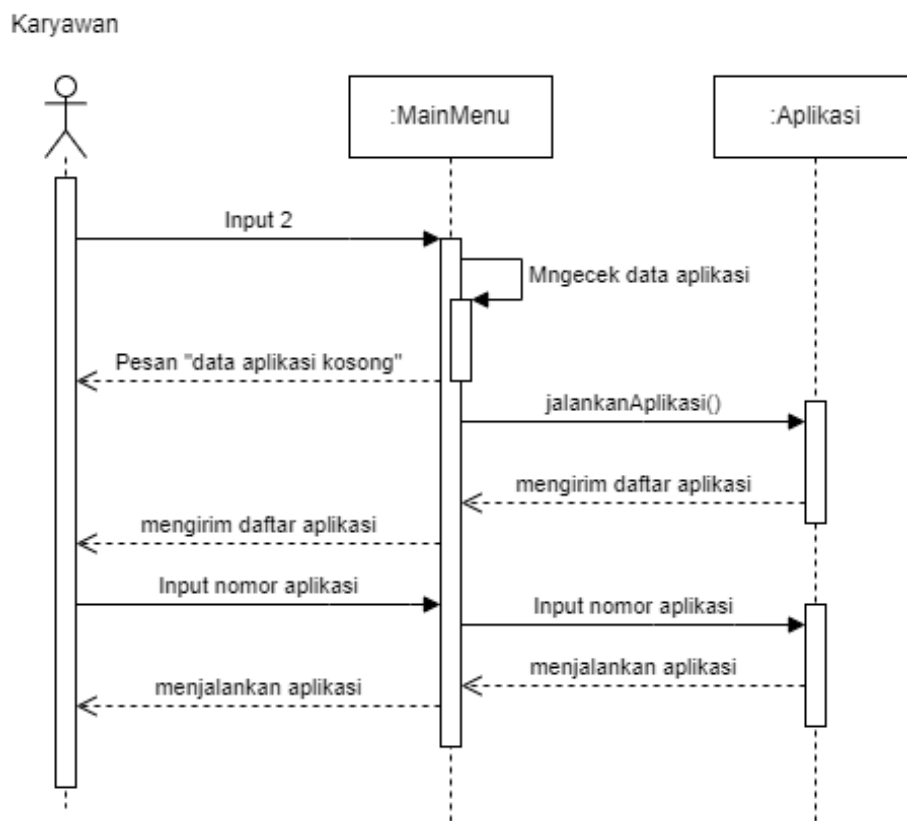


## Sequence Diagram: menjalankan group

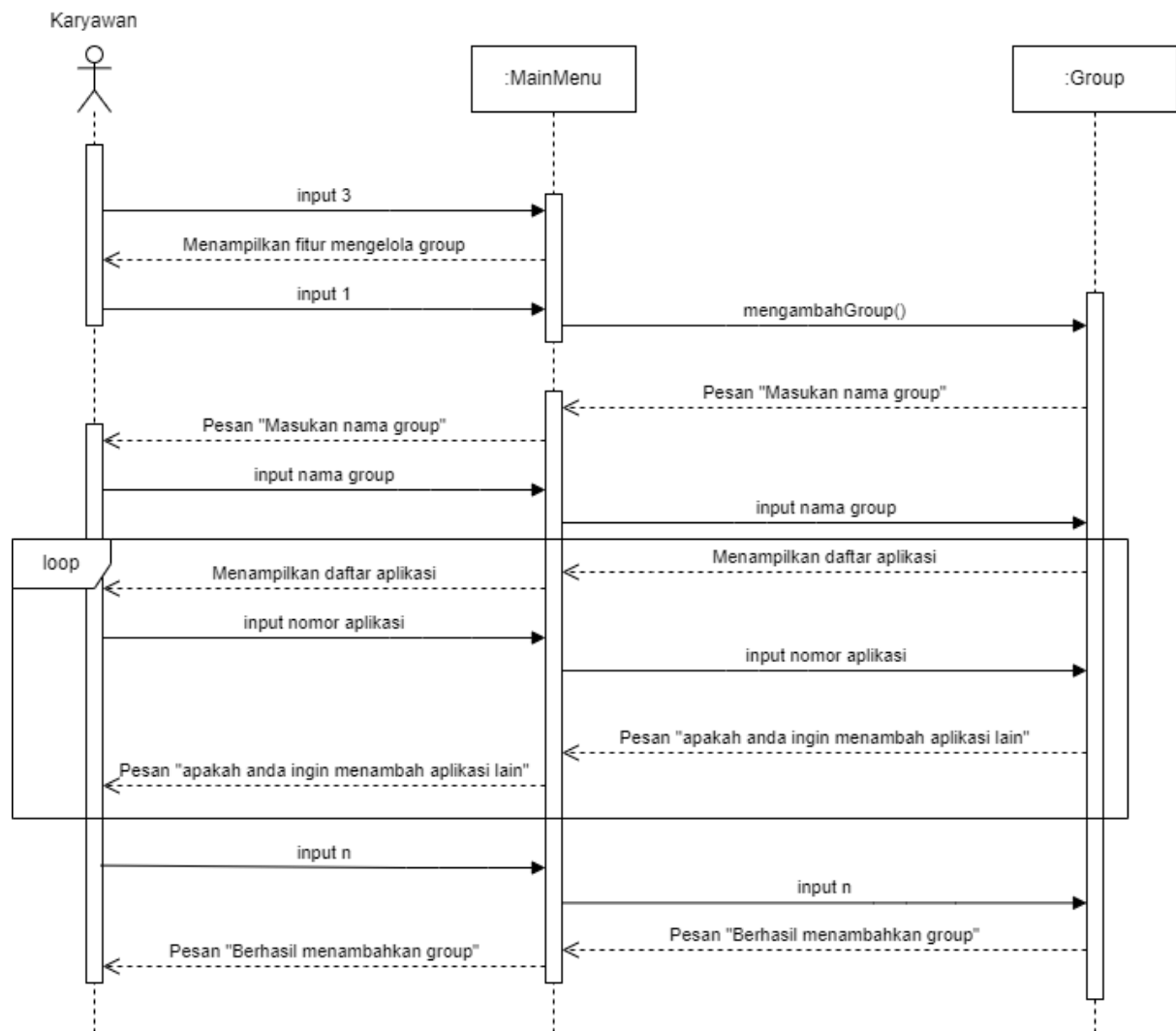
Karyawan



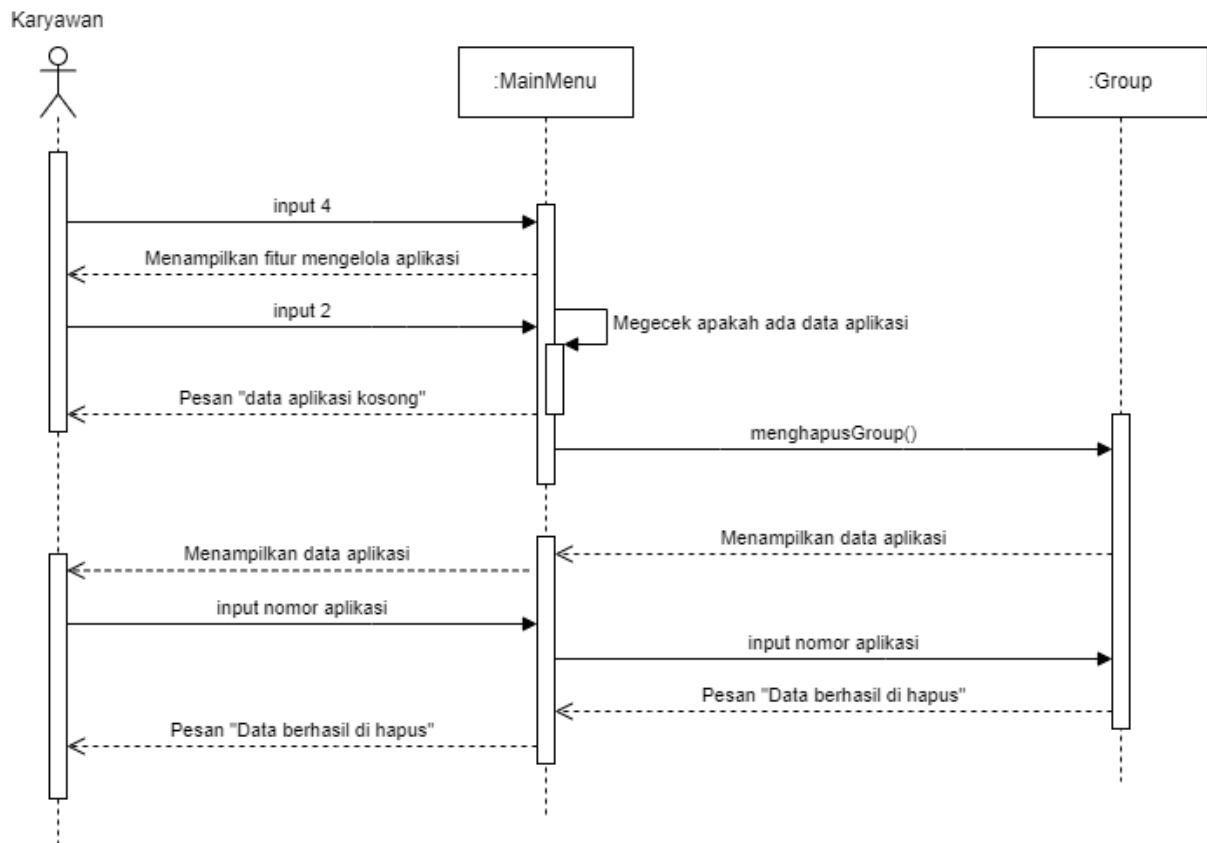
## Sequence Diagram: menjalankan aplikasi



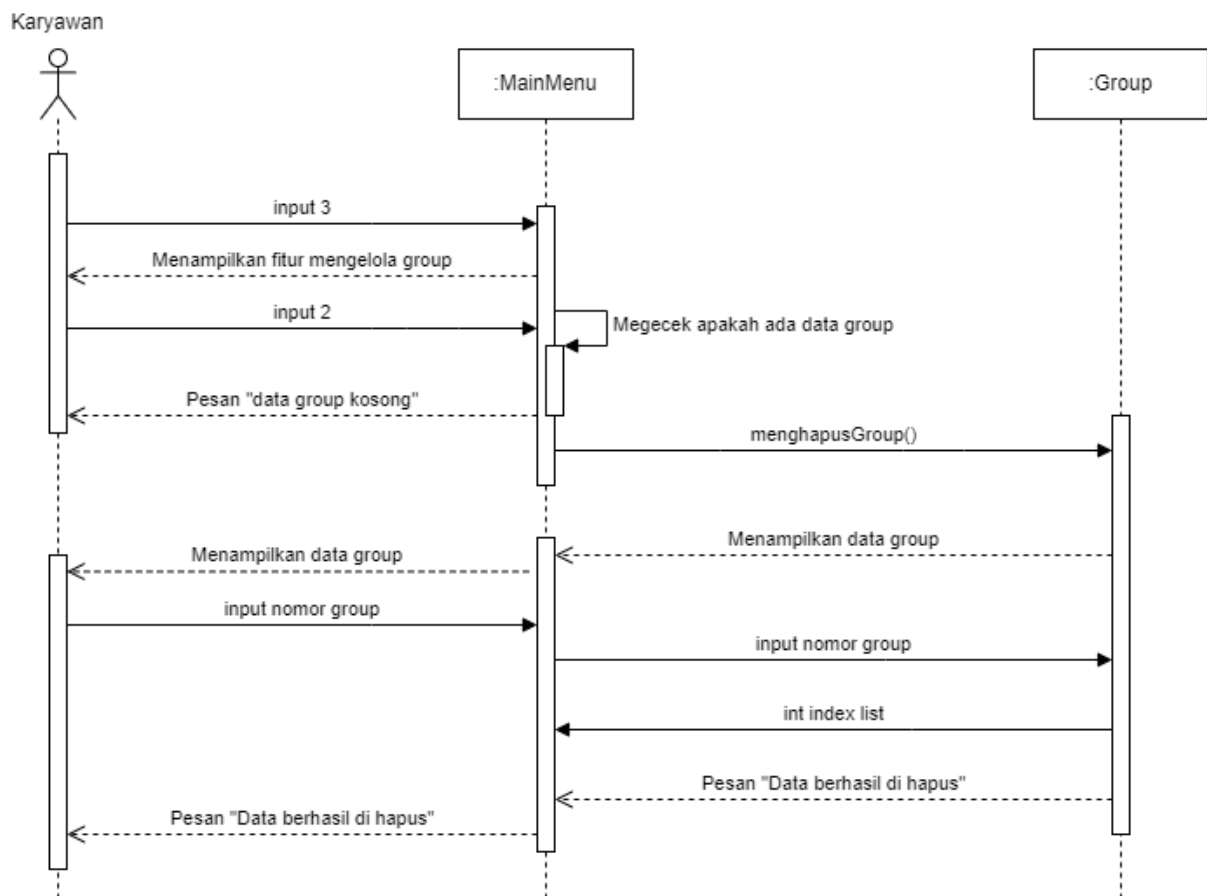
## Sequence Diagram: menambahkan group



## Sequence Diagram: menambahkan aplikasi



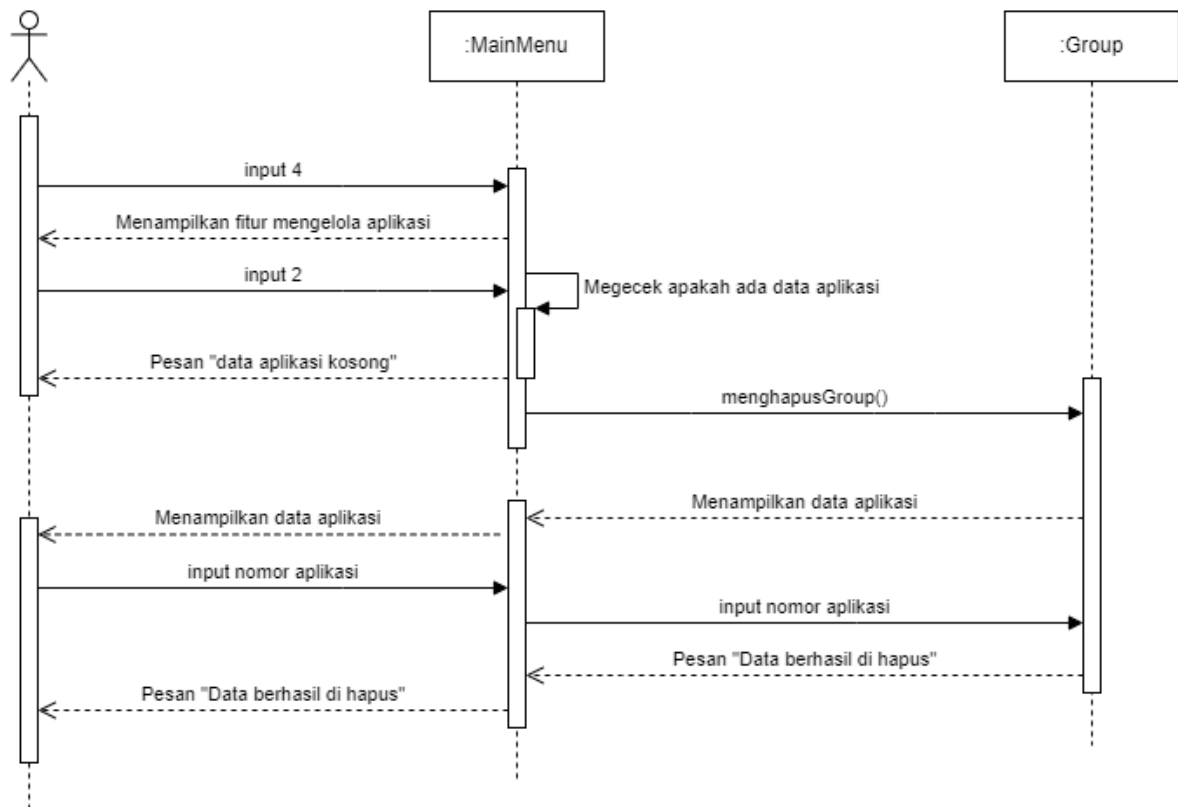
## Sequence Diagram: menghapus group





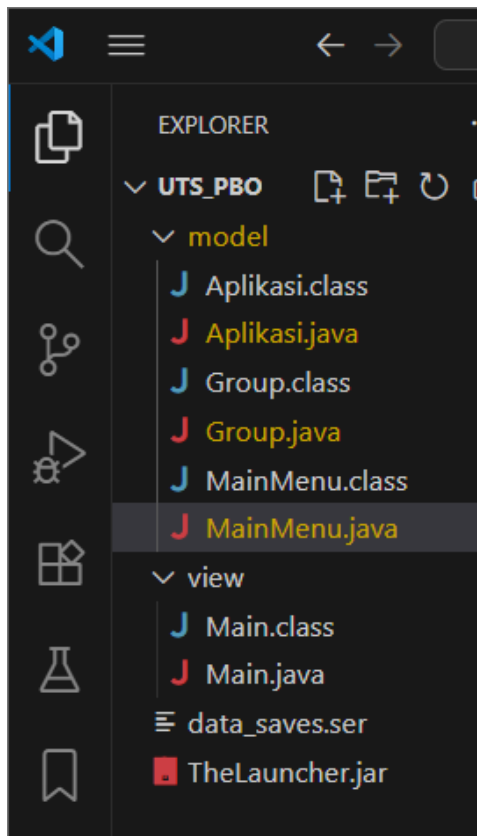
## Sequence Diagram: menghapus aplikasi

Karyawan





## Pra-Kode Program



## Kode Program

### Main.java

```
package view;
import model.MainMenu;
public class Main {
    public static void main(String[] args) {
        MainMenu mainMenu = new MainMenu();
        mainMenu.menjalankanMainMenu();
    }
}
```

Main
+ main(args: String[]); void

### MainMenu.java

```
package model;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
```

```

import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class MainMenu {

    String fileName = "data_saves.ser";

    List<Aplikasi> daftarAplikasi;
    Aplikasi aplikasi;

    List<Group> daftarGroup;
    Group group;

    public void menjalankanMainMenu(){

        inisialisasi();

        membacaSaveFile(daftarAplikasi, daftarGroup);

        membukaMenu(daftarAplikasi, daftarGroup, aplikasi, group);

        mengSaveData(daftarAplikasi, daftarGroup);

    }

    void inisialisasi(){

        List<Aplikasi> daftarAplikasi = new ArrayList<>();
        Aplikasi aplikasi = new Aplikasi("NIRVANA", "FLCL");

        List<Group> daftarGroup = new ArrayList<>();
        Group group = new Group("OASIS", new ArrayList<>());

        this.daftarAplikasi = daftarAplikasi;
        this.aplikasi = aplikasi;

        this.daftarGroup = daftarGroup;
        this.group = group;

    }

    void membacaSaveFile(List<Aplikasi> daftarAplikasi, List<Group> daftarGroup){

        try (ObjectInputStream ois = new ObjectInputStream(new FileInputStream(fileName))) {
            List<Aplikasi> dataBaca1 = (List<Aplikasi>) ois.readObject();
            List<Group> dataBaca2 = (List<Group>) ois.readObject();
            System.out.println("Data hasil dibaca dari file:" + fileName);
            for (Aplikasi m : dataBaca1) {
                daftarAplikasi.add(m);
            }
            for (Group m : dataBaca2) {
                daftarGroup.add(m);
            }
        } catch (IOException | ClassNotFoundException e) {
            //e.printStackTrace();
            System.out.println(".");
        }

    }

    void membukaMenu(List<Aplikasi> daftarAplikasi, List<Group> daftarGroup, Aplikasi
    aplikasi, Group group){

        Scanner input = new Scanner(System.in);

        int pilihan = 99;

        while (pilihan != 5) {

            System.out.println("Daftar fitur yang dapat anda gunakan: ");
            System.out.print("1.\tJalankan group\r\n" + //

```

```

                "2.\tJalankan aplikasi/website\r\n" + //
                "3.\tMengatur group \r\n" + //
                "4.\tMengatur aplikasi/website\r\n" + //
                "5.\tKeluar dan save\r\n" //
            );

            System.out.print("Ketik nomor fitur yang anda ingin jalankan: ");
            pilihan = input.nextInt();

            switch (pilihan) {
                case 1:
                    if (daftarGroup.isEmpty()) {
                        System.out.println("Tidak ada group dalam data, harap tambah group terlebih dahulu");
                    } else {
                        group.menjalankanGroup(daftarGroup);
                    }
                    break;
                case 2:
                    try {
                        daftarAplikasi.get(0).jalankanAplikasi(daftarAplikasi);
                    } catch (Exception e) {
                        System.out.println("Tidak ada aplikasi/website dalam data, harap tambah aplikasi terlebih dahulu");
                    }
                    break;
                case 3:
                    System.out.print("1.\tMenambah group\r\n" + //
                        "2.\tMenghapus group\r\n" +
                        "3.\tMain menu\r\n" + //
                        "4.\tKeluar\r\n");
                    System.out.println("Ketik nomer fitur yang ingin anda gunakan: ");
                    Scanner input78 = new Scanner(System.in);
                    int jawaban78 = input78.nextInt();

                    switch (jawaban78) {
                        case 1:
                            if (daftarAplikasi.isEmpty()) {
                                System.out.println("Tidak ada aplikasi/website dalam data, harap tambah aplikasi/website terlebih dahulu");
                            } else {
                                daftarGroup.add(group.menambahGroup(daftarAplikasi));
                            }
                            break;
                        case 2:
                            if (daftarGroup.isEmpty()) {
                                System.out.println("Tidak ada group dalam data, harap tambah group terlebih dahulu");
                            } else {
                                daftarGroup.remove(group.menghapusGroup(daftarGroup));
                            }
                            break;
                        case 4:
                            System.exit(0);
                            break;
                        default:
                            break;
                    }
                    break;
                case 4:
                    System.out.print("1.\tMenambah aplikasi/website\r\n" + //
                        "2.\tMenghapus aplikasi/website\r\n" +
                        "3.\tMain menu\r\n" + //
                        "4.\tKeluar\r\n");
                    System.out.println("Ketik nomer fitur yang ingin anda gunakan: ");
                    Scanner input5 = new Scanner(System.in);
                    int jawaban = input5.nextInt();

```

```

        switch (jawaban) {
            case 1:
                daftarAplikasi.add(aplikasi.tambahAplikasi());
                break;

            case 2:
                try {
                    daftarAplikasi.get(0).menghapusAplikasi(daftarAplikasi);

                } catch (Exception e) {
                    System.out.println("Tidak ada aplikasi/website dalam data,
harap tambah aplikasi terlebih dahulu");
                }
                break;

            case 4:
                System.exit(0);
                break;

            default:
                break;
        }
        break;
    default:
        break;
    }
}

void mengSaveData(List<Aplikasi> daftarAplikasi, List<Group> daftarGroup){
    try (ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream(fileName)))
    {
        oos.writeObject(daftarAplikasi);
        oos.writeObject(daftarGroup);
        System.out.println("Data berhasil disimpan ke " + fileName);
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}
}

```

MainMenu
~ fileName: String ~ daftarAplikasi: List<Aplikasi> ~ aplikasi: Aplikasi ~ daftarGroup: List<Group> ~ group: Group
+ menjalankanMainMenu: void ~ inisialisasi: void ~ membacaSaveFile(daftarAplikasi: List<Aplikasi>, daftarGroup: List<Group>): void ~ membukaMenu(daftarAplikasi: List<Aplikasi>, daftarGroup: List<Group>, aplikasi: Aplikasi, group: Group ): void ~ mengSaveData(daftarAplikasi: List<Aplikasi>, daftarGroup: List<Group> ): void

## Aplikasi.java

```
package model;

import java.util.Scanner;
import java.io.*;
import java.util.ArrayList;
import java.util.List;

public class Aplikasi implements Serializable {

    String namaAplikasi;
    String linkAplikasi;

    public Aplikasi(String namaAplikasi, String linkAplikasi) {
        this.namaAplikasi = namaAplikasi;
        this.linkAplikasi = linkAplikasi;
    }

    public void jalankanAplikasi(List<Aplikasi> daftarAplikasi) {

        System.out.println("Daftar aplikasi: ");

        int loop = 1;
        for (Aplikasi app : daftarAplikasi) {
            System.out.println( loop + ". " + app.namaAplikasi);
            loop++;
        }

        int totalAplikasi = daftarAplikasi.size();
        int pilihan;

        Scanner input = new Scanner(System.in);
        System.out.print("Ketik nomor aplikasi/website yang anda ingin jalankan: ");
        pilihan = input.nextInt();

        for (int i = 0; i < totalAplikasi; i++) {
            if (pilihan - 1 == i) {

                System.out.println("Menjalankan aplikasi/website: " +
daftarAplikasi.get(i).namaAplikasi);
            }
        }
    }
}
```

```

        try{
            Runtime.getRuntime().exec(new String[] {
                "cmd", "/c", "start", "", daftarAplikasi.get(i).linkAplikasi
            });
        }catch (IOException e) {
            e.printStackTrace();
        }

        break;
    }
}

public Aplikasi tambahAplikasi(){
    Scanner input = new Scanner(System.in);

    String nama;
    String link;

    System.out.println("Masuk nama aplikasi: ");
    nama = input.nextLine();

    System.out.println("Masukan link path aplikasi atau link website: ");
    link = input.nextLine();

    System.out.println("Aplikasi/website berhasil ditambahkan");

    return new Aplikasi(nama, link);
}

public void menghapusAplikasi(List<Aplikasi> daftarAplikasi){

    System.out.println("Daftar aplikasi: ");

    int loop = 1;
    for (Aplikasi app : daftarAplikasi) {
        System.out.println( loop + ". " + app.namaAplikasi);
        loop++;
    }

    int totalAplikasi = daftarAplikasi.size();
    int pilihan;

    Scanner input = new Scanner(System.in);
    System.out.print("Ketik nomor aplikasi yang anda ingin hapus: ");
    pilihan = input.nextInt();

    for (int i = 0; i < totalAplikasi; i++) {
        if (pilihan - 1 == i) {

            String jawaban;
            System.out.println("Apakah anda yakin ingin menghapus aplikasi: " +
            daftarAplikasi.get(i).namaAplikasi + ", Ketik y untuk ya ketik n untuk tidak");
            Scanner input2 = new Scanner(System.in);
            jawaban = input2.nextLine();

            if(jawaban.equals("y")){
                daftarAplikasi.remove(i);
                System.out.println("data aplikasi/website berhasil di hapus");
            }

            break;
        }
    }
}
}

```



--

Aplikasi
~ namaAplikasi: String ~ linkAplikasi: String
+ Aplikasi(namaAplikasi: String, linkAplikasi: String) + jalankanAplikasi(daftarAplikasi: List<Aplikasi>): void + tambahAplikasi(): Aplikasi + menghapusAplikasi(): void

## Group.java

```
package model;

import java.io.IOException;
import java.io.Serializable;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Group implements Serializable {

    String namaGroup;
    List<String> daftarLinkAplikasi = new ArrayList<>();

    public Group(String namaGroup, List<String> daftarLinkAplikasi){

        this.namaGroup = namaGroup;
        this.daftarLinkAplikasi = daftarLinkAplikasi;
    }

    public void menjalankanGroup(List<Group> daftarGroup){

        System.out.println("Daftar group: ");

        int loop = 1;
        for (Group app : daftarGroup) {
            System.out.println( loop + ". " + app.namaGroup);
            loop++;
        }

        int pilihan;

        Scanner input = new Scanner(System.in);
        System.out.print("Ketik nomor group yang ingin anda jalankan: ");
        pilihan = input.nextInt();

        System.out.println("Menjalankan group, dengan aplikasi/website: ");
        loop = 1;
        for (String app : daftarGroup.get(pilihan - 1).daftarLinkAplikasi) {
            System.out.println( loop + ". " + app);
            try{
                Runtime.getRuntime().exec(new String[] {
                    "cmd", "/c", "start", "", app
                });
            }catch (IOException e) {
                e.printStackTrace();
            }
            loop++;
        }
    }
}
```

```

public Group menambahGroup(List<Aplikasi> daftarAplikasi){
    String nama;
    List<String> daftarLink = new ArrayList<>();

    Scanner inputNama = new Scanner(System.in);
    System.out.println("Masukan nama group yang ingin dibuat: ");
    nama = inputNama.nextLine();

    int totalAplikasi = daftarAplikasi.size();
    int pilihan;
    String keluar = "y";

    while(keluar.equals("y")){
        int loop = 1;
        for (Aplikasi app : daftarAplikasi) {
            System.out.println( loop + ". " + app.namaAplikasi);
            loop++;
        }

        Scanner input = new Scanner(System.in);
        System.out.print("Ketik nomor aplikasi/website yang ingin anda tambah: ");
        pilihan = input.nextInt();

        for (int i = 0; i < totalAplikasi; i++) {
            if (pilihan - 1 == i) {

                daftarLink.add(daftarAplikasi.get(i).linkAplikasi);

                System.out.println("Aplikasi/website: " +
daftarAplikasi.get(i).namaAplikasi + " berhasil di tambah");

                break;
            }
        }
        Scanner input2 = new Scanner(System.in);
        System.out.println("apakah anda ingin menambah aplikasi/website lain, ketik y
untuk ya dan ketik n untuk tidak: ");
        keluar = input2.nextLine();
    }

    System.out.println("Group berhasil ditambah");

    return new Group(nama, daftarLink);
}

public int menghapusGroup(List<Group> daftarGroup){

    System.out.println("Daftar group: ");

    int loop = 1;
    for (Group app : daftarGroup) {
        System.out.println( loop + ". " + app.namaGroup);
        loop++;
    }

    int pilihan;

    Scanner input = new Scanner(System.in);
    System.out.print("Ketik nomor group yang ingin anda hapus: ");
    pilihan = input.nextInt();

    System.out.println("Group berhasil di hapus");

    return (pilihan - 1);
}
}

```

--

Group
~ namaGroup: String ~ daftarLinkAplikasi: List<String>
+ Group(namaGroup String, List<String> daftarLinkAplikasi)  + menjalankanGroup(daftarGroup: List<Group>): void + menambahGroup(daftarAplikasi: List<Aplikasi>): Group + menjalankanGroup(daftarGroup: List<Group>): void

## Compile & Run

### Compile

```
PS C:\PBOProject\UTS\UTS_PBO> javac model/* view/*  
Note: model\MainMenu.java uses unchecked or unsafe operations.  
Note: Recompile with -Xlint:unchecked for details.  
PS C:\PBOProject\UTS\UTS_PBO>
```

### Run

```
PS C:\PBOProject\UTS\UTS_PBO> java view/Main  
Data hasil dibaca dari file:data_saves.ser  
Daftar fitur yang dapat anda gunakan:  
1. Jalankan group  
2. Jalankan aplikasi/website  
3. Mengatur group  
4. Mengatur aplikasi/website  
5. Keluar dan save  
Ketik nomor fitur yang anda ingin jalankan:
```

## Testing

### Skenario menjalankan group (Gagal)

```
PS C:\PBOProject\UTS\UTS_PBO> java -jar TheLauncher.jar  
Data hasil dibaca dari file:  
Daftar fitur yang dapat anda gunakan:  
1. Jalankan group  
2. Jalankan aplikasi/website  
3. Mengatur group  
4. Mengatur aplikasi/website  
5. Keluar dan save  
Ketik nomor fitur yang anda ingin jalankan: 1  
Tidak ada group dalam data, harap tambah group terlebih dahulu
```

#### Skenario menjalankan group (Berhasil)

```
Daftar fitur yang dapat anda gunakan:
1. Jalankan group
2. Jalankan aplikasi/website
3. Mengatur group
4. Mengatur aplikasi/website
5. Keluar dan save
Ketik nomor fitur yang anda ingin jalankan: 1
Daftar group:
1. Yanto
Ketik nomor group yang ingin anda jalankan: 1
Menjalankan group, dengan aplikasi/website:
1. "C:\Users\MSI THIN\Desktop\Discord.lnk"
```

#### Skenario menjalankan group (Berhasil)

```
Daftar fitur yang dapat anda gunakan:
1. Jalankan group
2. Jalankan aplikasi/website
3. Mengatur group
4. Mengatur aplikasi/website
5. Keluar dan save
Ketik nomor fitur yang anda ingin jalankan: 1
Daftar group:
1. Yanto
Ketik nomor group yang ingin anda jalankan: 1
Menjalankan group, dengan aplikasi/website:
1. "C:\Users\MSI THIN\Desktop\Discord.lnk"
```

#### Skenario menjalankan aplikasi (Berhasil)

```
menjalankan aplikasi/website: Discord
Daftar fitur yang dapat anda gunakan:
1. Jalankan group
2. Jalankan aplikasi/website
3. Mengatur group
4. Mengatur aplikasi/website
5. Keluar dan save
Ketik nomor fitur yang anda ingin jalankan: 2
Daftar aplikasi:
1. Discord
Ketik nomor aplikasi/website/fitur yang anda ingin jalankan: 1
Menjalankan aplikasi/website: Discord
Daftar fitur yang dapat anda gunakan:
```

### Skenario menjalankan aplikasi (Gagal)

```
1. "C:\Users\MSI THIN\Desktop\Discord.Lnk"
Daftar fitur yang dapat anda gunakan:
1. Jalankan group
2. Jalankan aplikasi/website
3. Mengatur group
4. Mengatur aplikasi/website
5. Keluar dan save
Ketik nomor fitur yang anda ingin jalankan: 2
Tidak ada aplikasi/website dalam data, harap tambah aplikasi terlebih dahulu
```

### Skenario menambah group (Gagal)

```
1. Menambah group
2. Menghapus group
3. Main menu
4. Keluar
Ketik nomer fitur yang ingin anda gunakan:
1
Tidak ada aplikasi/website dalam data, harap tambah aplikasi/website terlebih dahulu
Daftar fitur yang dapat anda gunakan:
1. Jalankan group
2. Jalankan aplikasi/website
3. Mengatur group
4. Mengatur aplikasi/website
5. Keluar dan save
Ketik nomor fitur yang anda ingin jalankan:
```

### Skenario menambah group (Berhasil)

```
1. Menambah group
2. Menghapus group
3. Main menu
4. Keluar
Ketik nomer fitur yang ingin anda gunakan:
1
Masukan nama group yang ingin dibuat:
Buat main game
1. Unity
Ketik nomor aplikasi/website yang ingin anda tambah: 1
Aplikasi/website: Unity berhasil di tambah
apakah anda ingin menambah aplikasi/website lain, ketik y untuk ya dan ketik n untuk tidak:
n
```

### Skenario menghapus group (Gagal)

```
1. Menambah group
2. Menghapus group
3. Main menu
4. Keluar
Ketik nomer fitur yang ingin anda gunakan:
2
Tidak ada group dalam data, harap tambah group terlebih dahulu
```

#### Skenario menghapus group (Berhasil)

```
Ketik nomor fitur yang anda ingin jalankan: 3
1.      Menambah group
2.      Menghapus group
3.      Main menu
4.      Keluar
Ketik nomer fitur yang ingin anda gunakan:
2
Daftar group:
1. Yanto
Ketik nomor group yang ingin anda hapus: 1
Group berhasil di hapus
Daftar fitur yang dapat anda gunakan:
1.      Jalankan group
2.      Jalankan aplikasi/website
3.      Mengatur group
4.      Mengatur aplikasi/website
5.      Keluar dan save
Ketik nomor fitur yang anda ingin jalankan: |
```

#### Skenario menambah aplikasi (Berhasil)

```
1.      Menambah aplikasi/website
2.      Menghapus aplikasi/website
3.      Main menu
4.      Keluar
Ketik nomer fitur yang ingin anda gunakan:
1
Masuk nama aplikasi:
GitHub
Masukan link path aplikasi atau link website:
https://github.com/AkramNail/UTS_PBO/commits/main/
```

#### Skenario menghapus aplikasi (Gagal)

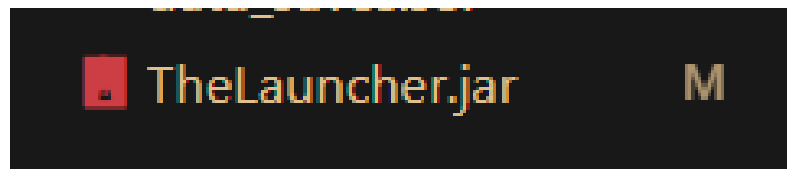
```
Ketik nomor fitur yang anda ingin jalankan: 4
1.      Menambah aplikasi/website
2.      Menghapus aplikasi/website
3.      Main menu
4.      Keluar
Ketik nomer fitur yang ingin anda gunakan:
2
Tidak ada aplikasi/website dalam data, harap tambah aplikasi terlebih dahulu
```

## Skenario menghapus aplikasi (Berhasil)

```
Ketik nomor fitur yang anda ingin jalankan: 4
1. Menambah aplikasi/website
2. Menghapus aplikasi/website
3. Main menu
4. Keluar
Ketik nomor fitur yang ingin anda gunakan:
2
Daftar aplikasi:
1. Unity
2. GitHub
Ketik nomor aplikasi yang anda ingin hapus: 2
Apakah anda yakin ingin menghapus aplikasi: GitHub, Ketik Y untuk ya ketik N untuk tidak
y
data aplikasi/website berhasil di hapus
Daftar fitur yang dapat anda gunakan:
```

## Build (Deploy)

```
PS C:\PBOProject\UTS\UTS_PBO> jar cfe TheLauncher.jar view/Main view/*.class model/*.class
PS C:\PBOProject\UTS\UTS_PBO> |
```



```
PS C:\PBOProject\UTS\UTS_PBO> java -jar TheLauncher.jar
Data hasil dibaca dari file:data_saves.ser
Daftar fitur yang dapat anda gunakan:
1. Jalankan group
2. Jalankan aplikasi/website
3. Mengatur group
4. Mengatur aplikasi/website
5. Keluar dan save
```

## URL Repository

[https://github.com/AkramNail/UTS\\_PBO/](https://github.com/AkramNail/UTS_PBO/)