

LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT

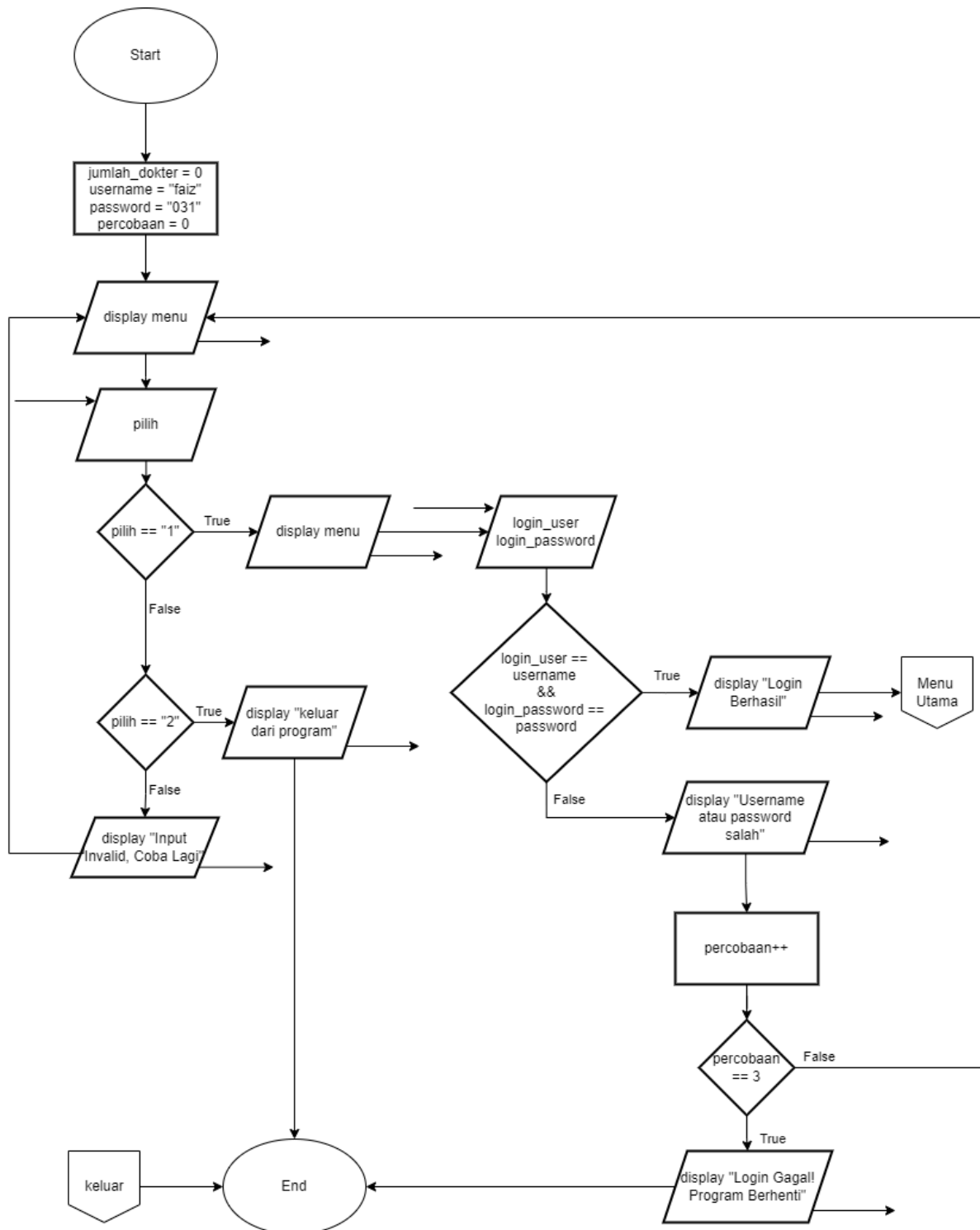


Disusun oleh:
Muhammad Faiz Lazuardi (2409106031)
Kelas (A2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

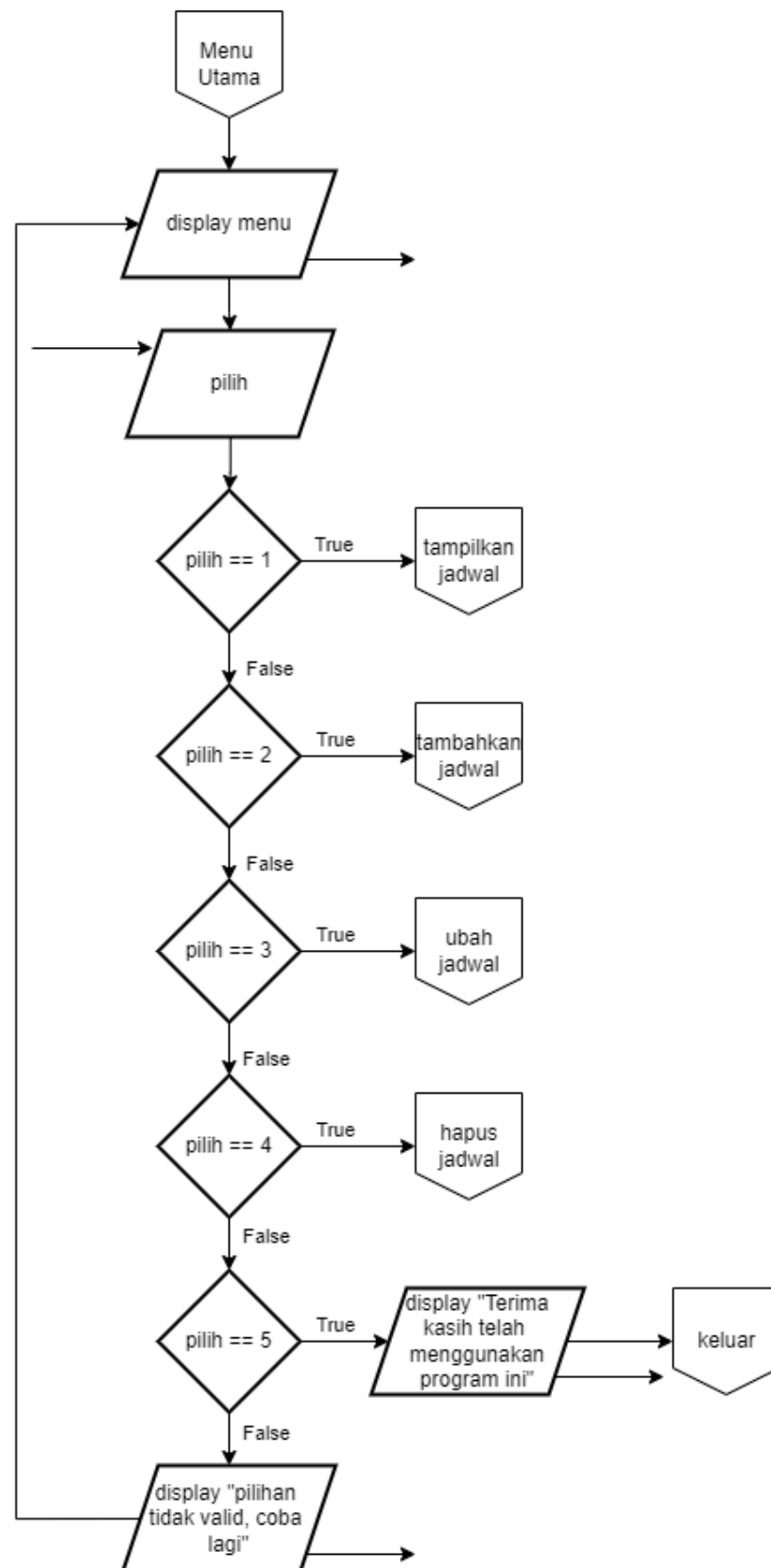
1. Flowchart

1.1 Menu Login



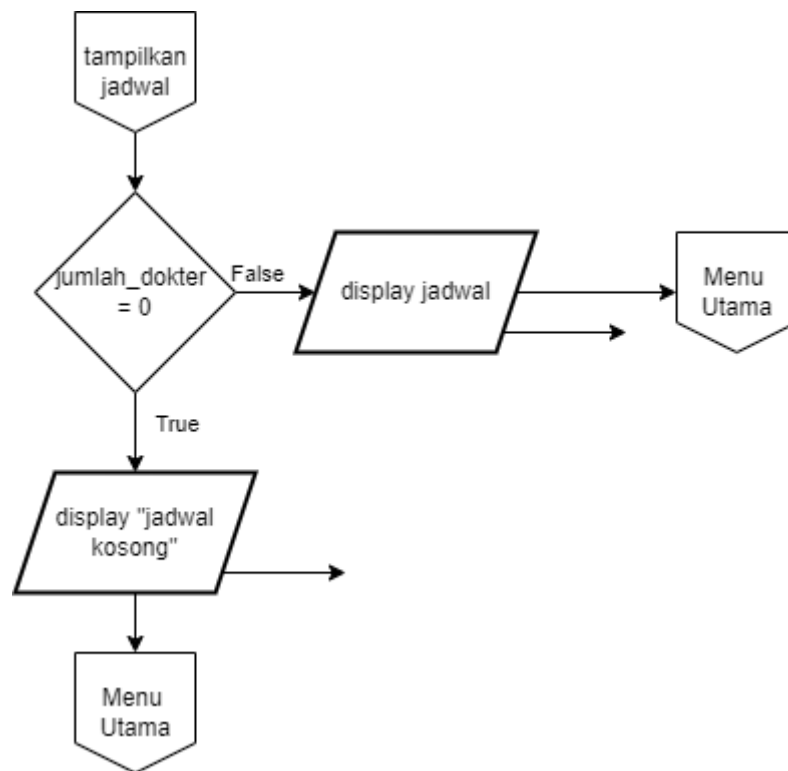
Gambar 1.1 Menu Login

1.2 Menu Utama



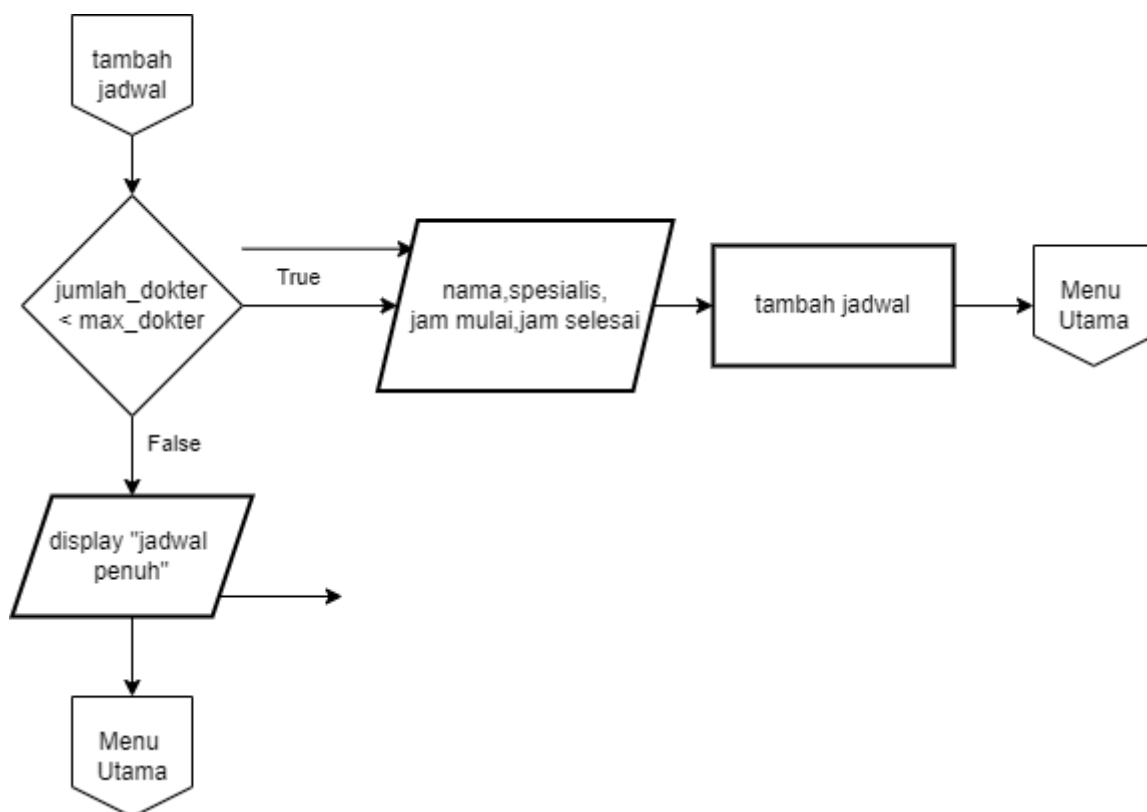
Gambar 1.2 Menu Utama

1.3 Menu Read



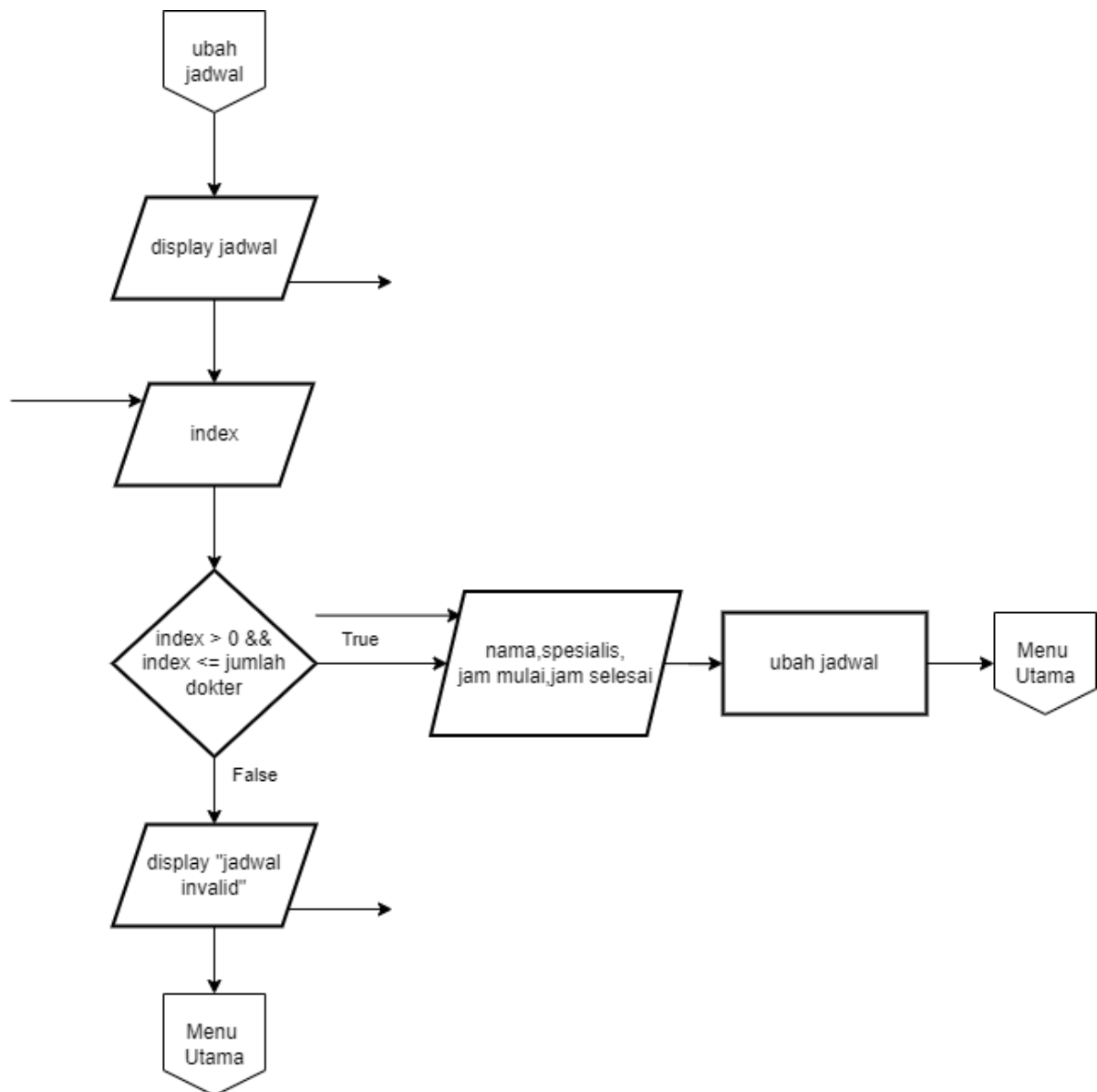
Gambar 1.3 Menu Read

1.4 Menu Create



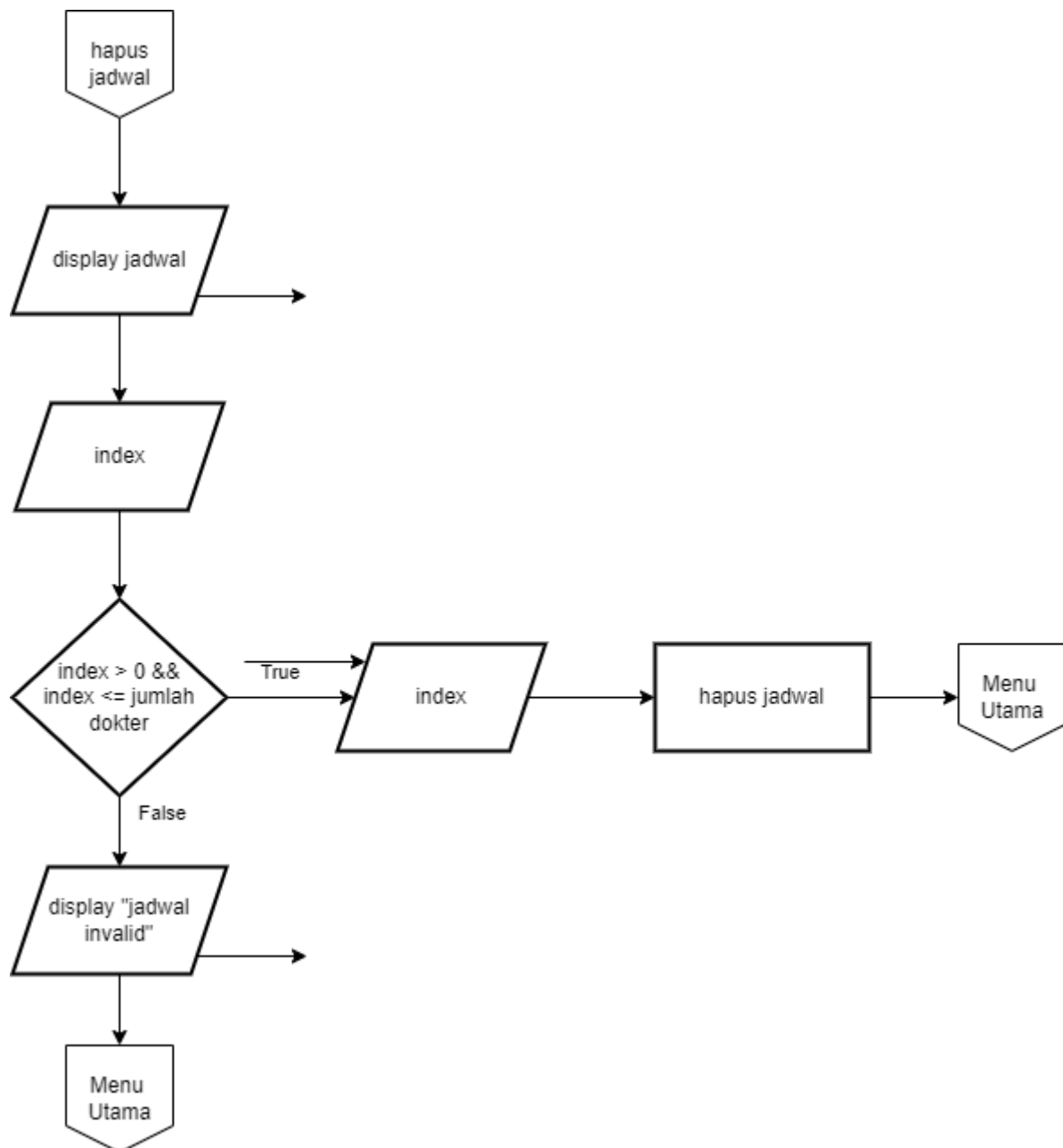
Gambar 1.4 Menu Create

1.5 Menu Update



Gambar 1.5 Menu Update

1.6 Menu Delete



Gambar 1.6 Menu Delete

2. Analisis Program

Sebuah program yang berfungsi untuk memanage jadwal praktek dokter spesialis.

3. Source Code

A. Login

Fitur ini digunakan sebagai syarat untuk menjalankan program manajemen jadwal praktek dokter spesialis. Data yang diminta berupa username dan password.

Source Code:

```
int percobaan = 0;
string username = "faiz", password = "031", login_user, login_password,
pilih;
while (percobaan < 3) {
    cout << "<==== Home Page =====>\n";
    cout << "1. Login\n";
    cout << "2. Keluar\n";
    cout << "Pilih Fitur: "; cin >> pilih;

    if(pilih == "1"){
        cout << "\n<==== Login Page =====>\n";
        cout << "Username: "; cin >> login_user;
        cout << "Password: "; cin >> login_password;
        system("cls");

        if (login_user == username and login_password == password) {
            cout << "Login berhasil!\n";
            return 0;
        } else {
            cout << "Username atau password salah!\n";
            percobaan++;
            if (percobaan == 3) {
                cout << "Login gagal! Program Berhenti.\n";
                exit(0);
            }
        }
    } else if(pilih == "2"){
        cout << "Keluar dari program.\n";
        exit(0);
    } else {
        cout << "Input Yang Anda Pilih Tidak Valid. Silahkan Coba
lagi.\n\n";
    }

}
return 0;
}
```

B. Menu Utama

Menu Utama berisi beberapa pilihan fitur. Pilihan 1 sampai 4 merupakan fitur CRUD dan pilihan 5 merupakan fitur keluar dari program.

Source Code:

```
cout << "\n<==== Jadwal Praktek Dokter >====>\n";
cout << "1. Tampilkan Jadwal\n";
cout << "2. Tambah Jadwal\n";
cout << "3. Ubah Jadwal\n";
cout << "4. Hapus Jadwal\n";
cout << "5. Keluar\n";
cout << "Pilih Fitur: "; cin >> pilih;
```

C. Pilihan 1

Menampilkan jadwal.

Source Code:

```
case 1:
    if (jumlah_dokter == 0) {
        cout << "Jadwal Kosong" << endl;
    }

    for(int i = 0; i < jumlah_dokter; ++i){
        cout << i + 1 << ". Nama: " << jadwal_praktek[i].nama_dokter << endl;
        cout << "    " << "Spesialis: " << jadwal_praktek[i].spesialis << endl;
        cout << "    " << "Jam Mulai: " << jadwal_praktek[i].jadwal.jam_mulai <<
endl;
        cout << "    " << "Jam Selesai: " << jadwal_praktek[i].jadwal.jam_selesai
<< endl;
        cout << "    " << endl;
    }

    break;
```


D. Pilihan 2

Menambahkan jadwal.

Source Code:

```
case 2:
    if (jumlah_dokter < MAX_DOKTER) {
        cout << "Masukkan Nama Dokter: "; cin >>
        jadwal_praktek[jumlah_dokter].nama_dokter;
        cout << "Masukkan Spesialis: "; cin >>
        jadwal_praktek[jumlah_dokter].spesialis;
        cout << "Masukkan Jam Mulai: "; cin >>
        jadwal_praktek[jumlah_dokter].jadwal.jam_mulai;
        cout << "Masukkan Jam Selesai: "; cin >>
        jadwal_praktek[jumlah_dokter].jadwal.jam_selesai;
        jumlah_dokter++;
        cout << "Jadwal berhasil ditambahkan!\n";

    } else {
        cout << "Jadwal Penuh.\n";
    }

    break;
```

E. Pilihan 3

Mengubah jadwal.

Source Code:

```
case 3:
    if (jumlah_dokter == 0) {
        cout << "Jadwal Kosong" << endl;
    }

    for(int i = 0; i < jumlah_dokter; ++i){
        cout << i + 1 << ". Nama: " << jadwal_praktek[i].nama_dokter << endl;
        cout << "    " << "Spesialis: " << jadwal_praktek[i].spesialis << endl;
        cout << "    " << "Jam Mulai: " << jadwal_praktek[i].jadwal.jam_mulai <<
        endl;
        cout << "    " << "Jam Selesai: " << jadwal_praktek[i].jadwal.jam_selesai
        << endl;
        cout << "    " << endl;
    }
```

```

        if(jumlah_dokter == 0) {

        } else {
            int index;
            cout << "Pilih Jadwal Yang Ingin Di Ubah: ";
            cin >> index;
            if (index > 0 && index <= jumlah_dokter) {
                cout << "Masukkan Nama Dokter baru: "; cin >> jadwal_praktek[index - 1].nama_dokter;
                cout << "Masukkan Spesialis baru: "; cin >> jadwal_praktek[index - 1].spesialis;
                cout << "Masukkan Jam Mulai baru: "; cin >> jadwal_praktek[index - 1].jadwal.jam_mulai;
                cout << "Masukkan Jam Selesai baru: "; cin >> jadwal_praktek[index - 1].jadwal.jam_selesai;
                cout << "Jadwal berhasil diperbarui!\n";

            } else {
                cout << "Jadwal Tidak Valid!\n";
            }
        }
        break;

```

F. Pilihan 4

Menghapus jadwal.

Source Code:

```

case 4:
    if (jumlah_dokter == 0) {
        cout << "Jadwal Kosong" << endl;
    }
    for(int i = 0; i < jumlah_dokter; ++i){
        cout << i + 1 << ". Nama: " << jadwal_praktek[i].nama_dokter << endl;
        cout << "    " << "Spesialis: " << jadwal_praktek[i].spesialis << endl;
        cout << "    " << "Jam Mulai: " << jadwal_praktek[i].jadwal.jam_mulai << endl;
        cout << "    " << "Jam Selesai: " << jadwal_praktek[i].jadwal.jam_selesai << endl;
        cout << "    " << endl;
    }

    if(jumlah_dokter == 0) {

    } else {

```

```

int index;
cout << "Pilih Jadwal Yang Ingin Di Hapus: ";
cin >> index;
if (index > 0 && index <= jumlah_dokter) {
    for (int i = index - 1; i < jumlah_dokter - 1; i++) {
        jadwal_praktek[i] = jadwal_praktek[i + 1];
    }
    jumlah_dokter--;
    cout << "Jadwal berhasil dihapus!\n";

} else {
    cout << "Jadwal Tidak Valid!\n";
}
}
break;

```

4. Screenshot Program

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 struct jam_praktek {
6     string jam_mulai;
7     string jam_selesai;
8 };
9
10 struct jadwal_dokter {
11     string nama_dokter;
12     string spesialis;
13     jam_praktek jadwal;
14 };
15
16 const int MAX_DOKTER = 10;
17 int jumlah_dokter = 0;
18 jadwal_dokter jadwal_praktek[MAX_DOKTER];
19
20 int main() {
21     int percobaan = 0;
22     string username = "fai", password = "B41", login_user, login_password, pilih;
23     while (percobaan < 4) {
24         cout << "<----- Home Page ----->\n";
25         cout << "1. Login\n";
26         cout << "2. Keluar\n";
27         cout << "Pilih Fitur: "; cin >> pilih;
28
29         if(pilih == "1"){
30             cout << "\n<----- Login Page ----->\n";
31             cout << "Username: "; cin >> login_user;
32             cout << "Password: "; cin >> login_password;
33             system("cls");
34
35             if (login_user == username and login_password == password) {
36                 cout << "Login berhasil!\n";
37
38                 while (true) {
39                     int pilih;
40                     cout << "\n<===== Jadwal Praktek Dokter =====>\n";
41                     cout << "1. Tampilkan Jadwal\n";
42                     cout << "2. Tambah Jadwal\n";
43                     cout << "3. Ubah Jadwal\n";
44                     cout << "4. Hapus Jadwal\n";
45                     cout << "5. Keluar\n";
46                     cout << "Pilih Fitur: "; cin >> pilih;
47                     system("cls");
48
49                     switch (pilih) {
50                         case 1:
51                             if (jumlah_dokter == 0) {
52                                 cout << "Jadwal Kosong" << endl;
53                             }
54
55                             for(int i = 0; i < jumlah_dokter; ++i){
56                                 cout << i + 1 << ". Nama: " << jadwal_praktek[i].nama_dokter << endl;
57                                 cout << " " << "Spesialis: " << jadwal_praktek[i].spesialis << endl;
58                                 cout << " " << "Jam Mulai: " << jadwal_praktek[i].jadwal.jam_mulai << endl;
59                                 cout << " " << "Jam Selesai: " << jadwal_praktek[i].jadwal.jam_selesai << endl;
60                                 cout << " " << endl;
61                             }
62
63                             break;
64                         case 2:
65                             if (jumlah_dokter < MAX_DOKTER) {
66                                 cout << "Masukkan Nama Dokter: "; cin >> jadwal_praktek[jumlah_dokter].nama_dokter;
67                                 cout << "Masukkan Spesialis: "; cin >> jadwal_praktek[jumlah_dokter].spesialis;
68                                 cout << "Masukkan Jam Mulai: "; cin >> jadwal_praktek[jumlah_dokter].jadwal.jam_mulai;
69                                 cout << "Masukkan Jam Selesai: "; cin >> jadwal_praktek[jumlah_dokter].jadwal.jam_selesai;
70                                 jumlah_dokter++;
71                                 cout << "Jadwal berhasil ditambahkan!\n";
72
73                             } else {
74                                 cout << "Jadwal Penuh.\n";
75                             }
76
77                             break;
```

Gambar 4.1 Code Program

```

78         case 3:
79             if (jumlah_dokter == 0) {
80                 cout << "Jadwal Kosong" << endl;
81             }
82             for(int i = 0; i < jumlah_dokter; ++i){
83                 cout << i + 1 << ". Nama: " << jadwal_praktek[i].nama_dokter << endl;
84                 cout << " " << "Spesialis: " << jadwal_praktek[i].spesialis << endl;
85                 cout << " " << "Jam Mulai: " << jadwal_praktek[i].jadwal.jam_mulai << endl;
86                 cout << " " << "Jam Selesai: " << jadwal_praktek[i].jadwal.jam_selesai << endl;
87                 cout << " " << endl;
88             }
89
90             if(jumlah_dokter == 0) {
91
92             } else {
93                 int index;
94                 cout << "Pilih Jadwal Yang Ingin Di Ubah: ";
95                 cin >> index;
96                 if (index > 0 && index <= jumlah_dokter) {
97                     cout << "Masukkan Nama Dokter baru: "; cin >> jadwal_praktek[index - 1].nama_dokter;
98                     cout << "Masukkan Spesialis baru: "; cin >> jadwal_praktek[index - 1].spesialis;
99                     cout << "Masukkan Jam Mulai baru: "; cin >> jadwal_praktek[index - 1].jadwal.jam_mulai;
100                     cout << "Masukkan Jam Selesai baru: "; cin >> jadwal_praktek[index - 1].jadwal.jam_selesai;
101                     cout << "Jadwal berhasil diperharui!\n";
102
103                 } else {
104                     cout << "Jadwal Tidak Valid!\n";
105                 }
106             }
107             break;
108
109         case 4:
110             if (jumlah_dokter == 0) {
111                 cout << "Jadwal Kosong" << endl;
112             }
113             for(int i = 0; i < jumlah_dokter; ++i){
114                 cout << i + 1 << ". Nama: " << jadwal_praktek[i].nama_dokter << endl;
115                 cout << " " << "Spesialis: " << jadwal_praktek[i].spesialis << endl;
116                 cout << " " << "Jam Mulai: " << jadwal_praktek[i].jadwal.jam_mulai << endl;
117                 cout << " " << "Jam Selesai: " << jadwal_praktek[i].jadwal.jam_selesai << endl;
118                 cout << " " << endl;
119             }
120
121             if(jumlah_dokter == 0) {
122
123             } else {
124                 int index;
125                 cout << "Pilih Jadwal Yang Ingin Di Hapus: ";
126                 cin >> index;
127                 if (index > 0 && index <= jumlah_dokter) {
128                     for (int i = index - 1; i < jumlah_dokter - 1; i++) {
129                         jadwal_praktek[i] = jadwal_praktek[i + 1];
130                     }
131                     jumlah_dokter--;
132                     cout << "Jadwal berhasil dihapus!\n";
133
134                 } else {
135                     cout << "Jadwal Tidak Valid!\n";
136                 }
137             }
138             break;
139         case 5:
140             cout << "Terima kasih telah menggunakan program ini!\n";
141             exit(0);
142         default:
143             cout << "Pilihan tidak valid! Coba lagi.\n";
144     }
145 }
146
147
148 } else {
149     cout << "Username atau password salah!\n";
150     percobaan++;
151     if (percobaan == 4) {
152         cout << "Login gagal! Program Berhenti.\n";
153         exit(0);
154     }
155 }
156 }else if(pilih == '2'){
157     cout << "Keluar dari program.\n";
158     exit(0);
159 }else {
160     cout << "Input Yang Anda Pilih Tidak Valid. Silahkan Coba lagi.\n\n";
161 }
162
163 }
164 return 0;
165 }

```

Gambar 4.2 Code Program

5. Git

5.1 Git Init

Git init merupakan command yang berfungsi untuk menginisiasi repository yang ada pada file lokal dan berlokasi di folder .git. git branch -m master main merupakan command yang berfungsi untuk merubah branch repository dari master ke main.

```
ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop
$ mkdir praktikum-apl

ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop
$ cd praktikum-apl/

ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl
$ pwd
/c/Users/ASUS/Desktop/praktikum-apl

ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl
$ git init
Initialized empty Git repository in C:/Users/ASUS/Desktop/praktikum-apl/.git/

ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl (master)
$ git branch -m master main

ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl (main)
```

Gambar 5.1 Git init dan Change branch

5.2 Git Add

Menambahkan file yang ingin di commit.

```
ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl/post-test (main)
$ git add .
```

Gambar 5.2 Membuat File dan Git add

5.3 Git Commit

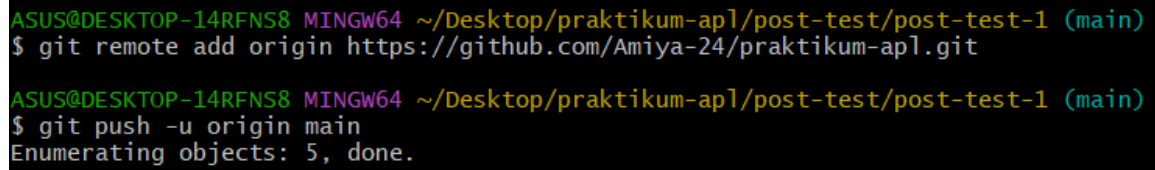
Menyimpan perubahan yang disimpan pada folder lokal dengan sebuah pesan.

```
ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl/post-test (main)
$ git commit -m "sementara"
[main 482f76f] sementara
 2 files changed, 165 insertions(+)
 create mode 100644 post-test/post-test-3/2409106031-MuhammadFaizLazuardi-PT-3.c
pp
 create mode 100644 post-test/post-test-3/2409106031-MuhammadFaizLazuardi-PT-3.e
xe
```

Gambar 5.3 Git Commit

5.4 Git Remote dan Git Push

Git remote berfungsi untuk menghubungkan repository yang ada di lokal dengan repository cloud pada github. Git push berfungsi untuk mengupload semua yang ada di lokal ke github.

A screenshot of a terminal window with a black background and green text. The prompt is 'ASUS@DESKTOP-14RFNS8 MINGW64 ~'. The first command is 'git remote add origin https://github.com/Amiya-24/praktikum-apl.git'. The second command is 'git push -u origin main'. The output of the second command is 'Enumerating objects: 5, done.'.

```
ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl/post-test/post-test-1 (main)
$ git remote add origin https://github.com/Amiya-24/praktikum-apl.git

ASUS@DESKTOP-14RFNS8 MINGW64 ~/Desktop/praktikum-apl/post-test/post-test-1 (main)
$ git push -u origin main
Enumerating objects: 5, done.
```

Gambar 5.4 Git Remote dan Git Push