**LAPORAN TUGAS AKHIR**

**PRAKTIKUM DASAR PEMROGRAMAN**

**SISTEM AKADEMIK BERBASIS COMMAND LINE**

****

Disusun oleh :

1. Annisa Eka Puspita (2341720131)

2. Noklent Fardian Erix (2341720082)

3. Muhamad Syaifullah (2341720013)

**PROGRAM STUDI D4 TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMARI**

**POLITEKNIK NEGERI MALANG**

**SEMESTER GANJIL**

**2023**

# **ABSTRAKSI**

Pada tugas akhir ini, penulis membuat sistem akademik berbasis command line. Sistem ini dibangun menggunakan bahasa pemrograman Java. Sistem ini memiliki fitur-fitur yang dapat digunakan untuk mengelola data akademik, seperti data mahasiswa, data dosen, data mata kuliah, dan data nilai.

Sistem ini dibangun dengan menggunakan struktur data dan algoritma yang sederhana. Hal ini dimaksudkan agar sistem ini mudah dipahami dan dikembangkan. Sistem ini juga dirancang agar dapat digunakan secara interaktif. Hasil pengujian menunjukkan bahwa sistem ini dapat berjalan dengan baik. Sistem ini dapat digunakan untuk mengelola data akademik dengan mudah dan efisien.

Selain itu, sistem ini juga memiliki beberapa fitur tambahan, yaitu: Sistem ini dapat digunakan untuk mencetak laporan akademik, seperti daftar mahasiswa, daftar dosen, daftar mata kuliah, dan daftar nilai. Sistem ini dapat digunakan untuk melakukan import dan export data akademik dari dan ke file. Sistem ini dapat digunakan untuk melakukan backup data akademik. Fitur-fitur tambahan ini dapat mempermudah penggunaan sistem ini dan meningkatkan efisiensi dalam pengelolaan data akademik.

# **KATA PENGANTAR**

Puji syukur kami panjatkan kehadorat Allah SWT atas segala rahmat dan hidah-Nya sehingga kami dapat menyelesaikan Laporan Tugas Akhir Praktikum Dasar Pemrograman Berbasis Command Line. Tujuan dari penyusunan laporan ini adalah sebagai bahan evaluasi hasil pembelajaran Pratikum Dasar Pemrograman.

Laporan ini merupakan salah satu syarat kelulusan mata kuliah Pratikum Dasar Pemrograman bagi Mahasiswa Program Studi Teknik Informatika , Jurusan Teknologi Infor masi Politeknik Negeri Malang.

Kami meyadari bahwa tanpa adanya dukungan dan kerja sama dari berbagai pihak, aktivitas ini tidak dapar berjalan dengan baik, untuk itu kami menyampaikan rasa terima kasih kepada.

1. Mungki Astiningrum, ST., M.Kom selaku Dosen mata kuliah Dasar Pemrograman dan Praktikum Dasar Pemrograman kelas kami.

# **DAFTAR ISI**

**DAFTAR ISI**

[**ABSTRAKSI** i](#_Toc153909536)

[**KATA PENGANTAR** ii](#_Toc153909537)

[**DAFTAR ISI** iii](#_Toc153909538)

[**DAFTAR GAMBAR** iv](#_Toc153909539)

[**BAB I PENDAHULUAN** 1](#_Toc153909540)

[**1.1** **Bahasa Pemrograman Java** 1](#_Toc153909541)

[**1.2 Command Line Interface** 2](#_Toc153909542)

[**BAB II SISTEM AKADEMIK** 3](#_Toc153909543)

[**2.1 Fitur Login** 3](#_Toc153909544)

[**2.2 Admin User Level** 3](#_Toc153909545)

[**2.2.1 Modul Mahasiswa** 3](#_Toc153909546)

[**2.2.1.1 List Mahasiswa** 3](#_Toc153909547)

[**2.2.1.2 Transkip Nilai** 3](#_Toc153909548)

[**2.2.1.3 Presensi Mahasiswa** 3](#_Toc153909549)

[**2.2.1.4 Check Username dan Password Mahasiswa** 3](#_Toc153909550)

[**2.2.2 Modul Dosen** 3](#_Toc153909551)

# **DAFTAR GAMBAR**

[Gambar 1.1 Bahasa Pemrograman Java 1 1](file:///D:\Tugas%20Akhir%20S1\Prakdaspro_Sistem_Adakemik\Other\Laporan\LaporanTA_kelompok2.docx#_Toc153911453)

# **BAB I PENDAHULUAN**

## **Bahasa Pemrograman Java**



Bahasa pemrograman Java telah menjadi salah satu bahasa yang paling berpengaruh dan banyak digunakan dalam dunia pengembangan perangkat lunak. Dikembangkan oleh James Gosling dan timnya di Sun Microsystems pada pertengahan tahun 1990-an, Java dirancang dengan fokus pada portabilitas lintas platform, memungkinkan program Java untuk dijalankan di berbagai sistem operasi tanpa perubahan kode. Keunggulan utama Java dalam hal ini adalah penggunaan bytecode, sebuah bentuk instruksi mesin virtual yang dapat dijalankan oleh Java Virtual Machine (JVM). JVM hadir pada berbagai sistem operasi, yang membuat program Java menjadi platform independen.

Salah satu fitur kunci dari bahasa pemrograman Java adalah paradigma pemrograman berorientasi objek. Dengan konsep objek dan kelas, Java memungkinkan pengembang untuk merancang aplikasi yang modular, mudah dipahami, dan mudah dikembangkan. Selain itu, Java juga dikenal dengan keamanan tingkat tinggi, berkat mekanisme kontrol akses yang diterapkan dalam JVM. Hal ini membuat Java menjadi pilihan populer untuk pengembangan aplikasi enterprise yang memerlukan standar keamanan yang tinggi.

Pada pembelajaran mata kuliah Praktikum dasar Pemrograman ini, kami telah diajari tentang hal hal yang ada pada bahasa pemrograman java. Seperti, Variable, Percabgan, Perulangan, Array, dan fungsi.

## **1.2 Command Line Interface**

Command Line Interface (CLI) adalah suatu antarmuka pengguna yang memungkinkan interaksi dengan sistem atau program melalui baris perintah teks. Berbeda dengan antarmuka pengguna grafis (GUI) yang menggunakan elemen visual seperti ikon dan jendela, CLI memungkinkan pengguna memberikan instruksi dengan mengetikkan perintah teks pada terminal atau baris perintah. Penggunaan CLI sering diutamakan dalam konteks efisiensi dan otomatisasi tugas, di mana pengguna dapat dengan cepat menjalankan perintah atau skrip untuk mengeksekusi tugas tertentu.

Salah satu keunggulan utama dari CLI adalah kemampuannya dalam scripting dan otomatisasi. Pengguna dapat membuat skrip atau file teks yang berisi serangkaian perintah untuk dieksekusi secara berurutan, memungkinkan otomatisasi tugas-tugas rutin. Hal ini menjadikan CLI pilihan yang populer di kalangan pengembang dan administrator sistem yang sering menghadapi tugas-tugas yang membutuhkan repetisi.

CLI umumnya digunakan pada sistem operasi Unix dan Linux, di mana terminal menyediakan akses langsung ke baris perintah. Walaupun CLI memerlukan pemahaman lebih dalam terkait sintaks dan perintah, kelebihannya dalam hal kustomisasi, kecepatan, dan otomatisasi membuatnya tetap menjadi alat yang sangat berguna dalam dunia teknologi informasi dan pengembangan perangkat lunak.

# **BAB II SISTEM AKADEMIK**

## **2.1 Fitur Login**

## **2.2 Admin User Level**

### **2.2.1 Modul Mahasiswa**

### **2.2.1.1 List Mahasiswa**

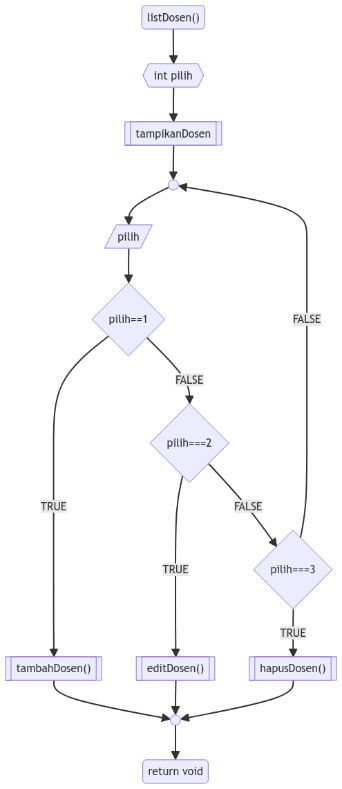
* **Daftarkan Mahasiswa**
* **Edit Data Mahasiswa**
* **Hapus Data Mahasiswa**

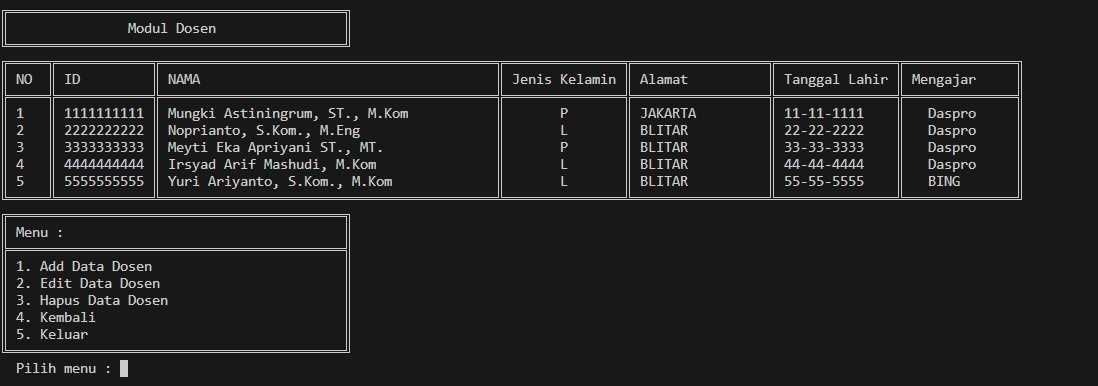
### **2.2.1.2 Transkip Nilai**

### **2.2.1.3 Presensi Mahasiswa**

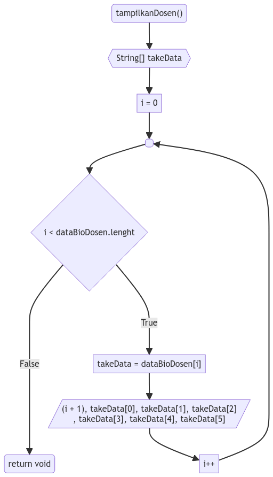
### **2.2.1.4 Check Username dan Password Mahasiswa**

### **2.2.2 Modul Dosen**





DI dalam modul Dosen kami mempunyai 3 fitur yaitu Add data Dosen, Edit Data dosen, Hapus data Dosen. Di dalam Modul dosen ini kami menggunakan fungsi showDataBioDosen untuk menampilkan biodata semua dosen yang ada pada array.



Code :  /\* <--- modulDosen ---> \*/

    static void modulDosen() {

        while (true) {

            renderTitle("Modul Dosen");

            showDataBioDosen(false);

            int userInput = pickMenu("Menu : ", new String[] {

                    "Add Data Dosen",

                    "Edit Data Dosen",

                    "Hapus Data Dosen",

                    "Kembali"

            });

            clearScreen();

            switch (userInput) {

                case 1 -> addDataDosen();

                case 2 -> editDataBioDosen();

                case 3 -> removeDataBioDosen();

                case 4 -> {

                    return;

                }

            }

        }

    }

 // Fungsi showDataBioDosen. Jika id ditemukan, maka akan menampilkan data

    static void showDataBioDosen(boolean isId, String... id) {

        String firstLine = "╔═════╦════════════╦══════════════════════════════════════════╦═══════════════╦═════════════════╦═══════════════╦══════════════╗";

        String middleLine = "╠═════╬════════════╬══════════════════════════════════════════╬═══════════════╬═════════════════╬═══════════════╬══════════════╣";

        String lastLine = "╚═════╩════════════╩══════════════════════════════════════════╩═══════════════╩═════════════════╩═══════════════╩══════════════╝";

        String formatTable = "║ %-3s ║ %-10s ║ %-40s ║       %-7s ║ %-15s ║ %-13s ║   %-10s ║%n";

        System.out.println(firstLine);

        System.out.format(

                "║ NO  ║ ID         ║ NAMA                                     ║ Jenis Kelamin ║ Alamat          ║ Tanggal Lahir ║ Mengajar     ║%n");

        System.out.println(middleLine);

        for (int i = 0; i < bioDosen.length; i++) {

            String[] takeBio = bioDosen[i];

            if (isId && id[0].equals(takeBio[0])) {

                System.out.printf(formatTable, (i + 1), takeBio[0], takeBio[1], takeBio[2], takeBio[3], takeBio[4],

                        takeBio[5]);

                break;

            }

            if (!isId)

                System.out.printf(formatTable, (i + 1), takeBio[0], takeBio[1], takeBio[2], takeBio[3], takeBio[4],

                        takeBio[5]);

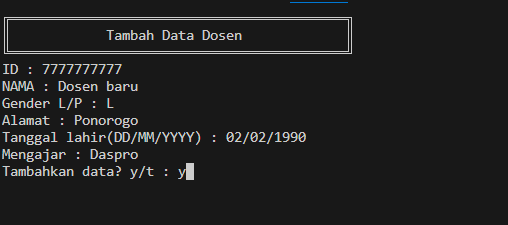
        }

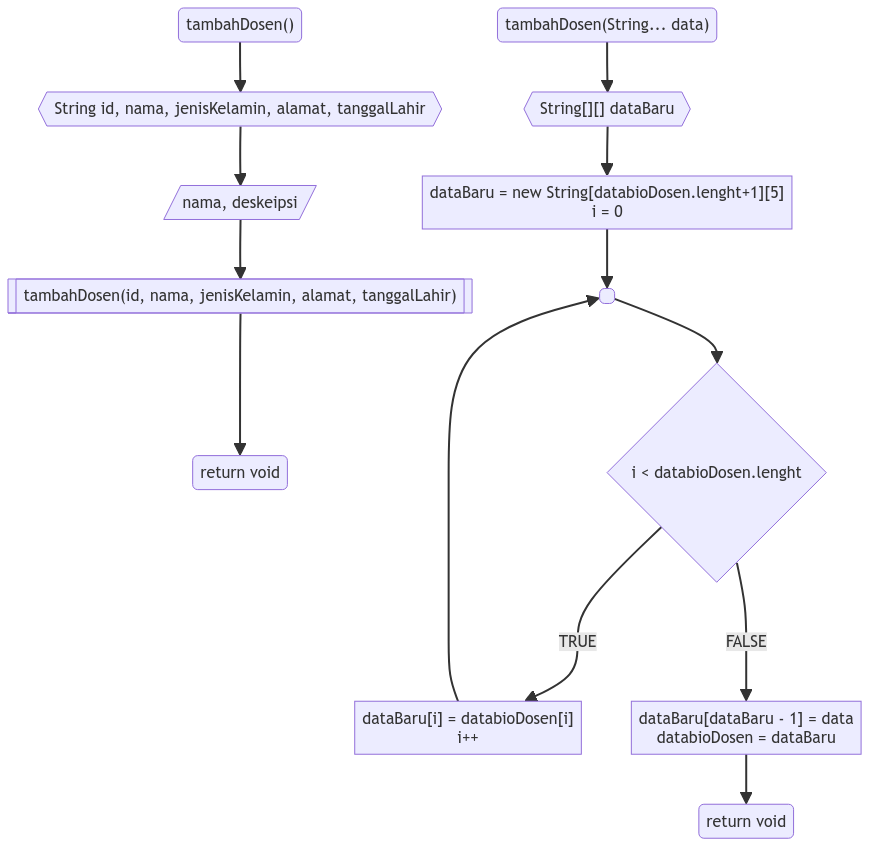
        System.out.println(lastLine);

    }

### 2.2.2.1 Add Data Dosen

Fitur add data dosen akan berfungsi menambahkan data dosen baru ketika kita memasukan Id (Jika id susdah ada maka akan muncul pengecualian), Nama, Gender , dan lainnya





// Fungsi addDataDosen. Menambahkan data dosen

    static void addDataDosen() {

        renderTitle("Tambah Data Dosen");

        String id = getInputStringNumberwithLimitChar("ID", 10, 10, false);

        if (has(bioDosen, id, 0)) {

            System.out.println("ID " + id + " sudah terdaftar");

            return;

        }

        String nama = getInputStringWithLimit("NAMA", 1, 25, false);

        String jenisKelamin = getInputUniqueWord("Gender L/P", 1, 1, true, "l", "p");

        String alamat = getInputStringWithLimit("Alamat", 1, 15, false);

        String tanggalLahir = getInputStringWithLimit("Tanggal lahir(DD/MM/YYYY)", 10, 10, false);

        String mengajar = getInputStringWithLimit("Mengajar", 1, 10, false);

        String userChoose = getInputUniqueWord("Tambahkan data? y/t", 1, 1, true, "y", "t");

        if (userChoose.equalsIgnoreCase("y")) {

            clearScreen();

            addDataDosen(id, nama.toUpperCase(), jenisKelamin.toUpperCase(), alamat, tanggalLahir, mengajar);

            return;

        } else

            System.out.println("Dibatalkan");

    }

    // Fungsi addDataDosen. Menambahkan data dosen

    static void addDataDosen(String... dataBio) {

        String[][] dosenBaru = new String[bioDosen.length + 1][6];

        for (int i = 0; i < bioDosen.length; i++)

            dosenBaru[i] = bioDosen[i];

        dosenBaru[dosenBaru.length - 1] = dataBio;

        bioDosen = dosenBaru;

        String[][] userBaru = new String[userDosen.length + 1][2];

        for (int i = 0; i < userDosen.length; i++)

            userBaru[i] = userDosen[i];

        userBaru[userBaru.length - 1] = new String[] { dataBio[0], dataBio[0] };

        userDosen = userBaru;

        System.out.println("Dosen telah berhasil ditambahkan");

    }