LAPORAN PROYEK  
SISTEM INFORMASI AKADEMIK



MUHAMMAD ERRIL PUTRA PRATIDINA  
NIM: 2341720183

KELAS 1-H  
PROGRAM STUDI D-IV TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG

2023

Daftar Isi

[BAB I PENDAHULUAN 2](#_Toc154818121)

[1.1. Latar Belakang 2](#_Toc154818122)

[1.2. Tujuan 2](#_Toc154818123)

[1.3. Manfaat 2](#_Toc154818124)

[BAB II PEMBAHASAN 3](#_Toc154818125)

[2.1. Pengenalan Program 3](#_Toc154818126)

[2.2. Penjelasan Algoritma Program 4](#_Toc154818127)

[A. Login 4](#_Toc154818128)

[B. Menu 5](#_Toc154818129)

[C. Biodata 6](#_Toc154818130)

[D. Pemilihan Kelas dan Murid 8](#_Toc154818131)

[E. Input Presensi 9](#_Toc154818132)

[F. Input Nilai 9](#_Toc154818133)

[G. Cetak Rapor 12](#_Toc154818134)

[2.3. Demonstrasi Program 13](#_Toc154818135)

[A. Melihat Biodata Murid 13](#_Toc154818136)

[B. Menginputkan Nilai, Presensi, dan Mencetak Rapor 15](#_Toc154818137)

[C. Alur Program Akun Murid 18](#_Toc154818138)

[BAB III PENUTUP 20](#_Toc154818139)

[3.1. Kesimpulan 20](#_Toc154818140)

[Lampiran 22](#_Toc154818141)

1. PENDAHULUAN
   1. Latar Belakang

Pendidikan merupakan salah satu kunci dalam pengembangan individu dan Masyarakat. Di era modern, institusi Pendidikan memiliki peran penting dalam mempersiapkan generasi masa depan. Seiring dengan perkembangan teknologi informasi yang pesat, SIAKAD sangat berperan penting dalam meningkatkan efisiensi, produktivitas, dan kualitas pendidikan.

Sistem Informasi Akademik (SIAKAD) merupakan respons atas berbagai permasalahan dan kebutuhan yang dihadapi oleh institusi pendidikan, baik di tingkat Sekolah Dasar (SD), Sekolah Menengah Pertama (SMP), dan Sekolah Menengah Atas (SMA). Pada project ini kami membuat Sistem Informasi Akademik untuk Sekolah Menengah Atas (SMA). SIAKAD dirancang untuk membantu institusi Pendidikan untuk mengelola administrasi dengan lebih efisien, program ini meliputi mengatur dan menampilkan profil, fitur login, melihat kalender akademik, pengelola presensi, pengelola nilai(guru), murid melihat nilai, dan mencetak rapor. SIAKAD memungkinkan akses mudah dan cepat ke berbagai data akademik. Guru dan siswa dapat dengan mudah mengakses informasi seperti jadwal, nilai, tugas, dan kalender akademik. SIAKAD juga dapat meningkatkan kualitas layanan Pendidikan dengan memungkinkan institusi Pendidikan untuk lebih fokus pada kebutuhan individu siswa, memantau perkembangan siswa dan memberikan umpan balik yang lebih baik.

* 1. Tujuan

Laporan ini ditulis untuk mencapai tujuan-tujuan:

1. Mengomunikasikan hasil kerja proyek akhir semester 1;
2. Mendokumentasikan pembuatan program;
3. Menjelaskan jalannya kode program.
   1. Manfaat

Program ini diharapkan membawa manfaat-manfaat:

1. Memudahkan pengolahan data-data akademik;
2. Mempersingkat *workflow* para pekerja di lingkungan pendidikan dan meningkatkan produktivitas;
3. Mengurangi ketidakkonsistenan data tercatat dengan menjadikan input data tersentralisasi.
4. PEMBAHASAN
5. Pengenalan Program

Sistem Informasi Akademik (SIAKAD) adalah aplikasi berbasis web yang dirancang untuk melakukan proses pengelolaan data akademik dan data terkait lainnya, sehingga seluruh kegiatan akademik dapat terkelola menjadi informasi yang bermanfaat dalam pengelolaan manajemen sekolah menengah atas, pengambilan keputusan, serta pelaporan di lingkungan perguruan tinggi yang melibatkan mahasiswa, dosen, dan petugas administrasi akademik.

Fitur yang ada di dalam project :

1. **Mengatur dan Menampilkan Profil**

SIAKAD mengandung data para guru yang terdiri dari informasi pribadi guru dan murid (nama, tempat dan tanggal lahir, jenis kelamin, dsb.) dan informasi kelas yang dimasuki murid dan guru. Hasilnya, tampilan informasi tersebut dapat dilihat oleh semua pengguna, yakni guru dan murid.

1. **Fitur Login**

Untuk mengakses dan menggunakan sistem ini, pengguna harus memiliki akses masuk yang diberikan oleh petugas administrasi. Akses masuk berupa username dan password yang terpaut pada sebuah akun yang sudah dibuat oleh petugas administrasi. Serangkaian proses masuk sistem ini ialah login. Dengan login, setiap murid dapat dan hanya masuk ke sistem menggunakan akun milik masing-masing.

**Akun**

Semua pengguna, baik guru maupun murid, diberikan akun. Kedua jenis akun tersebut memiliki akses ke fitur yang berbeda-beda. Akun guru atau admin dapat mengakses semua fitur yang disebutkan dalam subbab **2.1.** Akun murid hanya memiliki akses ke **Login, Menampilkan Profil/Biodata, dan Bagan Pengumuman Rapor.**

1. **Pemilihan Murid dan Kelas**

Sebelum masuk pada fitur input nilai, guru harus memilih siswa yang hendak dimasukkan nilainya.

1. **Pengelolaan Presensi**

Sistem Informasi Akademik menyediakan fitur cek hadir dan pengelolaan presensi. Dengan adanya data presensi yang terpusat pada SIAKAD, perhitungan nilai akhir dan pencetakan rapor menjadi lebih mudah.

1. **Pengelolaan Nilai**

Nilai semua mata pelajaran diambil, disimpan, dan diakumulasi secara sentral dan otomatis dengan SIAKAD. Semua input nilai, karena tersimpan secara sentral, dapat langsung dikalkulasi untuk mendapatkan nilai akhir yang akan ditulis pada rapor.

1. **Bagan Pengumuman Rapor untuk Siswa**

Setelah nilai-nilai tugas, ulangan, dan ujian dikalkulasi secara otomatis di siakad, murid dapat melihat nilai mereka di suatu bagan. Bagan rapor memudahkan admin sekolah dalam membagikan rapor. Dengan melalui siakad, pembagian lembaran kertas atau buku rapor serta proses pencetakan *hard copy* dan penataannya tidak diperlukan. Adapun dengan dengan data tersentralisasi pada siakad, risiko tercecer atau tertukarnya rapor menjadi tidak ada.

1. **Pencetakan Rapor**

Setelah semua penilaian dan presensi selesai dikalkulasi, rapor akan dicetak. Rapor menyajikan informasi murid pemilik rapor, nilai akhir dari semua mata pelajaran, total presensi dan total absensi, ruang tanda tangan wali kelas dan wali murid, dan bagan pesan dan kesan dari wali kelas.

1. Penjelasan Algoritma Program
2. Login

Program diawali dengan bagan login. Pengguna diminta untuk memasukkan akun yang sudah dibuatkan oleh admin. Masing-masing individu, baik siswa maupun guru, memiliki akun masing-masing dengan akses fitur yang berbeda. Hal tersebut menghendaki percabangan alur algoritma program ini.

**Akun**

Akun dibuat dengan mendeklarasikan array String berisi username dan password. Setiap isi dari array ini akan melalui percabangan, untuk murid dan guru, yang memiliki alur ke fitur selanjutnya, sesuai dengan jenis akun (murid atau guru).

        do {

            System.out.print("Masukkan username : ");

            username = sc.next();

            System.out.print("Masukkan password : ");

            password = sc.next();

            for (int i = 0; i < akunGuru[0].length; i++) {

                if (username.equals(akunGuru[0][i]) && akunGuru[1][i].equals(password)) {

                    System.out.println("\nAnda berhasil login");

                    loginSuccess = true;

                    nama = akunGuru[0][i];

                    break;

                }

            }

            for (int i = 0; i < akunMurid[0].length; i++) {

                if (username.equals(akunMurid[0][i]) && akunMurid[1][i].equals(password)) {

                    System.out.println("\nAnda berhasil login");

                    loginSuccess = true;

                    nama = biodata[0][i];

                    break;

                }

            }

            if (!loginSuccess) {

                System.out.println("\nUsername atau Password yang Anda masukkan salah!");

                System.out.println("Silahkan mencoba kembali\n");

            }

        } while (!loginSuccess);

1. Menu

Menu merupakan bagan tempat pengguna dapat memilih dua fitur. Akun murid dan guru memiliki akses yang berbeda, sehingga fitur yang ditampilkan dalam menu juga berbeda. Akun murid berisi menu Lihat Biodata dan Lihat Rapor, sedangkan akun guru berisi menu Lihat Biodata, Input Nilai, dan Cetak Rapor. **Akun murid baru ditambahkan pada revisi pasca-presentasi dasar program.**

**Menu untuk Akun Guru**

    static void Menu (String menu, String murid[][], String kelas[], String nama, String pilihKelas) {

        System.out.println("Biodata\nKelas");

        System.out.print("Pilih menu Anda : \n");

        menu = sc.next();

        switch (menu) {

            case "biodata":

                Biodata(false, murid, pilihKelas, pilihKelas, kelas);

                break;

            case "kelas":

                Kelas(kelas);

                break;

        }

**Menu untuk Akun Murid**

                System.out.println("Biodata\nRapor");

                System.out.print("Silahkan pilih menu Anda : \n");

                menu = sc.next();

                if (menu.equalsIgnoreCase("biodata")) {

                    biodata();

                    }

                    System.out.println("Apakah Anda ingin melihat rapor?(y/n)");

                    pilih = sc.next();

                    if (pilih.equals("y")) {

                        cetakRapor(nama, pilih, i, i, i, i, i, i, i, i, i, i, i, i, i, i, i);

                    }

                } else if (menu.equalsIgnoreCase("rapor")) {

                    cetakRapor(nama, pilih, i, i, i, i, i, i, i, i, i, i, i, i, i, i, i);

                }

                System.out.println("Apakah Anda ingin kembali ke laman awal?(y/n)");

                pilih = sc.next();

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

                    loginSuccess = false;

                    login(loginSuccess, pilih, menu, username, password, password, akunGuru, akunMurid);

                } else if (pilih.equals("n")) {

                    loginSuccess = false;

                    break;

                }

            }

1. Biodata

Setelah percabangan pada **Menu**, jika pengguna memilih biodata, program akan memanggil fungsi **Biodata**.

public static void Biodata(boolean loginSuccess, String murid[][], String pilih, String nama, String kelas[]){

Setiap elemen data, yakni jenis data tersimpan seperti nama, kelas, dan tanggal lahir, dicetak melalui sebuah algoritma perulangan. Hal ini untuk memanfaatkan perulangan dan array, sehingga penjalanan kode program menjadi otomatis. Integer **j** digunakan untuk mengindeks kolom elemen yang berisi elemen-elemen data siswa. Array **biodata[j][i-1]** juga berada di dalam perulangan. Integer **i** mengindeks array sesuai dengan nomor induk siswa (value dari **i** pula).

            for (int j = 0; j < biodata.length; j++) {

                switch (j) {

                    case 0:

                        System.out.print("Nama Lengkap\t\t: ");

                        break;

                    case 1:

                        System.out.print("Jenis Kelamin\t\t: ");

                        break;

                    case 2:

                        System.out.print("Tempat Lahir\t\t: ");

                        break;

                    case 3:

                        System.out.print("Tanggal Lahir\t\t: ");

                        break;

                    case 4:

                        System.out.print("Agama\t\t\t: ");

                        break;

                    case 5:

                        System.out.print("Alamat\t\t\t: ");

                        break;

                    case 6:

                        System.out.print("Nama Ayah\t\t: ");

                        break;

                    case 7:

                        System.out.print("Nama Ibu\t\t: ");

                        break;

                    case 8:

                        System.out.print("Pekerjaan Ayah\t\t: ");

                        break;

                    case 9:

                        System.out.print("Pekerjaan Ibu\t\t: ");

                    default:

                        break;

                }

                System.out.print(biodata[j][(i-1)]+"\n");

}

Di akhir fitur, terdapat pertanyaan kepada pengguna apakah ingin kembali ke **Menu** atau logout dari program.

            System.out.print("\nIngin melihat siswa lain? (y/n) : ");

            char isLagi = sc.next().charAt(0);

            if (isLagi == 'n') {

                isEnd = true;

                System.out.print("Apakah Anda ingin kembali ke menu?(y/n)");

                pilih = sc.next();

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

                    Menu(nama, murid, kelas, nama, pilih);

                } else if (pilih.equalsIgnoreCase("n")) {

                    System.out.print("Apakah Anda ingin logout?(y/n)");

                    pilih = sc.next();

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

                        siakad(loginSuccess, pilih, isLagi, i, i, isLagi, i, i, null, null, null, i, isLagi, i, i, i, isLagi, i, isLagi, null, null, null, i, i, i, isLagi, i, i, isLagi, i);

                    }

                }

            }

        }

    }

1. Pemilihan Kelas dan Murid

Fitur kedua yang ada pada menu adalah pemilihan kelas dan murid, yang merupakan fitur langkah-awal menuju fitur-fitur lain seperti input nilai, input presensi, dan cetak rapor. Program di bawah berfungsi untuk menampilkan kelas-kelas yang ada.

        System.out.println("Berikut kelas yang ada di sekolah ini : ");

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

            System.out.print(kelas[i] + "\t");

        }

        System.out.println();

Di bawah adalah algoritma pengecekan input kelas yang dihendaki pengguna.

        System.out.print("Silahkan pilih kelas : ");

        pilihKelas = sc.next();

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

            if (pilihKelas.equalsIgnoreCase(kelas[i])) {

                int j = 0;

                plusMurid(j, murid, i);

            }

        }

Fungsi **plusMurid**  berfungsi menunjukkan daftar murid yang ada dalam kelas yang dipilih pengguna.

    static void plusMurid (int j, String murid[][], int i) {

        if (j < murid[i].length) {

            System.out.println(murid[i][j]);

            j++;

            plusMurid(j, murid, i);

        }

    }

Selanjutnya algoritma pemilihan murid berdasarkan nomor induknya.

        System.out.print("Silahkan pilih nomor induk siswa : ");

        nama = sc.next();

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

            for (int j = 0; j < murid[i].length; j++) {

                if (nama.equalsIgnoreCase(murid[i][j])) {

                    System.out.print("\n" + murid[i][j] + "\n\n");

                    int k = murid[i][j].length();

                    nama = biodata[0][k-1];

                }

            }

        }

1. Input Presensi

Presensi diinputkan tepat setelah pengguna memilih murid. Algoritma input dilengkapi dengan loop untuk mencegah input yang invalid.

        //PRESENSI

        System.out.println("Masukkan jumlah kehadiran siswa dalam satu semester (maksimal 132 hari) : ");

        double presensi;

        do {

            presensi = sc.nextDouble();

            if (presensi > 132) {

                System.out.println("\nAnda memasukkan jumlah yang tidak valid.\nSilakan coba lagi.");

                System.out.println("\nMasukkan jumlah kehadiran siswa dalam satu semester (maksimal 132 hari) : ");

                }

        } while (presensi > 132);

Presensi dikalkulasi dengan persentase. Dalam satu semester, diketahui terdapat 132 hari sekolah. Dengan , didapatkan keluaran berupa persentase kehadiran selama satu semester.

jmlPresensi = (presensi/132)\*100;

        //System.out.printf("Persentase kehadiran siswa dalam 1 semester : %.1f\n", jmlPresensi);

        double totalNilaiAkhir=0, totalBobotMapel=0;

        for (int i = 0; i < banyakMapel; i++) {

            totalNilaiAkhir =+ nilaiA[i];

            totalBobotMapel =+ bobotMapel[i];

        }

1. Input Nilai

**Input nilai mendapatkan revisi pasca-presentasi daspro.** Terdapat banyak variabel yang dideklarasikan secara global. Hal tersebut untuk memudahkan pemanggilan nilai variabel pada fungsi-fungsi yang memerlukan, seperti fungsi **KalkulasiNilai()** dan **CetakRapor()**.

Variabel bobot mata pelajaran **bobotMapel[]** menyimpan nilai bobot atau sistem kredit semester. Sebagai contoh, semua bobot mata pelajaran diinisiasi dengan nilai integer 4. Kemudian terdapat kelompok proporsi dari tugas, ulangan harian, dan ujian (**proporsiTugas, proporsiUH, proporsiU**). Proporsi digunakan untuk menentukan besar pengaruh suatu jenis nilai pada nilai akhir suatu mapel (nilai yang ditulis pada rapor). Terdapat variabel **banyakMapel**. Variabel ini dapat diubah oleh pemrogram untuk menentukan berapa banyak mata pelajaran yang nilainya akan diinputkan. Nilai variabel tersebut juga menentukan berapa banyak pengulangan fungsi kalkulasi nilai dipanggil. Nama-nama mata pelajaran juga inisiasi pada **mapel[]** dengan panjang menyesuaikann **banyakMapel**.

Variabel **tugas[][]**, **uH[][]**, dan **ujian[]** merupakan variabel tempat menyimpan nilai. Indeks baris dialokasikan untuk nilai masing-masing mapel. Sedangkan indeks kolom dialokasikan untuk nilai-nilai pada mapel yang sama. Variabel **nilaiA[]** adalah tempat penyimpanan nilai akhir yang nantinya akan dicetak pada rapor.

    static double proporsiTugas   = 0.3;

    static double proporsiUH      = 0.35;

    static double proporsiU       = 0.35;

    static Scanner sc = **new** Scanner(System.in);

    // Deklarasi untuk inp-kalk nilai

        // Masukkan banyak mapel

    static int banyakMapel = 8;

    static double tugas[][] = **new** double[banyakMapel][4];

    static double uH[][]    = **new** double[banyakMapel][4];

    static double ujian[]   = **new** double[banyakMapel];

    static String mapel[]   = {"Matematika", "Ilmu Pengetahuan Alam", "Bahasa Indonesia", "Bahasa Inggris", "Ilmu Pengetahuan Sosial", "Pendidikan Pancasila dan Kewarganegaraan", "Pendidikan Agama dan Budi Pekerti", "Seni Budaya"};

    static double nilaiA[]  = **new** double[banyakMapel];

    static double rata[][]  = **new** double[banyakMapel][2];

    static int bobotMapel[] = {4,4,4,4,4,4,4,4};

Fungsi **KalkulasiNilai()** ditulis di dalam loop dengan perulangan sebanyak jumlah mata pelajaran yang ada.

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

           nilaiA[i] = KalkulasiNilai(i);

        }

**Penginputan nilai tugas dan ulangan harian.** Langkah pertama algoritma di dalam fungsi kalkulasi ialah input nilai tugas. Perulangan dua lapisan digunakan untuk mengerjakan pekerjaan ini. Perulangan lapisan dalam digunakan untuk mengulang penginputan nilai tugas. Perulangan akan berulang sebanyak jumlah tugas dalam satu semester, jumlah yang sudah diinisiasikan sebelumnya pada variabel global. Perulangan lapisan dalam merupakan mekanisme pencegahan jika terdapat input yang invalid. Model algoritma ini juga diterapkan pada penginputan nilai ulangan harian.

        int j =0;

        System.out.println("===== "+mapel[i]+" =====");

        do {

            System.out.print("\nNilai Tugas "+mapel[i] +"\n Tugas ke-"+(j+1)+" : ");

            do {

                tugas[i][j] = sc.nextDouble();

                if (tugas[i][j] < 0 || tugas[i][j] > 100) {

                    System.out.println("Anda memasukkan nilai yang tidak valid. \nSilahkan coba lagi.");

                System.out.print("\nNilai Tugas "+mapel[i]+" \n Tugas ke-"+(j+1)+" : ");

                }

            } while (tugas[i][j] < 0 || tugas[i][j] > 100);

            j++;

        } while (j<tugas[i].length);

**Penginputan nilai ujian.** Hal yang sama juga diterapkan pada penginputan nilai ujian. Namun, nilai ujian hanya diinputkan satu kali atau tidak menggunakan perulangan berlapis.

        j=0;

        System.out.print("\nNilai Ujian "+mapel[i]+" : ");

        do {

            ujian[i] = sc.nextDouble();

            if (ujian[i] < 0 || ujian[i] > 100) {

                System.out.println("Anda memasukkan nilai yang tidak valid. \nSilahkan coba lagi.");

                System.out.print("Nilai Ujian "+mapel[i]+" : ");

            }

        } while (ujian[i] < 0 || ujian[i] > 100);

**Kalkulasi nilai.** Nilai yang diinputkan kemudian dikalkulasi. Langkah awal ialah menghitung rata-rata nilai dalam satu baris array milik mapel tersebut. Sehingga didapatkan rata-rata nilai tugas dan nilai ulangan harian. Variabel **rata[i][0]** dan **rata[i][1]** secara berurutan menyimpan rata-rata nilai tugas dan ulangan harian.

                //rata-rata nilai dalam array

        double total=0;

        for(j=0; j<tugas[i].length; j++){

            total += tugas[i][j];

        }

        rata[i][0] = total/tugas[i].length;

        total=0;

        for(j=0; j<uH[i].length; j++){

            total+= uH[i][j];

        }

        rata[i][1]= total/uH[i].length;

**Kalkulasi nilai akhir dan penentuan remedi.** Terakhir adalah kalkulasi nilai akhir mapel dan keputusan remedi atau tidak. Variabel **nilaiAkhir** digunakan untuk menyimpan hasil akhir kalkulasi, yang akan dicetak pada rapor.

Kemudian pemilihan digunakan untuk menentukan apakah murid tersebut memerlukan remedi atau tidak.

        double nilaiAkhir = (rata[i][0]\*proporsiTugas+ujian[i]\*proporsiU+rata[i][1]\*proporsiUH)\*bobotMapel;

        System.out.print("Hasil Nilai Akhir "+mapel[i]+" : " + nilaiAkhir);

        if (nilaiAkhir<75) {

            System.out.println("\nRemedi.");

        } else {

            System.out.println("\nTidak Remedi.");

        }

        return nilaiAkhir;

1. Cetak Rapor

Cetak rapor menampilkan semua nilai variabel yang sudah diinputkan, yakni nilai akhir setiap mata pelajaran, persentase kehadiran, nama dan kelas murid, dan nilai rata-rata.

        System.out.println("Nama Siswa\t: " + nama);

        System.out.println("Kelas\t\t: " + pilihKelas);

        System.out.println("\n============================== NILAI ==============================");

        for (int i = 0; i < banyakMapel; i++) {

            cetakMapel(i);

        }

        System.out.println("\n============================== KEHADIRAN ==============================");

        System.out.printf("Presentase Kehadiran : %.1f\n ", jmlPresensi);

        System.out.println("\n============================== HASIL ==============================");

        System.out.println("Indeks Prestasi\t\t: " + indeksPrestasi);

        System.out.println("Rata-rata Nilai\t\t: " + rataRata);

        kelulusan(indeksPrestasi, jmlPresensi);

Untuk nilai-nilai mata pelajaran, pencetakan diletakkan dalam fungsi. Hal ini agar program dapat menyesuaikan banyaknya cetakan berdasarkan jumlah mata pelajaran.

    static void cetakMapel(int i){

        System.out.println("Mata Pelajaran : " + mapel[i]);

        System.out.printf("Tugas\t\t: %.1f\n", rata[i][0]); // rata[i][0] untuk rata2 tugas

        System.out.printf("UH\t\t: %.1f\n", rata[i][1]); // rata[i][1] untuk rata2 uh

        System.out.printf("Ujian\t\t: %.1f\n", ujian[i]);

        System.out.printf("Nilai Akhir\t: %.1f\n", nilaiA[i]);

    }

Terdapat pula mesin penentu kelulusan yang memiliki keluaran apakah murid lulus atau tidak.

    static void kelulusan(double indeksPrestasi, double jmlPresensi){

        if(indeksPrestasi >= 2.5 && jmlPresensi >= 70)  {

                System.out.println("\nSiswa memenuhi syarat untuk naik kelas.\n");

        } else {

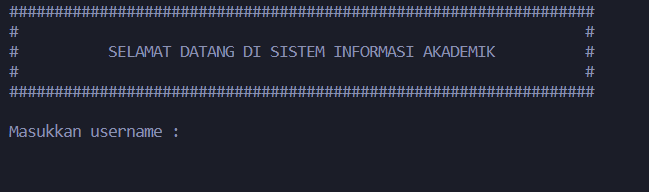
            System.out.println("\nSiswa tidak memenuhi syarat untuk naik kelas.\n");

        }

    }

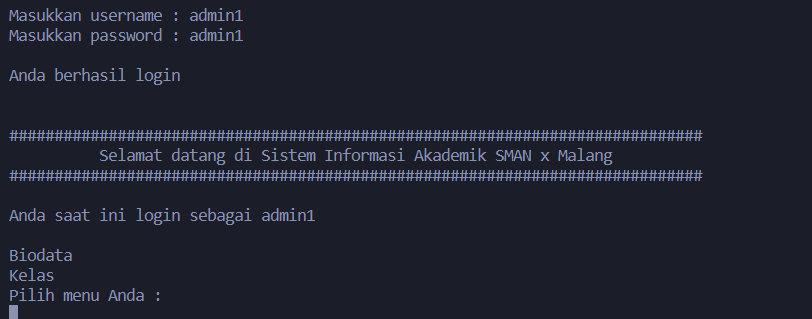
1. Demonstrasi Program
2. Melihat Biodata Murid

Tampilan awal program.



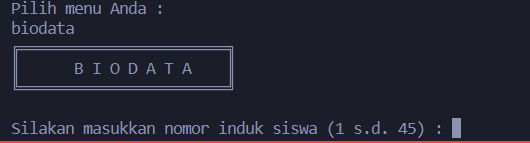
Gambar 1

Masukkan username dan password admin. Sementara terdapat dua akun admin, yakni **admin1** dan **admin2**. Keduanya memiliki akses fitur yang sama.



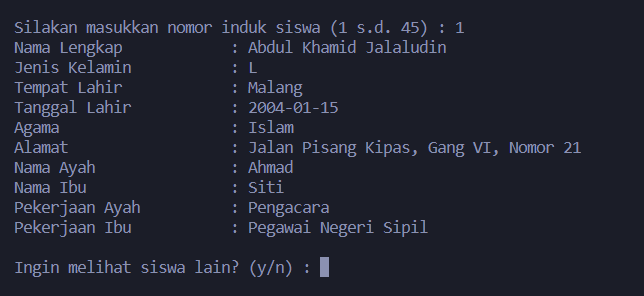
Gambar 2

Pilih menu biodata dengan mengetik “biodata”.



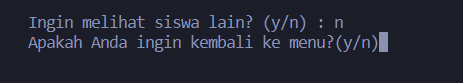
Gambar 3

Pilih siswa mana yang hendak dilihat biodatanya. Masukkan nomor induknya. Data siswa dapat di lihat seperti di bawah ini.



Gambar 4

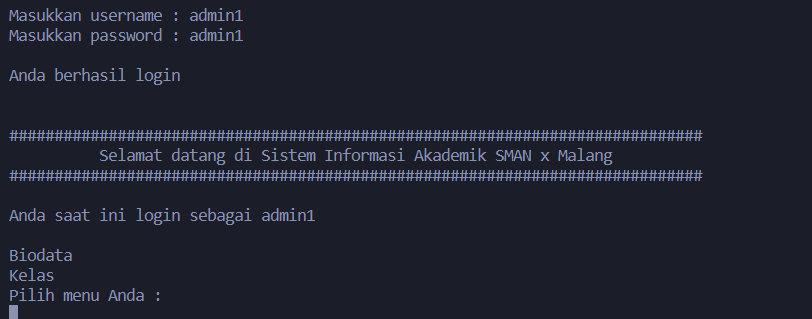
Silakan pilih **y** untuk biodata melihat siswa lain. Program akan mengulang langkah sebelumnya. Jika sudah selesai, silakan pilih **n.** Kemudian terdapat pilihan untuk kembali ke menu atau tidak. Jika pengguna memilih iya, program akan kembali ke bagian menu seperti pada Gambar 2. Jika pengguna memilih tidak, program akan berakhir.



Gambar 5

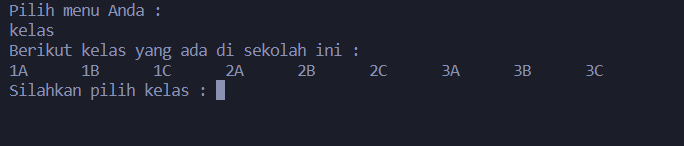
1. Menginputkan Nilai, Presensi, dan Mencetak Rapor

Jika pengguna sebelumnya-memilih untuk kembali ke menu, tampilan akan seperti ini.



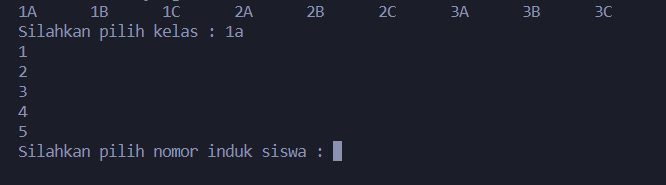
Gambar 6

Pilih menu kelas dengan mengetik “kelas”.



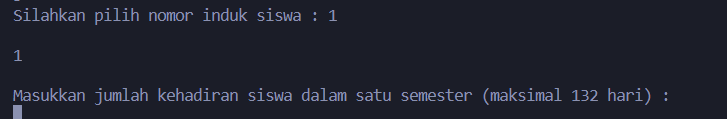
Gambar 7

Kemudian pilih kelas tempat murid yang hendak dimasukkan nilainya berasal. Semisalnya pengguna memilih kelas 1A, program akan menampilkan daftar anak yang berada pada kelas tersebut.



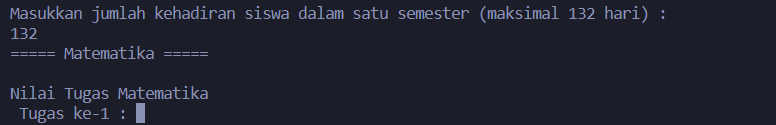
Gambar 8

Pilih salah satu nomor dari siswa yang berada dalam kelas. Selanjutnya ada bagan penginputan jumlah kehadiran siswa dalam satu semester.



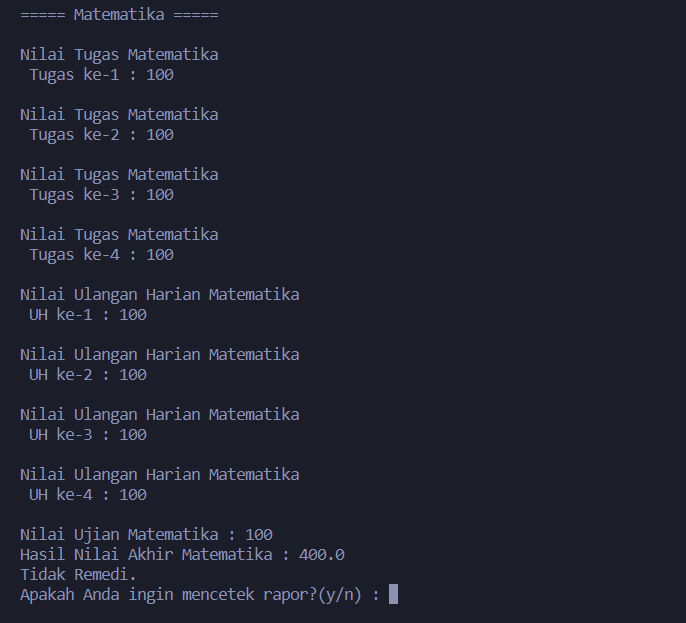
Gambar 9

Untuk demonstrasi, masukkan jumlah maksimum presensi. Selanjutnya, program mulai meminta penginputan nilai tugas mata pelajaran matematika.



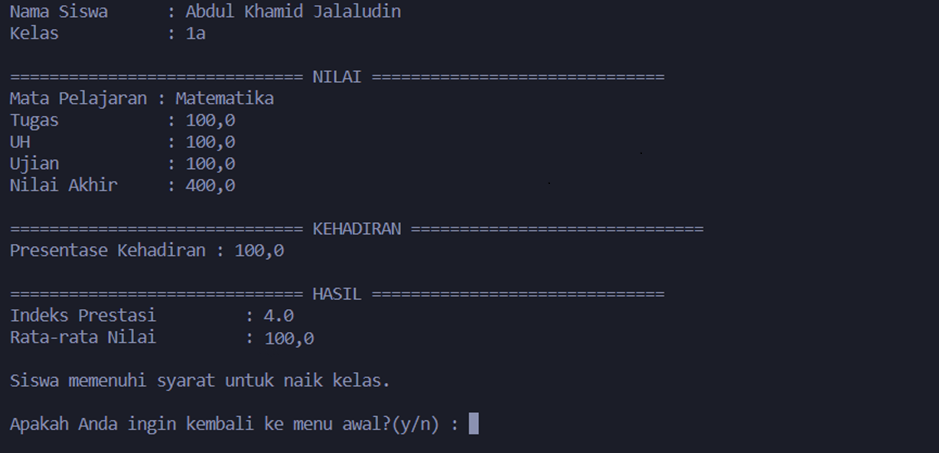
Gambar 10

Masukkan nilai-nilai sesuai pada tempatnya.



Gambar 11

Tekan y untuk melanjutkan ke pencetakan rapor.

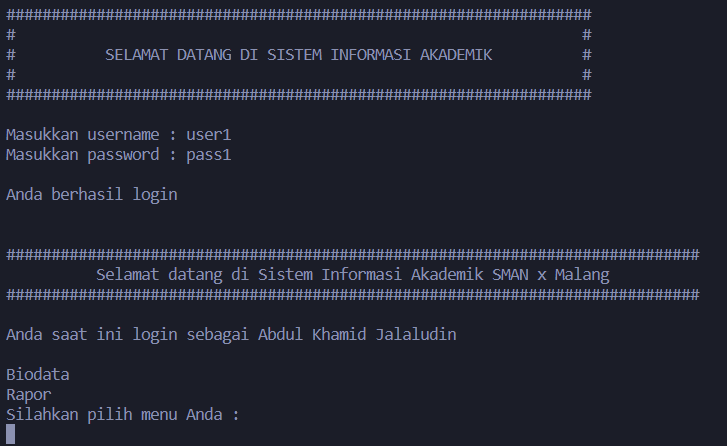


Gambar 12

Jika ingin kembali ke menu awal, tekan y. Jika ingin log out program, tekan n.

1. Alur Program Akun Murid

Login dengan akun-akun murid yang telah dibuatkan oleh admin.



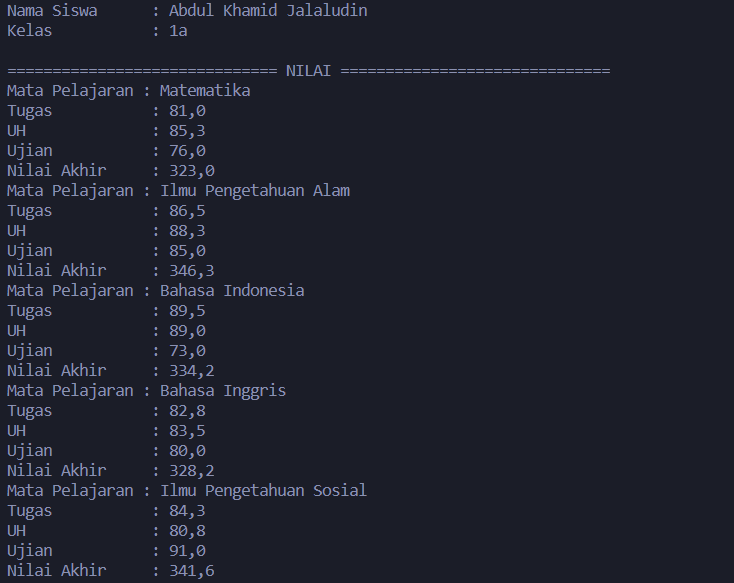
Gambar 13

Pertama, coba pilih menu biodata. Program akan menampilkan data diri sang pemilik akun. Selanjutnya, pilih y untuk melihat rapor.

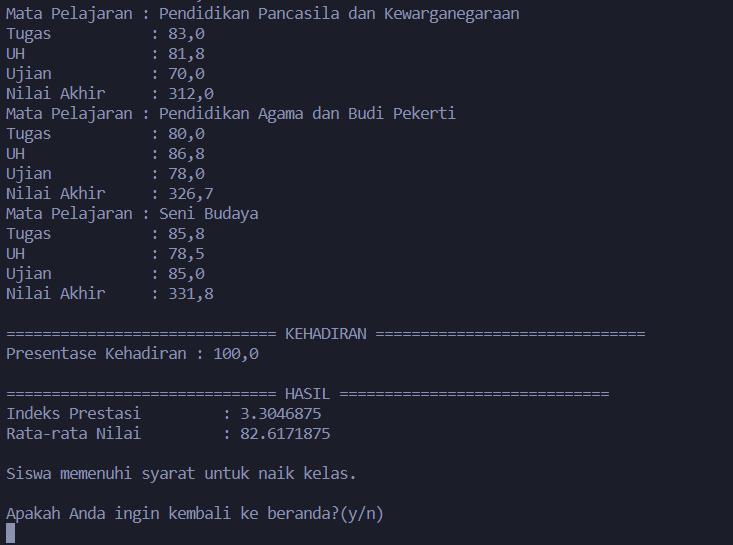


Gambar 14

Rapor terlihat seperti gambar di bawah. Di akhir bagan, terdapat keterangan apakah murid tersebut lulus atau tidak lulus.



Gambar 15



Gambar 16

1. PENUTUP
2. Kesimpulan

Adanya perkembangan teknologi informasi memudahkan pekerjaan administrasi manusia, termasuk dalam pengelolaan data di lingkungan sekolah. Kemudahan yang muncul meliputi bertambahnya kecepatan pengelolaan data, berkurangnya aktivitas pencatatan tangan, dan berkurangnya kemungkinan kesalahan penulisan data.

Pemakaian teknologi informasi pada bidang pengelolaan data, admisi, dan surat menyurat juga dapat mengurangi tindakan kecurangan dan penyelewengan wewenang. Program komputer berjalan dengan prosedur yang berurut dan logis, sehingga intervensi oleh suatu pihak tidak dapat dilakukan sebagaimana yang dilakukan pada pencatatan manual. Setiap pengguna yang mengakses suatu program memiliki cakupan akses yang berbeda-beda, sehingga pengguna tidak dapat memanipulasi data yang berada di luar ranahnya kecuali dengan melakukan peretasan.

Oleh karena itu, penggunaan sistem informasi akademik merupakan sebuah loncatan inovasi yang sangat baik. Kini, siakad telah digunakan secara masif oleh sekolah-sekolah di Indonesia.

Lampiran

Profil GitHub penulis : <https://github.com/VozSoldat>

Utas GitHub repositori kelompok : <https://github.com/aorysan/sistemAkademik>